

Національний лісотехнічний університет України
(повне найменування вищого навчального закладу)

Навчально-науковий інститут деревообробних та
комп'ютерних технологій і дизайну
(повне найменування інституту, назва факультету (відділення))

Кафедра інформаційних технологій
(повна назва кафедри (предметної, циклової комісії))

Пояснювальна записка
до дипломної роботи

другий (магістерський)
(рівень вищої освіти)

на тему: Система підтримки прийняття рішень при виборі авто в Україні
чи з-за кордону

Виконав: студент _____ курсу групи _____
спеціальності
122 “Комп’ютерні науки”
(шифр і назва напрямку підготовки, спеціальності)

Бачало Р.В.
(прізвище та ініціали)

Керівник Крошній І.М.
(прізвище та ініціали)

Рецензент _____
(прізвище та ініціали)

Львів – 2022

Національний лісотехнічний університет України

(повне найменування вищого навчального закладу)

ННІ деревообробних та комп'ютерних технологій і дизайну

Кафедра інформаційних технологій

Рівень вищої освіти другий (магістерський)

Спеціальність 122 “Комп'ютерні науки”

(шифр і назва)

ЗАТВЕРДЖУЮ
Завідувач кафедри

_____ Крошній І. М.
“ _____ ” _____ 20__ року

З А В Д А Н Н Я
НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Бачалові Ростиславові Володимировичу

(прізвище, ім'я, по батькові)

1. Тема роботи Система підтримки прийняття рішень при виборі авто в Україні чи з-за кордону

керівник роботи Крошній Ігор Миколайович доцент, к.т.н., доцент,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від “20” 12 2021 року № C-617

2. Термін подання студентом роботи 12.12.2022

3. Вихідні дані до роботи 1) Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking 1st Edition, Kindle Edition - O'Reilly Media; 1st edition July 27, 2013 2) The Art of Data Science Paperback – Lulu.com June 8, 2016 3) Understanding Machine Learning: From Theory to Algorithms - Cambridge University Press; 1st edition May 19, 2014

4) Pascal Bornet. INTELLIGENT AUTOMATION: Learn how to harness Artificial Intelligence to boost business & make our world more human - Kindle Edition October 14, 2020

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити) 1) Стан проблемної області 2) Інформаційне забезпечення 3) Математичне забезпечення 4) Програмне забезпечення 5) Розроблення стартап проекту

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
Презентація

6. Дата видачі завдання 20.12.2021 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/П	Назва етапів дипломної роботи	Строк виконання етапів роботи	Примітка
1	Огляд інформаційних джерел, що описують облік ефективності діяльності персоналу	20.12.2021-08.01.2022	
2	Огляд наявних в мережі Інтернет схожих програмних продуктів.	08.01.2022-30.01.2022	
3	Огляд та порівняльний аналіз сучасних мов програмування та технологій	30.01.2022-02.02.2022	
4	Встановлення програмного середовища та побудова архітектури програмного продукту	02.02.2022-04.05.2022	
5	Розробка бекенд частини продукту	04.05.2022-22.07.2022	
6	Розробка допоміжних систем для парсингу та обробки даних	22.07.2022-01.08-2022	
7	Розробка фронтенд частини продукту для візуалізації даних та результатів	01.08-2022-06.10.2022	
8	Тестування розробленого програмного продукту	06.10-2022-06.11.2022	
9	Внесення змін в програмний код після виявлення несправностей під час тестування	06.11-2022-28.11.2022	
10	Внесення змін в програмний код після виявлення несправностей під час тестування	06.11-2022-28.11.2022	
11	Написання пояснювальної записки	28.11-2022-5.12.2022	

Студент

_____ (підпис)

Бачало Р. В

_____ (прізвище та ініціали)

Керівник роботи

_____ (підпис)

Крошній І. М

_____ (прізвище та ініціали)

РЕФЕРАТ

Дипломна робота містить 68 сторінок пояснювальної записки, 35 рисунків, 1 таблицю, 2 додатки, 31 джерел.

В даній роботі було розроблено тривірневий веб-застосунок для підтримки прийняття рішень при виборі автомобіля в Україні чи з-за кордону, за допомогою асинхронного парсингу в реальному часі даних з веб-ресурсів. Також було імплементовано алгоритм прогнозування вартості автомобіля на основі його характеристик за допомогою машинного навчання. Веб-застосунок створений для надання допомоги користувачам орієнтуватись в ціновому сегменті, автомобільного ринку, як при купівлі автомобіля, так і при його продажі.

Ключові слова: C#, Angular, Web API, ML.NET машинне навчання, регресія, автомобіль, оцінка.

ABSTRACT

The thesis contains 60 pages of explanatory note, 35 figures, 1 table, 2 appendices, 31 sources.

In this work, a three-level web application was developed to support decision-making when choosing a car in Ukraine or from abroad using asynchronous real-time parsing of data from web resources. An algorithm for predicting the value of a car based on its characteristics using machine learning was also implemented. Web creation system to help users navigate the price segment of the automobile market both when buying a car and when selling it.

Keywords: C#, Angular, Web API, ML.NET machine learning, regression, car, estimation.

ТЕХНІЧНЕ ЗАВДАННЯ

Необхідно розробити веб-застосунок для парсингу та відображення даних про автомобілі в реальному часі, а також прогнозування вартості атомобіля за допомогою штучного навчання, а саме:

1. Побудувати трирівневу архітектуру застосунку на основі ASP.NET Core Web API на бекенді;
2. Створити парсер для наповнення бази даних з назвами марок та моделей автомобілей;
3. Розробити методи асинхронного парсингу інтернет ресурсів для порівняння цін автомобілей в реальному часі;
4. Створити метод валідації даних на основі регулярних виразів;
5. На основі Robotic Process Automation розробити робота для парсингу даних та наповнення вибірки в середовищі UiPath;
6. Провести тренування моделі на основі зібраних даних та інтегрувати це в застосунок за допомогою ML.Net;
7. Розробити фронтенд частину застосунку на основі Angular;

ВСТУП.....	8
РОЗДІЛ 1. СТАН ПРОБЛЕМНОЇ ОБЛАСТІ.....	10
1.1 Актуальність автомобілів з Європи.....	10
1.2 Оцінка вартості автомобіля	11
1.3 Застосування машинного навчання в бізнесі.....	11
Висновки до розділу	12
РОЗДІЛ 2. ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ	13
2.1 Реалізація збору даних для оцінки вартості автомобіля	13
2.2 Опис даних для порівняння автомобілів.....	14
2.3 Розмитнення автомобіля	14
Висновки до розділу	16
РОЗДІЛ 3. МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ.....	17
3.1 Машинне навчання.....	17
3.1.1 Підкатегорії машинного навчання.....	18
3.2 Регресійний аналіз в машинному навчанні.....	19
3.2.1 Види регресій.....	20
3.3 Інтелектуальна автоматизація	20
Висновки до розділу	21
РОЗДІЛ 4. ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ	22
4.1. Використані програмні середовища.....	22
4.1.1 Visual Studio 2019	22
4.1.2 Visual Studio Code.....	23
4.1.3 UiPath Studio	23
4.2 Опис трирівневої архітектури.....	23
4.3 Додаткові інструменти та фреймворки	24
4.3.1 ASP.Net CORE	24
4.3.2 Web API.....	25
4.3.3 ML.Net.....	26
4.3.4 Angular	27
4.3.5 HTML AGILITY PACK.....	28
4.3.6 SQL Server Management Studio (SSMS).....	28
4.3.7 Postman.....	28
4.4 Ключові моменти реалізації системи	29
4.5 Тестування та візуалізація	37
Висновки до розділу	41
РОЗДІЛ 5. РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ	42
5.1 Опис ідеї проекту	42
5.2 Аналіз технологічних можливостей реалізації ідеї проекту	43
5.3 Аналіз ринкових можливостей запуску стартап-проекту	43
5.3.1 Аналіз проектів конкурентів.....	43
5.4 Розроблення маркетингової системи.....	44
5.4.1 Стратегія монетизації проекту.....	44
5.4.2 Просування проекту	45
Висновки до розділу	46
ВИСНОВКИ	47

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	48
ДОДАТКИ	51
ДОДАТОК А	51
ДОДАТОК Б	60

ВСТУП

Кількість автомобілей на дорогах збільшується з кожним роком. За понад сто років приватний автомобіль змінив сучасне суспільство, забезпечивши незалежність і свободу пересування. Мобільність стає все більш важливою через постійно зростаючі відстані між домом, роботою, навчальними закладами, магазинами та закладами відпочинку. Автомобілі є нашим джерелом мобільності номер один, пересічний українець проїжджає майже 12 000 кілометрів на рік. Майже 80% внутрішніх поїздок здійснюються на автомобілі – будь то приватний автомобіль чи таксі. Без індивідуальної мобільності адекватна участь у соціальному та економічному житті у багатьох випадках була б неможливою, особливо для людей, які живуть у віддалених районах, людей похилого віку та осіб з обмеженими можливостями. Люди, які живуть у містах, можуть покладатися на таксі, або послуги на замовлення замість власного приватного автомобіля або як доповнення до нього. Мобільність, пов'язана з роботою, має особливе значення, оскільки близько половини всіх кілометрів, пройдених транспортними засобами, пов'язані з роботою.

Вартість нових автомобілей не є доступною для більшості людей у світі, тому вживані атомобілі завжди актуальні. Ринок вживаних автомобілів в Україні динамічно розвивається, найбільшим попитом користуються авто з Європи. Виявлено, що українці цікавляться автомобілями різного цінового та класового сегменту. Разом із цим, кожен хто підходить до етапу купівлі автомобіля стикається з проблемою вибору, адже не завжди бажання та потреби співпадають з виділеним бюджетом. Проте, кожен, зацікавлений в тому, щоб вигідно купити автомобіль у відмінному стані. Більшу частину українського ринку, займають авто пригнані з Європи. Це спричинено низькою ціною, старіших автомобілів, в Європі та їхнім хорошим технічним станом. Всі, хто розглядають можливості пригону автомобіля, цікавляться різницею ціни автомобіля в Україні та іншій європейській країні. Потенційні покупці самостійно проводять дослідження чи є суттєва вигода в самостійному пригоні, або купівлі вже пригнаного, транспортного зособу. Такі дослідження, серед людей які в цьому не розбираються, в більшості випадків, не дають ніяких результатів.

Актуальність проблеми також обумовлена, недобросовісними диллерами, вживаних європейських авто, які істотно завишають ціни на пригнані автомобілі, а також неорієнтованість покупців у ціновій політиці. Було з'ясовано, що в Україні відсутні, будь-які, сервіси які б допомогли вирішити цю проблему.

Варто зазначити, що не менш актуальною є проблема продажу власного автомобіля. Новий автомобіль після виїзду за межі автосалону одразу втрачає 20% від вартості. Тобто, людина, яка купувала авто в салоні, не зможе його продати по ціні купівлі, навіть не враховуючи рік, пробіг та інші деталі. Для того, щоб виставити автомобіль на торгову площадку, продавець, повинен проаналізувати цінову політику схожих автомобілей, щоб правильно оцінити власний.

Машинне навчання вже давно застосовується для вирішення багатьох користувацьки проблем. Його можна використовувати для прогнозного аналізу, який в основному є обробкою великих обсягів даних. Машинне навчання у веб-застосунках використовують для асиміляції безпеки та фільтрації шкідливих даних та багато іншого.

Метою роботи було створити систему підтримки прийняття рішень для аналізу цін та вибору автомобіля, за допомогою створення алгоритму парсингу, актуальних автомобілів з європейських та українських торгових площадок, для порівняння цін, а також за допомогою штучного інтелекту, оцінювати приблизну вартість, на основі параметрів автомобіля. Реалізувати все у вигляді користувацького веб – застосунку.

Практична значимість. За допомогою зрозумілого інтерфейсу люди зможуть з легкістю порівнювати ціни на авто в Україні та Європі, а також отримати наближену вартість автомобіля, який їх цікавить, за допомогою машинного навчання.

Об'єкт дослідження – система підтримки прийняття рішень при виборі авто в Україні чи з-за кордону.

Предмет дослідження – алгоритми парсингу даних та машинне навчання в системі підтримки прийняття рішення.

РОЗДІЛ 1. СТАН ПРОБЛЕМНОЇ ОБЛАСТІ

1.1 Актуальність автомобілів з Європи

Популярність машин з Європи зумовлена, у першу чергу, географічним положенням України. Привезти авто з Європи – швидше та простіше, ніж доставити з Америки. Окрім того, в Європі значно більше шансів купити «не биту» машину. До основних переваг пригону автомобіля з Європи можна віднести:

- **Ціна.** Враховуючи усі витрати на розмитнення, транспортування і тд., пригнане авто, в більшості випадках, буде дешевшим, ніж аналогічне на українському ринку;
- **Технічний стан.** Європейці дбають про технічний стан свого авто, адже там це є обов'язковим. Також варто звернути увагу на стан доріг у Європі, через відсутність поганого дорожнього покриття, ходова частина автомобіля служить довше;
- **Пробіг автомобіля.** Через заборону скручування пробігу автомобілів у Європі, завжди можна знати справжній пробіг, тому під час підбору авто це буде великою перевагою.
- **Експортування.** Багатьох від пригону автомобіля з Європи відштовхує питання експортування, тобто як доставити авто з європейського міста. У теперішній час існує безліч українських компаній які надають послуги перевезення авто в Україну за відносно не велику ціну. Також варто врахувати географічне положення України, тому доставка автомобіля не буде проблемою.

На сьогодні велика кількість автомобілів на українському ринку пригнані та продаються вже розмитнені, проте їх продають за ціною українського ринку. Тому якщо покупець знає, що автомобіль який він розглядає є приганим з-за кордону, за допомогою розробленого сервісу він зможе приблизно оцінити скільки він може переплатити.

1.2 Оцінка вартості автомобіля

Оцінка вартості автомобіля, в більшості, потрібна у двох випадках. Перше – це коли людина знає приблизні параметри вживаного автомобіля який вона б розглянула для купівлі, проте не орієнтується в його ціні, і їй цікаво яка вартість такого автомобіля і які параметри найбільше впливають на ціну. Друге – це коли людина хоче продати свій автомобіль, на вживаному ринку, адже після першого власника автомобіль перестає вважатись новим і втрачає свою цінність. Також варто зауважити такі параметри як пробіг автомобіля, рік його випуску, марку, модель, які в сумі формують ціну автомобіля. Тому для двох випадків потрібна оцінка авто, але для цього потрібно провести чимало часу в інтернеті, щоб переглянути схожі варіанти та визначити ціну. Зважаючи на це функція оцінки вартості автомобіля в розробленому сервісі, допоможе користувачам з оцінкою, а також зекономить їм час.

1.3 Застосування машинного навчання в бізнесі

В еру великих даних цей тип штучного інтелекту користується великим попитом. Машинне навчання можна використовувати в багатьох сферах для оптимізації і робочих процесів, від маркетингу та реклами до підтримки клієнтів і дослідження продуктів. Практичні застосування машинного навчання забезпечують бізнес-результати, які можуть істотно вплинути на прибутки компанії. Нові технології в цій галузі швидко розвиваються та розширюють застосування ML до майже безмежних можливостей. Галузі, які залежать від величезної кількості даних і потребують системи для їх ефективного й точного аналізу, прийняли ML як найкращий спосіб створювати моделі, розробляти стратегії та планувати.

Рекомендації, надані такими популярними потоковими платформами, як Spotify і Netflix, базуються на алгоритмах машинного навчання. Аналізуючи пісні, які ми слухали, або шоу, які ми дивилися, разом із масою даних про інші пісні, шоу та споживчі звички, ці алгоритми визначають і пропонують додатковий вміст, який може нам сподобатися. Численні варіанти використання машинного навчання показують, наскільки корисною ця технологія може бути для будь-якого бізнесу. Незалежно від

того, де і як воно використовується, компанії описують переваги машинного навчання в термінах експоненціальних здобутків і покращень.

Висновки до розділу

У даному розділі було підкреслено, популярність та її причини, пригону автомобілів з-за кордону. Наведено приклади необхідності розробки функціоналу для порівняння автомобілів в Україні та Європі. Визначили коли людині необхідно оцінити вартість автомобіля та актуальність цієї потреби. Також розібрались з позитивними сторонами застосування машинного навчання в бізнесі.

РОЗДІЛ 2. ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ

2.1 Реалізація збору даних для оцінки вартості автомобіля

Для оцінки вартості автомобіля потрібна вибірка даних з ціною автомобіля при певних параметрах. Після аналізу автомобільних торгових площадок було прийнято рішення обрати загальні параметри такі як модель, пробіг, тип палива, тип коробки передач, рік та ємність двигуна.

При формуванні технічних завдань було зрозуміло, що вручну зібрати потрібну кількість даних не получится. Для найбільш точного передбачення результату потрібно тренувати модель на основі великої та чіткої вибірки. За допомогою RPA платформи UiPath було створено мануального парсинг робота який переходив на сайт в певну категорію автомобілів та відкриваючи сторінки копіював html, відшукував дані автомобілів, та валідував (рис 2.1). Валідацію проходили ті дані які містили чітку інформацію по усіх параметрах, та записувались в excel таблицю з готовими даними (рис.2.2). Дані які не проходили валідацію записувались в excel таблицю для подальшого ручного уточнення. (рис. 2.3)

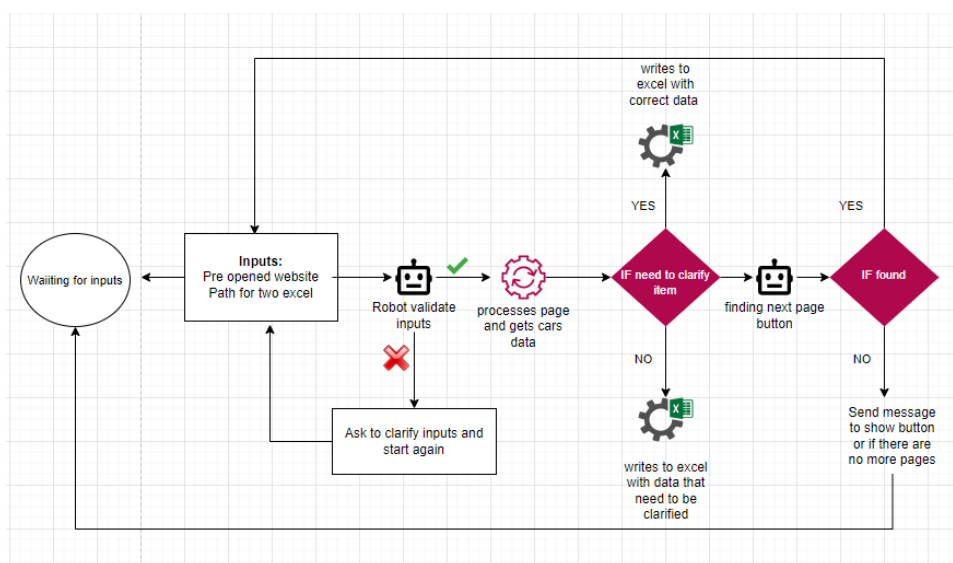


Рис.2.1 Узагальнена діграма роботи процесу

model	price	mileage	fueltype	gearbox	year	engine_capacity
BMW 320	733320	75000	gasoline	Machine	2017	2
BMW 320	297402	212000	diesel	Manual/Mechanics	2006	2
BMW 320	814759	65000	gasoline	Machine	2018	2
BMW 320	362586	221000	gasoline	Manual/Mechanics	2005	2
BMW 320	277032	195000	gasoline	Manual/Mechanics	2005	2

Рис.2.2 Приклад валідних даних вибірки

A	B	C	D	E	F	G
model	price	mileage	fueltype	gearbox	year	engine_capacity
BMW 320	209 811	320000	diesel	Machine	2001	
BMW 320	305 550	180000	diesel	Manual/Mechanics	2003	
BMW 320	203 700	376000	diesel	Machine	2002	
BMW 320	252 588	360000	diesel	Manual/Mechanics	2005	
BMW 320	256 662	198000	gasoline	Machine	2003	
BMW 320	138 516	98000	gasoline	Machine	2015	
BMW 320	305 550	160000	gasoline	Manual/Mechanics	2006	
BMW 320	158 886	350000	diesel	Manual/Mechanics	2001	
BMW 320	175 182	220000	gasoline	Manual/Mechanics	2004	
BMW 320	105 924	320000	gasoline	Manual/Mechanics	1992	
BMW 320	317 772	344000		Machine	2007	2
BMW 320	366 660	246000		Manual/Mechanics	2008	2
BMW 320	236 292	180000		Manual/Mechanics	2003	2
BMW 320		430000	diesel	Manual/Mechanics	2001	2

Рис.2.3 Приклад даних для ручної валідації

2.2 Опис даних для порівняння автомобілів

Для порівняння автомобілів було написано парсер який отримує дані з сторінки з автомобілями, та у вигляді Json передає на фронтенд сторону застосунку де відбувається візуалізація цих даних.(рис.2.1)

```

5,
{
  "name": "BMW 320",
  "img": "https://cdn4.riastatic.com/photosnew/auto/photo/bmw_320__469243009bx.jpg",
  "price": "6800",
  "link": "https://auto.ria.com/uk/auto_bmw_320_33418836.html",
  "shortdesc": "",
  "kilometrs": " 245 тис. км ",
  "typeoffuel": " Дизель, 2 л. "
},
5

```

Рис.2.4 Приклад отриманих даних після парсингу

2.3 Розмитнення автомобіля

В Україні скасували так зване “нульове розмитнення” після 1 липня 2022 року,тому тепер при ввезенні автомобілів громадяни повинні сплачувати податки.

З квітня по червень 2022 року в Україні діяло нульове розмитнення на більшість видів транспорту. Проте збір в Пенсійний фонд та оплати сертифікації залишилась. Це рішення було прийнято для того, щоб поповнити український авторинок дешевими автомобілями, адже після початку повномаштабної війни багато автівок громадян було знищено.

Такий порядок був скасований 1 липня 2022 року. Попри наявність нових зареєстрованих у парламенті законопроектів щодо нового порядку розмитнення автомобілів прийнятий не був. У тому числі довгостроковий проект який розроблявся Кабінетом міністрів, мета якого впровадити розмитнення через «Дію», але і цей проект досі не затверджений парламентом. Тому порядок розмитнення автомобілей залишився таким яким був до війни.

Розмитнення автомобілів проводиться за формулою:

Сума розмитнення = Ввізне мито + ПДВ + Акциз.

Також після сплати обов'язкових митних платежів власнику автівки потрібно пройти сертифікацію тільки після якої він зможе зареєструвати автомобіль в Сервісному центрі МВС де йому нарахують збір в Пенсійний фонд який дорівнює 3-5% від вартості автомобіля.

За допомогою розробленого сервісу громадяни зможуть розрахувати суму цих платежів одразу під час перегляду та порівняння автомобілей (рис 2.3.1). Також додатково було додано функцію як “Калькулятор розмитнення”(рис. 2.3.2)

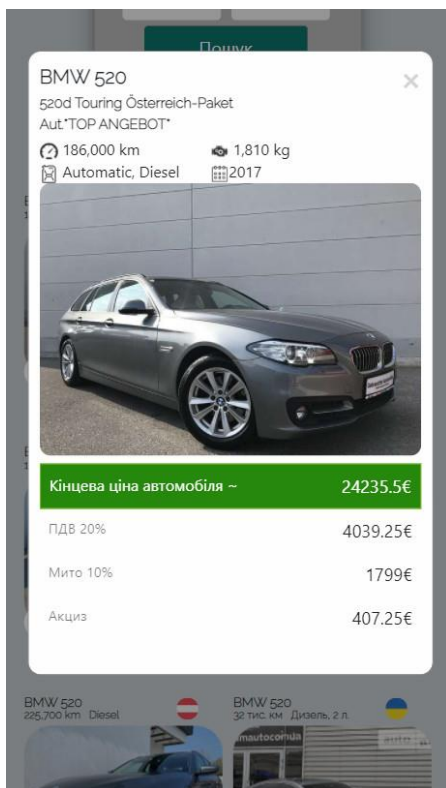


Рис. 2.3.1 Вікно результатів

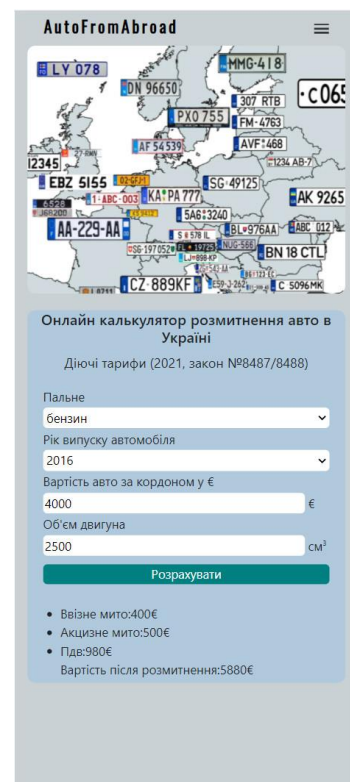


Рис. 2.3.2 Калькулятор розмитнення

Висновки до розділу

У розділі було розглянуто основні джерела та підхід для отримання даних. Було проведено аналіз та перевірку подрядку проведення розмитнення автомобіля та загальні види податків.

Для парсингу інформації про автомобілі було вибрано найбільші торгові площадки України та Європи.

РОЗДІЛ 3.МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ

3.1 Машинне навчання

Машинне навчання – це підгалузь штучного інтелекту, яка широко визначається як здатність машини імітувати інтелектуальну поведінку людини. Системи штучного інтелекту використовуються для виконання складних завдань у спосіб, подібний до того, як люди вирішують проблеми.

Мета штучного інтелекту полягає в тому, щоб створити комп'ютерні моделі, які демонструють «розумну поведінку», як люди, за словами Бориса Каца, головного наукового співробітника та керівника InfoLab Group в CSAIL. Це означає машини, які можуть розпізнавати візуальну сцену, розуміти текст, написаний природною мовою, або виконувати дію у фізичному світі.

Машинне навчання є одним із способів використання ШІ. Він був визначений у 1950-х роках піонером штучного інтелекту Артуром Семюелом як «галузь дослідження, яка дає комп'ютерам здатність навчатися без явного програмування».

За словами Майкі Шульмана, викладача Массачусетського технологічного інституту Слоуна та керівника відділу машинного навчання в Kensho, який спеціалізується на штучному інтелекті для фінансів і розвідувальних спільнот США, це визначення вірне. Він порівняв традиційний спосіб програмування комп'ютерів або «програмне забезпечення 1.0» із випічкою, де рецепт передбачає точну кількість інгредієнтів і наказує пекарю змішувати протягом точного проміжку часу. Традиційне програмування так само вимагає створення докладних інструкцій, яких комп'ютер повинен виконувати.

Але в деяких випадках написання програми, яку слідує машині, займає багато часу або є неможливим, наприклад, навчити комп'ютер розпізнавати зображення різних людей. Хоча люди можуть легко виконати це завдання, важко вказати комп'ютеру, як це зробити. Машинне навчання використовує підхід, який дозволяє комп'ютерам навчитися самостійно програмувати через досвід.

Машинне навчання починається з даних – чисел, фотографій або тексту, наприклад банківських транзакцій, фотографій людей або навіть хлібобулочних виробів, записів про ремонт, даних часових рядів із датчиків або звітів про продажі.

Дані збираються та готуються для використання як навчальні дані або інформація, на основі якої навчатиметься модель машинного навчання. Чим більше даних, тим краще програма.

Звідти програмісти вибирають модель машинного навчання для використання, надають дані та дозволяють комп'ютерній моделі навчитися знаходити закономірності чи робити прогнози. Згодом програміст також може налаштувати модель, зокрема змінити її параметри, щоб підштовхнути її до більш точних результатів. (Веб-сайт дослідника Джанель Шейн AI Weirdness – це цікавий погляд на те, як навчаються алгоритми машинного навчання та як вони можуть помилятися – як це сталося, коли алгоритм спробував згенерувати рецепти та створив шоколадний пиріг із куркою та куркою.)

Деякі дані беруться з навчальних даних, які використовуються як дані оцінювання, які перевіряють, наскільки точна модель машинного навчання, коли їй показуються нові дані. Результатом є модель, яку можна використовувати в майбутньому з різними наборами даних.

Успішні алгоритми машинного навчання можуть робити різні речі, написав Малоун у нещодавньому дослідженні про штучний інтелект та майбутнє роботи, співавторами якого були професор Массачусетського технологічного інституту та директор CSAIL Даніела Рус і Роберт Лаубахер, заступник директора Центру колективного розуму Массачусетського технологічного інституту. .

«Функція системи машинного навчання може бути описовою, тобто система використовує дані, щоб пояснити, що сталося; прогнозний, тобто система використовує дані, щоб передбачити, що станеться; або наказовий, що означає, що система використовуватиме дані, щоб робити пропозиції щодо того, які дії вжити», – пишуть дослідники.

3.1.1 Підкатегорії машинного навчання

Існує три підкатегорії машинного навчання:

- **Контрольовані** моделі машинного навчання навчаються за допомогою позначених наборів даних, що дозволяє моделям навчатися та ставати точнішими з часом. Наприклад, алгоритм буде навчений із зображеннями

собак та інших речей, усі позначені людьми, і машина навчиться способам самостійно ідентифікувати зображення собак. Кероване машинне навчання є найпоширенішим типом, який використовується сьогодні.

- У **неконтрольованому** машинному навчанні програма шукає шаблони в немаркованих даних. Машинне навчання без нагляду може знаходити шаблони або тенденції, які люди явно не шукають. Наприклад, програма машинного навчання без контролю може переглядати дані онлайн-продажів і ідентифікувати різні типи клієнтів, які здійснюють покупки.
- Машинне навчання **підсиленням** навчає машини методом спроб і помилок виконувати найкращі дії, встановлюючи систему винагород. Навчання з **підсиленням** може навчити моделі грати в ігри або навчити автономні транспортні засоби керувати, повідомляючи машині, коли вона прийняла правильні рішення, що допомагає їй з часом дізнатися, які дії їй слід виконати.

3.2 Регресійний аналіз в машинному навчанні

Для розроблення системи оцінювання наближеної вартості автомобіля було обрано саме регресійний підхід машинного навчання, адже він максимально підходить для вирішення задач такого типу.

Регресійний аналіз – це статистичний метод для моделювання зв'язку між залежною (цільовою) і незалежною (прогнозною) змінними з однією або кількома незалежними змінними. Більш конкретно, регресійний аналіз допомагає нам зрозуміти, як змінюється значення залежної змінної відповідно до незалежної змінної, коли інші незалежні змінні залишаються фіксованими. Він передбачає постійні/дійсні значення, такі як температура, вік, зарплата, ціна тощо.

Регресія – це методика навчання під наглядом, яка допомагає знайти кореляцію між змінними та дає нам змогу передбачити безперервну вихідну змінну на основі однієї або кількох змінних предиктора. Він в основному використовується для передбачення, прогнозування, моделювання часових рядів і визначення причинно-наслідкового зв'язку між змінними.

У регресії ми будемо графік між змінними, який найкраще відповідає заданим точкам даних. Використовуючи цей графік, модель машинного навчання може робити прогнози щодо даних. Простими словами: «Регресія показує лінію або криву, яка проходить через усі точки даних на графіку цільового прогнозу таким чином, що вертикальна відстань між точками даних і лінією регресії є мінімальною». Відстань між точками даних і лінією вказує, чи зафіксувала модель міцний зв'язок чи ні.

Приклади регресійних задач :

- Прогнозування дощу за температурою та іншими факторами
- Визначення тенденцій ринку
- Прогнозування ДТП через необдумане водіння.

3.2.1 Види регресій

Існує ряд різних підходів, які використовуються в машинному навчанні для виконання регресії. Для досягнення регресії машинного навчання використовуються різні популярні алгоритми. Різні методиками можуть включати різну кількість незалежних змінних або обробляти різні типи даних. Різні типи регресійних моделей машинного навчання також можуть передбачати різне співвідношення між незалежними та залежними змінними. Наприклад, методи лінійної регресії припускають, що зв'язок є лінійним, тому не буде ефективним для наборів даних із нелінійними зв'язками.

Деякі з найпоширеніших методів регресії в машинному навчанні можна згрупувати в такі типи регресійного аналізу:

- Проста лінійна регресія
- Множинна лінійна регресія
- Логістична регресія

3.3 Інтелектуальна автоматизація

Для створення вибірки даних, на основі якої проводиться машинне в розробленому сервісі, було прийнято рішення створити парсинг робота який буде наповнювати вибірку даними. Це зумовлено тим, що для найбільш точного передбачення результату потрібно тренувати модель на основі великої та чіткої

вибірки. Парсинг робот створювався на основі практики інтелектуальної автоматизації.

Інтелектуальна автоматизація — це практика автоматизації бізнес-процесів, які вимагають прийняття рішень, за допомогою передових технологій, таких як штучний інтелект, машинне навчання та RPA, для підвищення ефективності роботи.

Інтелектуальна автоматизація поєднує технології когнітивного та штучного інтелекту, включаючи обробку природної мови та машинне навчання з автоматизацією, щоб створювати інтелектуальні процеси та робочі процеси, які навчаються та адаптуються по ходу.

Ці системи можуть обробляти напівструктуровані дані (замовлення на купівлю, рахунки-фактури тощо), а також неструктуровані дані (відео, електронні листи, голосові повідомлення тощо), і вони можуть навчатися, відстежуючи поведінку людей.

Програми ІА охоплюють від збору даних, аналізу та прийняття рішень до управління автономними автомобілями та складними роботами.

Висновки до розділу

В даному розділі було розглянуто підходи машинного навчання та вибрано регресійну модель для розробленого сервісу. Також підкреслено необхідність використання інтелектуальної автоматизації для створення вибірки з даними. Загалом описано можливості інтелектуальних та математичних систем при розробці системи підтримки прийняття рішень.

РОЗДІЛ 4. ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ

4.1. Використані програмні середовища

Інтегроване середовище розробки (IDE) – це програмне забезпечення для створення програм, яке об'єднує загальні інструменти розробника в єдиний графічний інтерфейс користувача (GUI). IDE зазвичай складається з:

- Редактор вихідного коду: текстовий редактор, який може допомогти у написанні програмного коду за допомогою таких функцій, як підсвічування синтаксису за допомогою візуальних підказок, забезпечення автозавершення для певної мови та перевірка наявності помилок під час написання коду.
- Автоматизація локальної збірки: утиліти, які автоматизують прості, повторювані завдання як частину створення локальної збірки програмного забезпечення для використання розробником, як-от компіляція вихідного коду комп'ютера у двійковий код, пакування двійкового коду та виконання автоматизованих тестів.
- Налаштовувач: програма для тестування інших програм, яка може графічно відображати місце помилки в оригінальному коді.

4.1.1 Visual Studio 2019

Microsoft Visual Studio - це інтегроване середовище розробки (IDE), яке використовується для розробки додатків консолі та графічного інтерфейсу користувача разом із додатками Windows Forms, додатками ASP.NET, веб-сайтами, веб-додатками та веб-службами з використанням власного коду та керованого коду. Microsoft Visual Studio підтримує (різною мірою) різні мови програмування, якщо існує служба для певної мови. З коробки мовні пакети включають Visual C++, Visual Basic і Visual C#.

Ця технологія не містить бази даних. Однак ця технологія має можливість імпортувати та експортувати дані з різних типів баз даних, включаючи базу даних Oracle і сервер Microsoft Structured Query Language (SQL).

Microsoft Visual Studio доступна у версіях Community, Express, Professional, Premium, Ultimate, Enterprise і Test Professional. Кожне з цих видань надає однакові основні функції. Це оцінювання охоплює всі випуски Microsoft Visual Studio.

Microsoft Visual Studio доступна у версіях Community, Express, Professional, Premium, Ultimate, Enterprise і Test Professional. Кожне з цих видань надає однакові основні функції. Це оцінювання охоплює всі випуски Microsoft Visual Studio.

4.1.2 Visual Studio Code

Visual Studio Code, також званий VS Code, – це редактор вихідного коду, створений Microsoft для Windows, Linux і macOS. Функції включають підтримку налагодження, виділення синтаксису, інтелектуальне завершення коду, фрагменти, рефакторинг коду та вбудований Git. Користувачі можуть змінювати тему, комбінації клавіш, параметри та встановлювати розширення, які додають додаткові функції.

4.1.3 UiPath Studio

UiPath Studio — це IDE (інтегроване середовище розробки), яке допомагає проектувати, розробляти, налагоджувати та публікувати процес або бібліотеку. UiPath Studio — це розширене програмне забезпечення для автоматизації, яке надає всім, від бізнес-користувачів до досвідчених розробників RPA, правильну систему автоматизації для створення чудових програмних роботів, а організаціям — правильні інструменти керування для керування ними всіма.

4.2 Опис трирівневої архітектури

Трирівнева архітектура – це усталена архітектура програмного додатка, яка організовує програми на три логічні та фізичні обчислювальні рівні:

- рівень презентації або інтерфейс користувача;
- Рівень програми, на якому обробляються дані;
- Рівень даних, де зберігаються й керуються дані, пов'язані з програмою.

Головна перевага трирівневої архітектури полягає в тому, що кожен рівень працює на власній інфраструктурі, кожен рівень може одночасно розроблятися окремою командою розробників і за потреби оновлюватися або масштабуватися, не впливаючи на інші рівні.

Протягом десятиліть трирівнева архітектура була переважною архітектурою для програм клієнт-сервер. Сьогодні більшість трирівневих

додатків є цілями для модернізації з використанням хмарних технологій, таких як контейнери та мікросервіси, а також для міграції в хмару.

Рівень презентації (PL) є найвищим рівнем 3-рівневої архітектури, і його головна роль полягає в тому, щоб відображати результати для користувача, або, інакше кажучи, представляти дані, які ми отримуємо від рівня доступу до бізнесу та пропонуємо результати передньому користувачеві .

Рівень бізнес логіки (BLL) взаємодіє з рівнем доступу до даних і рівнем презентації для обробки дій, які призводять до логічних рішень та оцінок. Основним завданням цього шару є обробка даних між іншими шарами.

Рівень доступу до даних (DAL). Основною функцією цього рівня є доступ і зберігання даних з бази даних, а процес передачі даних до рівня доступу до бізнесу надходить на рівень презентації за запитом користувача.

4.3 Додаткові інструменти та фреймворки

4.3.1 ASP.Net CORE

ASP.NET Core - це консольна програма, яка читає та записує інформацію в порт. Платформа .NET Core надає базову модель консольної програми, яку можна запускати між платформами за допомогою інтерфейсу командного рядка. Додавання бібліотеки веб-сервера перетворює її на веб-програму ASP.NET Core.

Веб інтерфейс користувача ASP.NET Core - це повна структура інтерфейсу користувача. Існує три загальні підходи до створення сучасного веб-інтерфейсу за допомогою ASP.NET Core:

- **Web UI**

ASP.NET Core - це повна структура інтерфейсу користувача. Існує три загальні підходи до створення сучасного веб-інтерфейсу користувача за допомогою ASP.NET Core

- **Web API**

ASP.NET Core підтримує створення служб RESTful, також відомих як веб-API, за допомогою C#. Для обробки запитів Web-API використовує контролери – класи, які походять від ControllerBase.

- **Програми в реальному часі**

ASP.NET Core SignalR - це бібліотека з відкритим вихідним кодом, яка спрощує додавання веб-функціональності в режимі реального часу до програм. Веб-функції в режимі реального часу дозволяють коду на стороні сервера миттєво передавати вміст клієнтам (ігри, соціальні мережі, голосування, програми для спільної роботи, карти тощо).

- **Віддалений виклик процедури (RPC)**

gRPC, фреймворк Remote Procedure Call з відкритим кодом, можна розмістити на ASP.NET Core. Ідея RPC полягає в тому, що комп'ютерна програма може викликати та виконувати процедуру (підпрограму або службу) у віддаленій системі так само, як вона викликала б локальну підпрограму, але деталі мережевого зв'язку приховані від користувача.

4.3.2 Web API

У комп'ютерному програмуванні інтерфейс прикладного програмування (API) - це набір визначень підпрограм, протоколів та інструментів для створення програмного забезпечення та додатків.

Простіше кажучи, API – це свого роду інтерфейс, який має набір функцій, які дозволяють програмістам отримати доступ до певних функцій або даних програми, операційної системи чи інших служб.

Веб-API, як випливає з назви, є API через Інтернет, доступ до якого можна отримати за допомогою протоколу HTTP. Це концепція, а не технологія. Ми можемо створювати веб-API, використовуючи різні технології, такі як Java, .NET тощо. Наприклад, API REST Twitter надають програмний доступ для читання та запису даних, за допомогою яких ми можемо інтегрувати можливості Twitter у нашу власну програму.

ASP.NET Web API є розширюваною структурою для створення служб на основі HTTP, до яких можна отримати доступ у різних програмах на різних платформах, таких як веб, Windows, мобільні тощо. Він працює більш-менш так само, як веб-додаток ASP.NET MVC, за винятком що він надсилає дані як відповідь замість перегляду html. Це як веб-сервіс або служба WCF, але винятком є те, що він підтримує лише протокол HTTP.

Характеристики веб-API ASP.NET:

- ASP.NET Web API є ідеальною платформою для створення служб RESTful.
- ASP.NET Web API побудований на основі ASP.NET і підтримує конвеєр запитів/відповідей ASP.NET
- ASP.NET Web API зіставляє HTTP дієслова з іменами методів.
- ASP.NET Web API підтримує різні формати даних відповідей. Вбудована підтримка форматів JSON, XML, BSON.
- Веб-API ASP.NET може бути розміщено на IIS, на власному хості або іншому веб-сервері, який підтримує .NET 4.0+.
- Платформа ASP.NET Web API включає новий HttpClient для зв'язку із сервером Web API. HttpClient можна використовувати на стороні сервера ASP.MVC, додатку Windows Form, консольному додатку або інших програмах.

4.3.3 ML.Net

Для реалізації прогнозування ціни автомобіля було використано та інтегровано бібліотеку ML.Net адже відповідно до поставленої задачі тут була можливість застосувати алгоритми для тренування регресивні моделі, а також бібліотека сама визначала який алгоритм, а то і кілька алгоритмів, використовувати відповідно до величини та складності вибірки.[3]

Доступні алгоритми:[2]

- LbfgsPoissonRegressionTrainer
- LightGbmRegressionTrainer
- SdcaRegressionTrainer
- OlsTrainer
- OnlineGradientDescentTrainer
- FastTreeRegressionTrainer
- FastTreeTweedieTrainer
- FastForestRegressionTrainer
- GamRegressionTrainer

ML.NET - це кросплатформна платформа машинного навчання (ML) з відкритим кодом для .NET.

ML.NET дозволяє розробникам легко створювати, навчати, розгортати та використовувати спеціальні моделі у своїх програмах .NET, не вимагаючи попереднього досвіду розробки моделей машинного навчання чи досвіду роботи з іншими мовами програмування, такими як Python або R. Фреймворк забезпечує завантаження даних із файлів і баз даних, забезпечує перетворення даних і включає багато алгоритмів ML.

За допомогою ML.NET ви можете навчати моделі для різноманітних сценаріїв, таких як класифікація, прогнозування та виявлення аномалій.

4.3.4 Angular

Angular є частиною екосистеми JavaScript і одним із найпопулярніших інструментів розробки програмного забезпечення сьогодні. Він був представлений Google у 2009 році та отримав теплі рекомендації від спільноти розробників. Згідно з опитуванням StackOverflow 2022 року, 23 відсотки розробників програмного забезпечення застосовують фреймворк для створення інтерфейсів користувача. Отже, давайте поговоримо про основні функції цих інструментів і дослідимо їхні переваги та недоліки з точки зору інженерного інтерфейсу.

Angular, що підтримується Google, - це платформа розробки програмного забезпечення з відкритим вихідним кодом на основі JS для створення інтерфейсів користувача (front-end). Його історія починається з 2009 року, коли Міско Хевері та Адам Абронс, інженери Google, розробили фреймворк, відому як AngularJS, і офіційно випустили його в 2010 році.

TypeScript замість JavaScript. На відміну від свого попередника на основі JS, Angular використовує TypeScript, суворий синтаксичний набір JavaScript, розроблений Microsoft для розробки великих програм.

Компоненти як основні будівельні блоки. AngularJS дозволяє розробникам створювати багаторазові фрагменти коду за допомогою директив - розширень HTML, які призначають певну поведінку елементу. Angular успадкував усі стандартні директиви, але представив багаторазові, самодостатні компоненти як основні будівельні блоки. У двох словах, компонент — це директива, пов'язана з шаблоном HTML, яка визначає, що відображається на сторінці.

Краща продуктивність. І AngularJS, і Angular значно скорочують час розробки. Однак завдяки компонентній архітектурі та більш ефективному зв'язуванню даних програми Angular можуть працювати в п'ять разів швидше, ніж AngularJS.

Зручність для мобільних пристроїв. AngularJS не розроблявся з урахуванням мобільних браузерів. Під час розробки Angular Google врахував цю прогалину, тому новий фреймворк отримав підтримку мобільного Інтернету та рідних мобільних додатків.

4.3.5 HTML AGILITY PACK

Це новий аналізатор HTML, який створює DOM для читання/запису та підтримує звичайний XPATH або XSLT. Це бібліотека коду .NET, яка дозволяє розбирати HTML-файл. Синтаксичний аналізатор дуже толерантний до неправильного HTML в «реальному світі». Об'єктна модель дуже схожа на те, що пропонує System.Xml, але для документів HTML (або потоків).

4.3.6 SQL Server Management Studio (SSMS)

SQL Server Management Studio (SSMS) – це інтегроване середовище для керування будь-якою інфраструктурою SQL, від SQL Server до бази даних SQL Azure. SSMS надає інструменти для налаштування, моніторингу та адміністрування екземплярів SQL Server і баз даних. Використовуйте SSMS для розгортання, моніторингу та оновлення компонентів рівня даних, які використовуються вашими програмами, а також створення запитів і сценаріїв.

4.3.7 Postman

Postman – це програма, яка використовується для тестування API. Це HTTP-клієнт, який тестує HTTP-запити, використовуючи графічний інтерфейс користувача, за допомогою якого ми отримуємо різні типи відповідей, які згодом необхідно перевірити. Postman дає можливість групувати різні запити. Ця функція відома як «колекції» і допомагає організувати тести.

Postman також дозволяє нам створювати різні середовища за допомогою генерації/використання змінних; наприклад, змінна URL-адреси, яка призначена для

різних тестових середовищ (dev-QA), що дозволяє нам виконувати тести в різних середовищах за допомогою наявних запитів.

4.4 Ключові моменти реалізації системи

Було створено пусте рішення у яке поступово було додано чотири проекти, один для реалізації бізнес рівня та інший рівня даних, третім було додано проект Web Api + Angular, а останнім було додано проект ML.NET (рис.4.4.1)

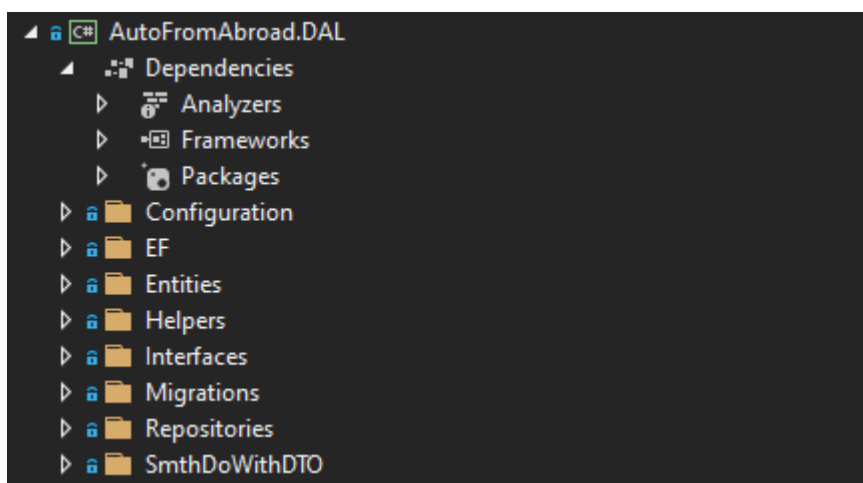


Рис.4.4.2 Структура рівня DAL

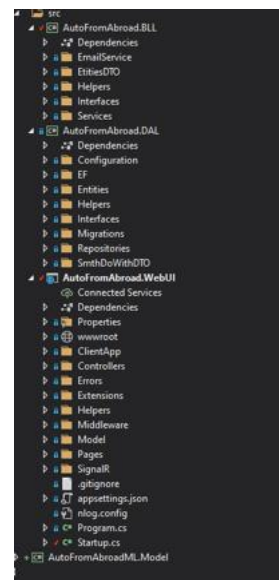


Рис.4.4.1 Структура проекту

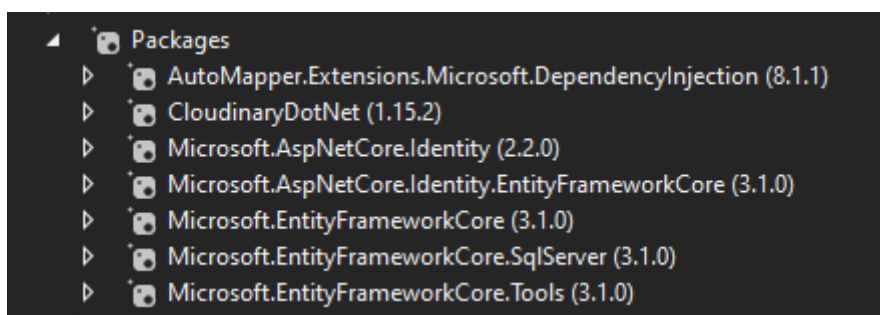


Рис.4.4.3. Підключені пакети рівня DAL

На рівні DAL описані основні моделі та класи для взаємодії з базою даних. Під час розробки використовувався підхід code first, тобто спершу було описано всі сутності та класи і на основі них була створена база даних з усіма таблицями та зв'язками. В процесі розробки, після додавання нових сутностей та моделей даних, виконувалась функція міграції. Функція міграції в EF Core забезпечує спосіб

поступового оновлення схеми бази даних, щоб підтримувати її синхронізацію з моделлю даних програми, зберігаючи при цьому наявні дані в базі даних.

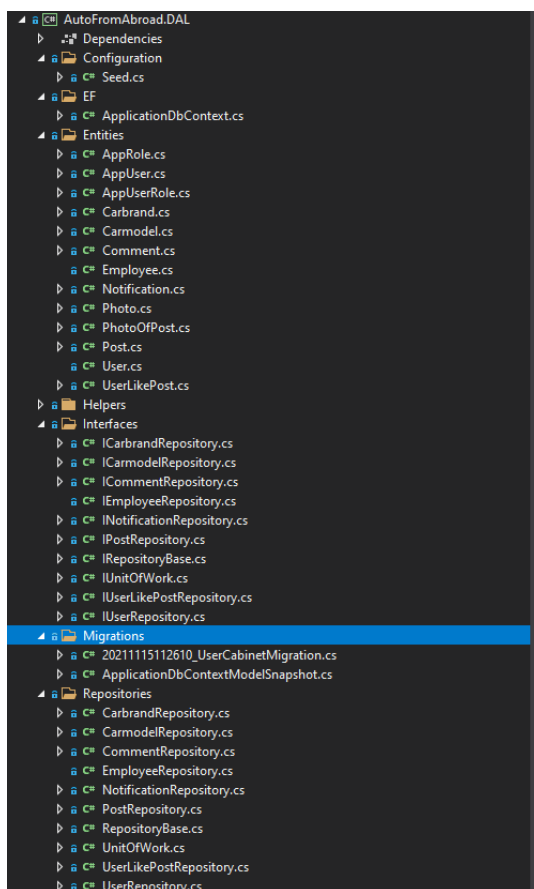


Рис.4.4.4 Структура рівня DAL

Було реалізовано паттерн Repository і Unit of Work ці паттерни призначені для створення рівня абстракції між рівнем доступу до даних і рівнем бізнес-логіки програми. Реалізація цих шаблонів може допомогти захистити вашу програму від змін у сховищі даних і може полегшити автоматичне модульне тестування.

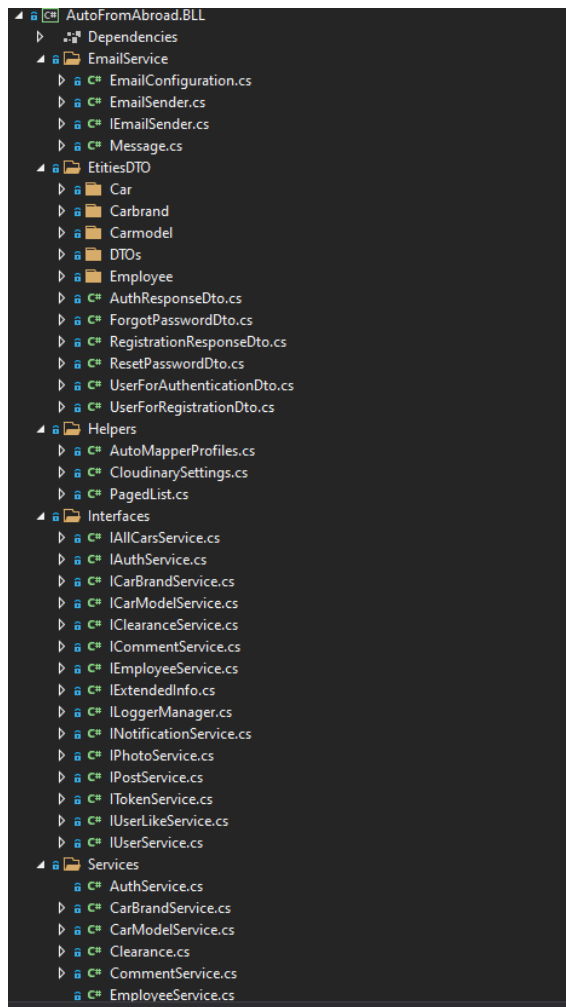


Рис.4.4.5 Структура рівня BLL

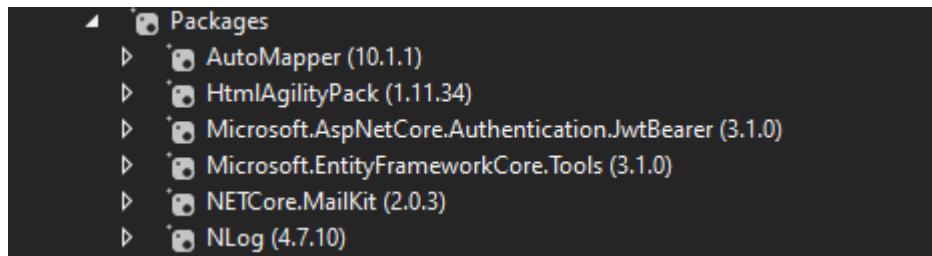


Рис.4.4.6 Пакети рівня BLL

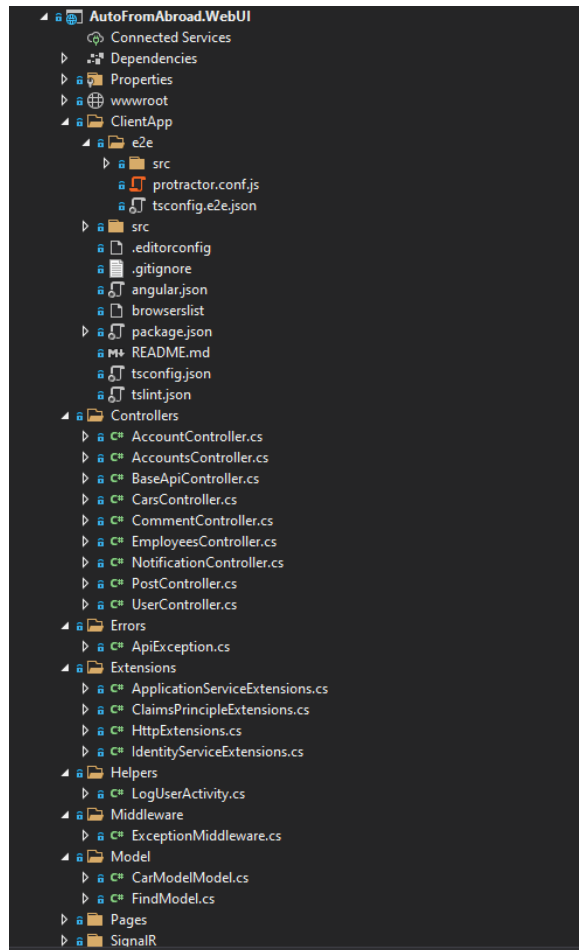


Рис.4.4.7 Структура ріння PL

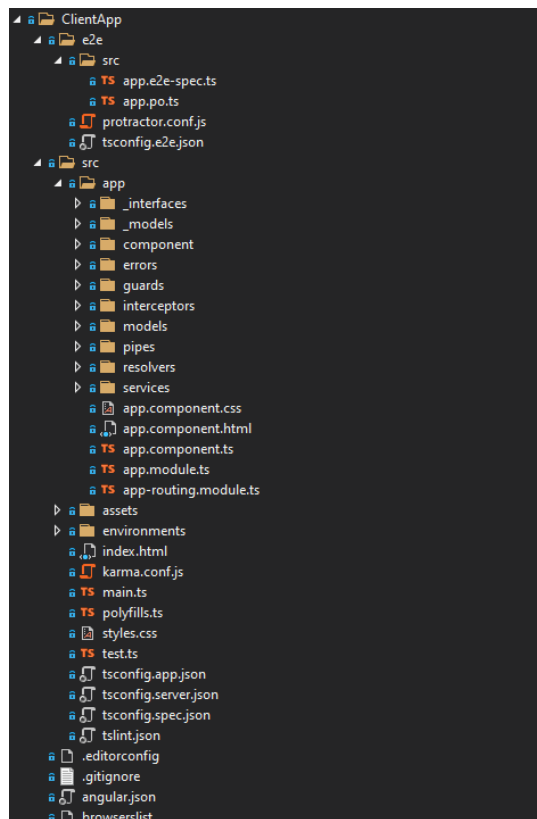


Рис.4.4.8 Структура Angular проекту

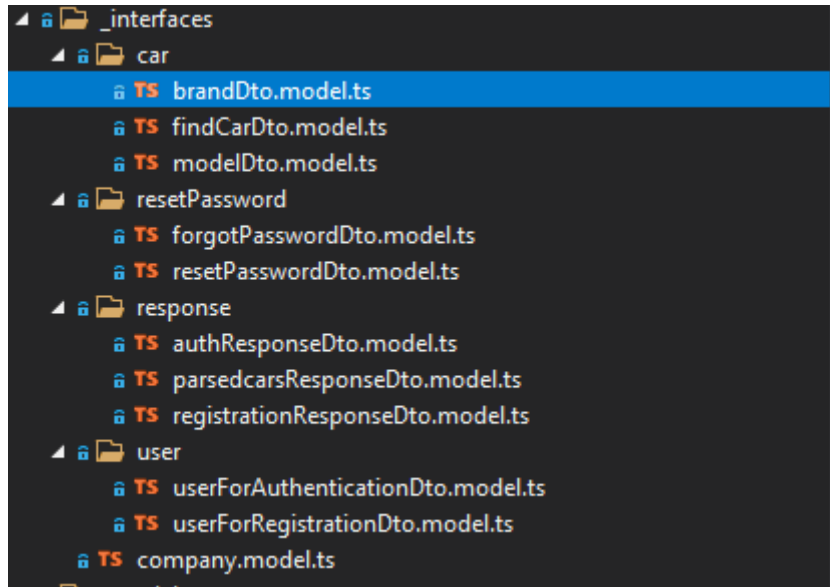


Рис.4.4.9 Інтерфейси ts

Інтерфейси в Typescript – це концепція лише під час розробки, вони не включені в остаточний JavaScript після процесу збірки. За допомогою ключового слова `implements`, ми використовуємо інтерфейс як тип даних. (Рис.4.4.10)

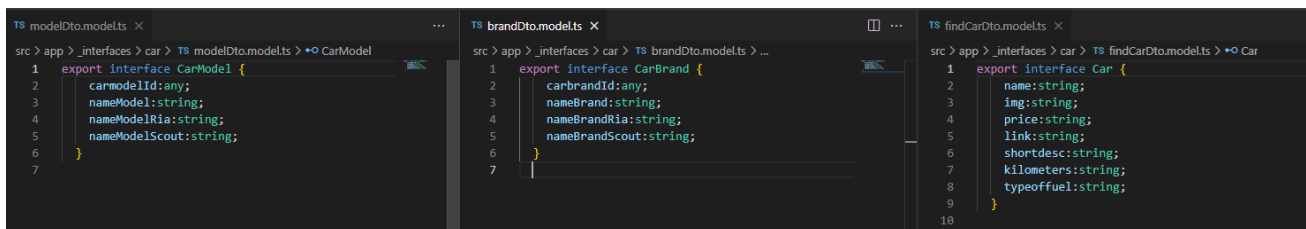


Рис.4.4.10 Приклад інтерфейсів

Компоненти – це будівельні блоки Angular-додатків. Компоненти включають TypeScript-клас, який має декоратор `@Component()`, HTML-шаблон та стилі. Компоненту пошукової панелі (Рис.4.4.10)

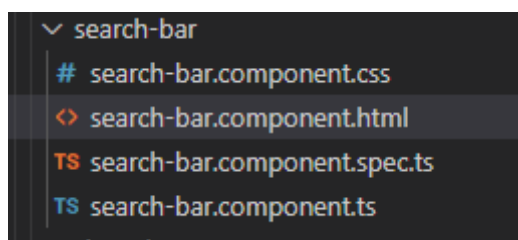


Рис.4.4.10 search-bar

```

src > app > search-bar > search-bar.component.html > div.search-sec > div.form-inline > div.cols
1 <div class="search-sec">
2   <div class="form-inline">
3     <div class="cols">
4       <select id="brand" class="selectpicker" title="Brand" (change)="getModels($event.target.value)" data-live-search="true" data-size="10">
5         <option *ngFor="let car of carBrands" [value]="car.carBrandId" [attr.data-ria]="car.nameBrandRia" [attr.data-scout]="car.nameBrandScout">{{car.nameBrand}}
6       </select>
7     </div>
8     <div class="cols">
9       <select id="model" class="selectpicker" title="Model" data-live-search="true" data-size="10">
10        <option *ngFor="let model of carModels" [value]="model.carModelId" [attr.data-modelria]="model.nameModelRia" [attr.data-modelscout]="model.nameModelScout">
11      </select>
12    </div>
13    <div class="cols id="country">
14      <select id="country" class="selectpicker" title="Country" data-live-search="true" data-size="10">
15        <option *ngFor="let country of Countries" [attr.data-id]="country.id">{{country.name}}</option>
16      </select>
17    </div>
18    <div class="cols">
19      <select id="year" class="selectpicker" title="Year" data-live-search="true" data-size="10">
20        <option *ngFor="let number of numbers">{{number}}</option>
21      </select>
22    </div>
23    <div class="cols">
24      <input id="search" class="btnsearch" (click)="findCars()" [disabled]="!searchBtn" value="Пошук" type="button" [ngStyle]="{'background-color': searchBtn ? '#8e
25    </div>
26  </div>

```

Рис.4.4.11 Частина search-bar.component.html

Машинне навчання було додане окремим проектом в архітектуру (Рис 4.4.12).

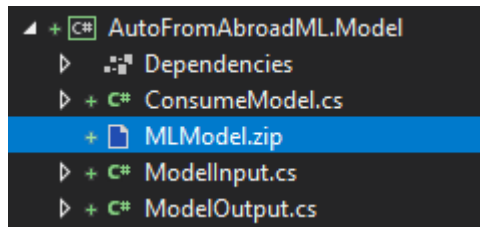


Рис.4.4.12 Структура ML проекту

Під час навчання моделі бібліотека підбрала найбільш ефективні алгоритми

Top 5 models explored						
	Trainer	RSquared	Absolute-loss	Squared-loss	RMS-loss	Duration #Iteration
1	OlsRegression	0.6637	12463.93	301074356.80	17351.49	0.6 1
2	OlsRegression	0.6637	12463.93	301074356.80	17351.49	0.6 2
3	LbfgsPoissonRegression	0.6628	12043.46	301932879.55	17376.22	0.6 3
4	SdcaRegression	0.6628	12495.70	301966262.52	17377.18	11.1 4
5	SdcaRegression	0.6625	12439.04	302153263.68	17382.56	0.8 5

Рис.4.4.13 Результати роботи алгоритмів навчання

На рисунку 4.4.14 представлений метод контролера який який повертає результат оцінки вартості автомобіля. Результат повертається у вигляді json об'єкта в якому відображається ціна у гривнях та доларх (рис. 4.4.15)

```

[HttpGet("GetPrediction")]
public IActionResult GetPricePrediction([FromQuery] string model, float engine_capacity, string Gearbox, float Mileage, string FuelType, float year)
{
    var input = new ModelInput() {
        Model = model,
        Engine_capacity = engine_capacity,
        Gearbox = Gearbox,
        Mileage = Mileage,
        Fueltype = FuelType,
        Year = year
    };
    ModelOutput result = ConsumeModel.Predict(input);
    var PredictedPrice = new PricePrediction()
    {
        UAH = result.Score.ToString(),
        USD = (result.Score / _iExchangeRateParser.UahToUsd()).ToString()
    };
    return Ok(PredictedPrice);
}

```

Рис.4.4.14 Метод контролера оцінки вартості

```

{
  "uah": "302142.56",
  "usd": "7432.7812"
}

```

Рис.4.4.15 Результат

Клас ParseCars реалізує інтерфейс IAllCarsService, тут реалізуються два основні методи які виконують парсинг веб-сторінок.

Парсинг сторінки відбувається за допомогою бібліотеки HtmlAgilityPack. Так як неможливо скачати всю інформацію про автомобілі з обох сайтів, було вирішено отримувати інформацію про заданий автомобіль одразу після введення користувачем даних. Після отримання необхідних даних ми формуємо url посилання яке відобразить необхідну інформацію, після цього ця веб сторінка завантажується і методами бібліотеки HtmlAgilityPack ми витягуємо усю необхідну для нас інформацію.

Парсинг автомобілів з сайту <https://auto.ria.com> реалізує метод ParseRio (рис.4.4.16).

```
public List<Car> ParseRio(string nameId, string modelId, string year)
```

Рис.4.4.16 Метод ParseRio

Метод ParseRio повинен повертати список об'єктів класу Car .

Спочатку ми відповідно до отриманих параметрів формуємо url посилання та зберігаємо його у змінній типу string з назвою url (рис. 4.4.17).

```
string url = "https://auto.ria.com/uk/search/?indexName=auto,order_auto,newauto_search&year[0].gte=" + year + "&year[0].lte=" + year + "&categories.main.id=1&brand.id[0]=" + nameId + "&model.id[0]=" + modelId + "&price.currency=1&abroad.not=0&custom.not=1&page=0&size=10";
```

Рис. 4.4.17 Формування url посилання

Наступним кроком є створення списку для подальшого заповнення також отримання веб сторінки за сформованим посилання (рис. 4.4.18).

```
List<Car> cars = new List<Car>();  
HtmlWeb webDoc = new HtmlWeb();  
HtmlDocument doc = webDoc.Load(url);
```

Рис. 4.4.18 Завантаження документа сторінки

Після того як ми отримали xml документ сторінки з даними нам необхідно витягнути усю необхідну для нас інформацію, для цього ми використовуємо елементи бібліотеки HtmlAgilityPack а також мову запитів XPath.

XPath - це мова запитів для отримання вузлів і також обчислення їх значень із вмісту XML-документа. Для того щоб знати у яких вузлах міститься необхідна інформація було досліджено XML документ сайту.

Було виявлено блок у якому під час загрузки сторінки зберігалась інформація про автомобілі та сформовано колекцію вузлів (рис. 4.4.19).

```
HtmlNodeCollection Id = doc.DocumentNode.SelectNodes("//div[@class='content-bar']");
```

Рис. 4.4.19 Колекція нод

Після цього потрібно пройти по усіх елементах колекції та витягнути за допомогою XPath інформацію, обробити її та додати у колекцію Cars (рис. 4.4.20).

```
21 Car car = new Car();
22
23 if (Id!=null)
24 {
25     foreach (HtmlNode o in Id)
26     {
27         HtmlNode i = o.SelectSingleNode("./span[@class='blue bold']");
28         if (i != null)
29         {
30             Regex regex = new Regex(@"(\w+\s+\w+)");
31             car.Name = regex.Match(i.InnerText).Value;
32
33         }
34         HtmlNode j = o.SelectSingleNode("./div[@class='price-ticket']");
35         if (j != null)
36         {
37             car.price = j.Attributes["data-main-price"].Value;
38
39         }
40         HtmlNode p = o.SelectSingleNode("./a[@class='m-link-ticket']");
41         if (p != null)
42         {
43             car.link = p.Attributes["href"].Value;
44         }
45         HtmlNode m = o.SelectSingleNode("./img[@class='outline m-auto']");
46         if (m != null)
47         {
48             car.Img = m.Attributes["src"].Value;
49         }
50         HtmlNode n = o.SelectSingleNode("./div[@class='content']/div[@class='definition-data']/p[@class='descriptions-ticket']/span");
51         if (n != null)
52         {
53             car.shortdesc = n.InnerText;
54         }
55         HtmlNode l = o.SelectSingleNode("./div[@class='content']/div[@class='definition-data']/ul[@class='unstyle characteristic']/li");
56         if (l != null)
57         {
58             car.kilometrs = l.InnerText;
59         }
60         HtmlNode b = o.SelectSingleNode("./div[@class='definition-data']/ul[@class='unstyle characteristic']/li[@class='item-char']");
61         if (b != null)
62         {
63             car.typeoffuel = b.InnerText;
64         }
65         cars.Add(car);
66         car = new Car();
67     }
68 }
69 return cars;
70
71 }
```

Рис. 4.4.20 Парсинг та заповнення колекції

Також для окремих стрічок ми використовуємо регулярні вирази. Регулярні вирази представляють з себе послідовність символів, які визначають шаблон пошуку. Зазвичай такі шаблони використовуються для редагування стрічок відповідно до побудованих шаблонів. Для створення та використання шаблону ми використовуємо клас `Regex` (рис. 4.4.21).

```
Regex regex = new Regex(@"(\w+\s+\w+)");
car.Name = regex.Match(i.InnerText).Value;
```

Рис. 4.4.21 Регулярний вираз для виділення перших двох слів стрічки

Метод ParseScout реалізований аналогічно методу ParseRia тільки з своїми XPath шляхами та регулярними виразами. Також якщо не вдалось отримати зображення, або його немає, тоді в колекцію записується посилання на зображення за замовчуванням.

```
[HttpGet("GetRia")]
0 references | 0 changes | 0 authors, 0 changes
public IActionResult ListRia([FromQuery] string brand, string model, string year)
{
    var getcarsRia = _allCarsService.ParseRia(brand, model, year);
    return Ok(getcarsRia);
}
[HttpGet("GetScout")]
0 references | 0 changes | 0 authors, 0 changes
public IActionResult ListScout([FromQuery] string brand, string model, string year, string country)
{
    var getcarsScout = _allCarsService.ParseScout(brand, model, year, country);
    return Ok(getcarsScout);
}
```

Рис. 4.4.22 Методи контролера парсингу автомобілів

```
export class FindCarsService {
    constructor(private http: HttpClient, private _envUrl: EnvironmentUrlService) { }

    getBrandList(): Observable<CarBrand[]> {
        return this.http.get<CarBrand[]>(this._envUrl.urlAddress + "/api/getcars/GetBrands");
    }
    getModellis(id: number) {
        return this.http.get<CarModel[]>(this._envUrl.urlAddress + "/api/getcars/GetModels?id=" + id);
    }
    findCarsRia(name:string, model:string, year:string){
        return this.http.get<Car[]>(this._envUrl.urlAddress + "/api/getcars/GetRia?brand="+name+"&model="+model+"&year="+
    }
    findCarsScout( name:string, model:string, year:string, country:string){
        return this.http.get<Car[]>(this._envUrl.urlAddress + "/api/getcars/GetScout?brand="+name+"&model="+model+"&year="+
    }
}
```

Рис. 4.4.23 Сервіс виклику http запитів на стороні фронтенду

4.5 Тестування та візуалізація

Було проведено ручне тестування веб-застосунку. На найфундаментальнішому рівні ручне тестування - це процес тестування програмного забезпечення, у якому тестові випадки - або конкретні оцінки функції, функції чи продуктивності - виконуються без використання будь-яких автоматизованих інструментів. Будь-яке відхилення від очікуваної поведінки або результату може вважатися дефектом. Було проведено наступні види тестування:

- Functional testing (функціональне тестування);
- Performance testing (тестування продуктивності);
- Stability testing (тестування стабільності);
- UI testing (тестування інтерфейсу користувача).

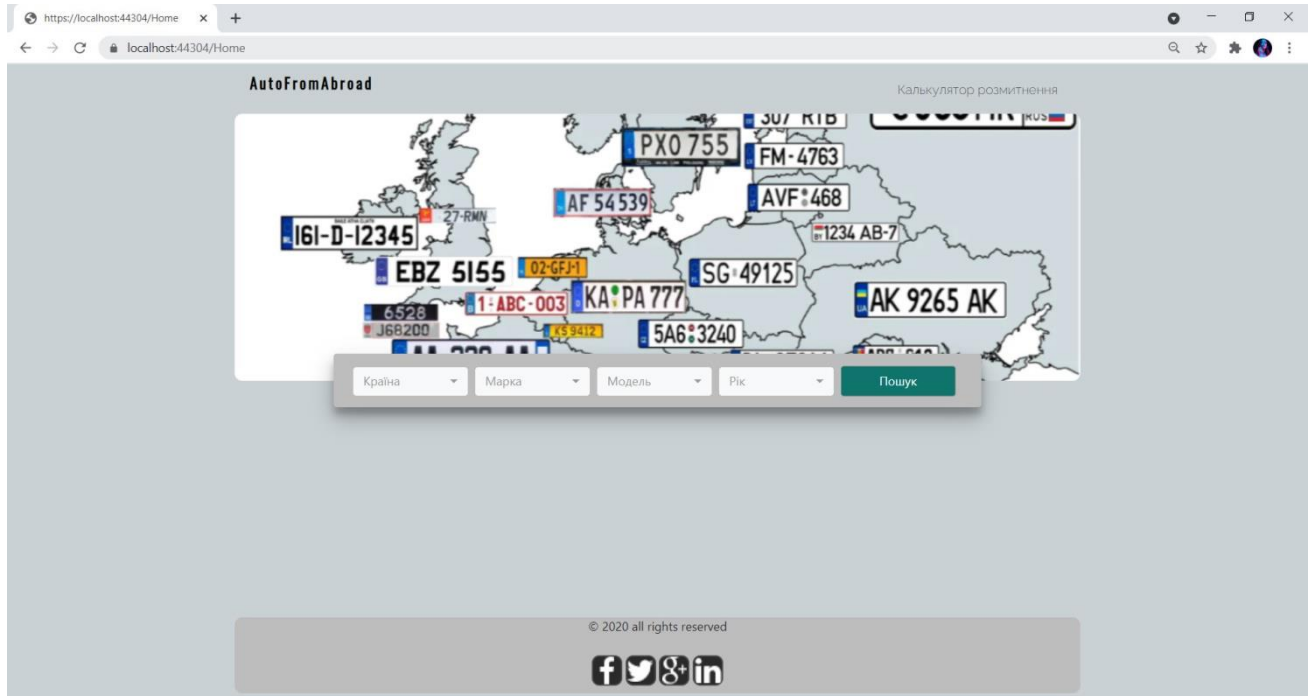


Рис. 4.5.1 Домашня сторінка

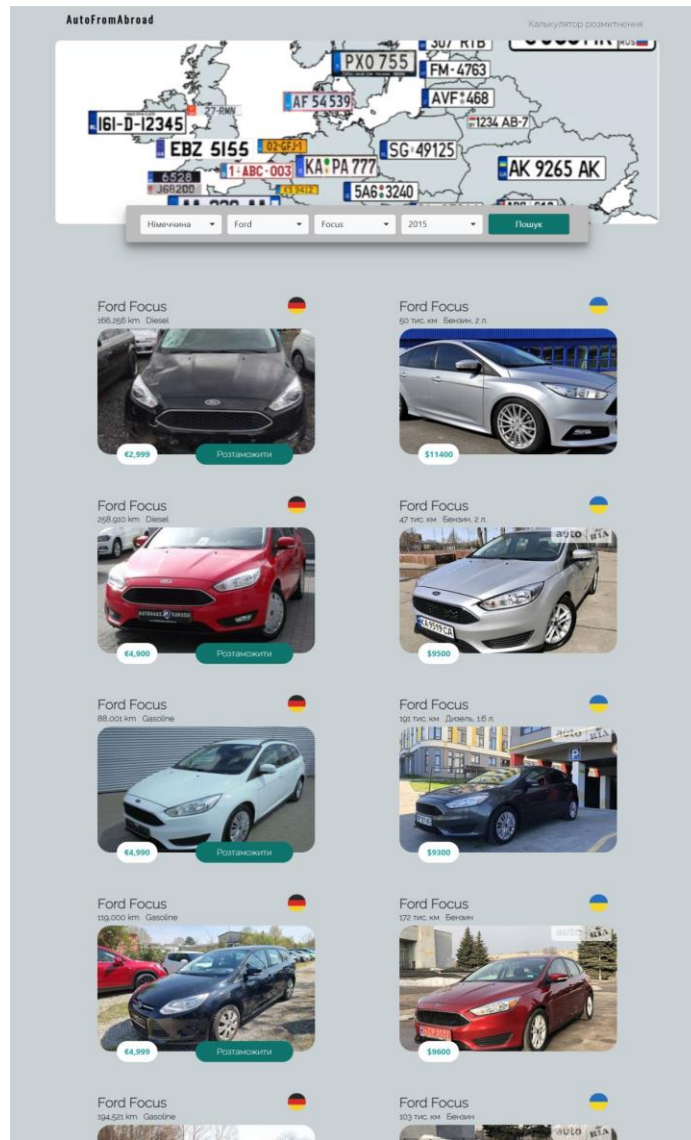


Рис. 4.5.2 Результат пошуку десктопна версія

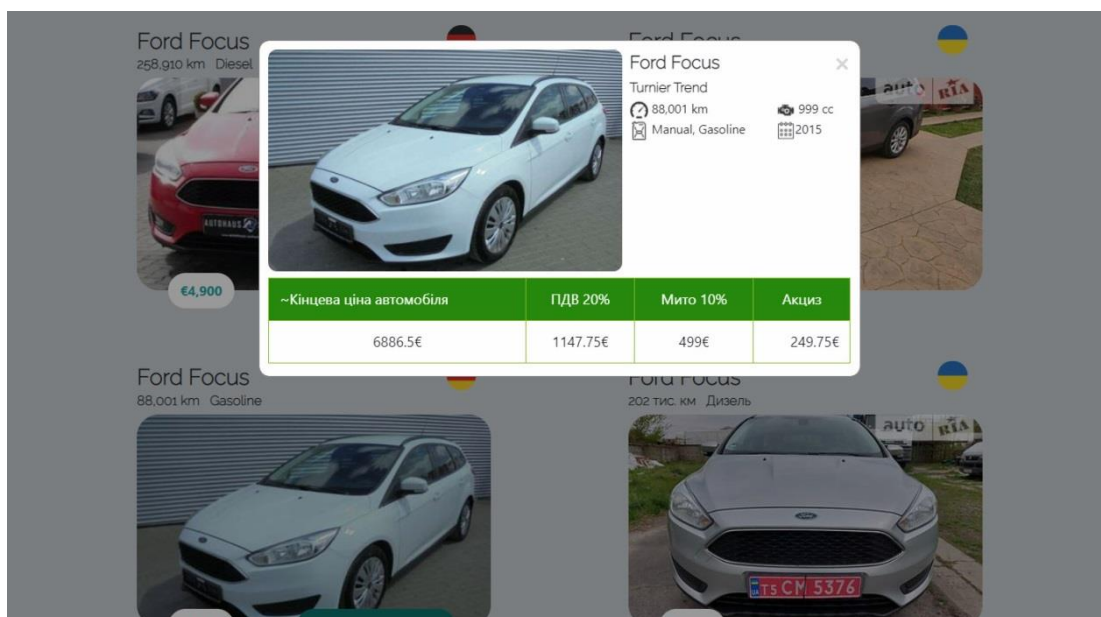


Рис. 4.5.3 Результат обчислення на основі автомобіля

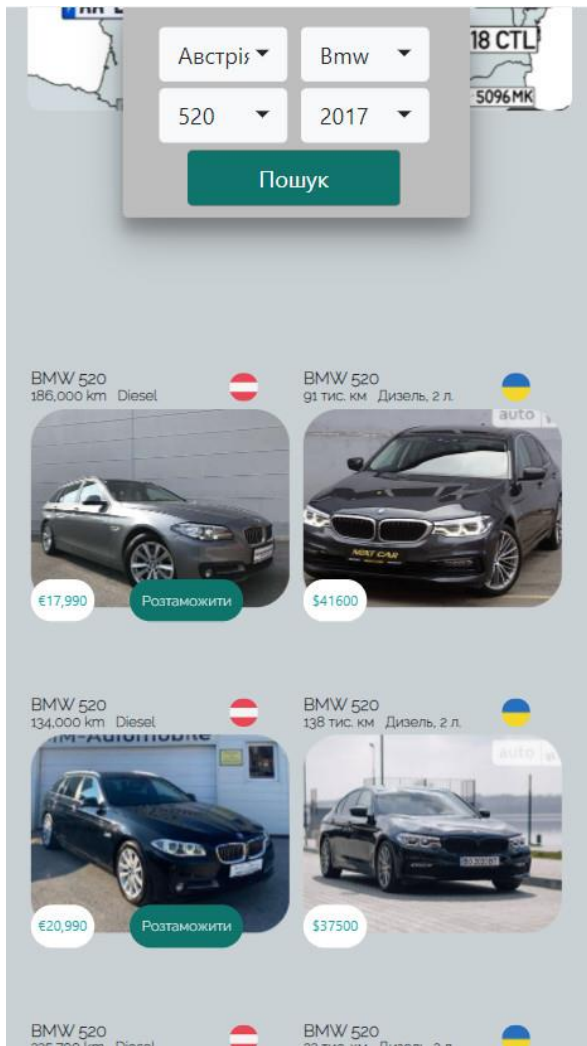


Рис. 4.5.4 Результат пошуку моб.

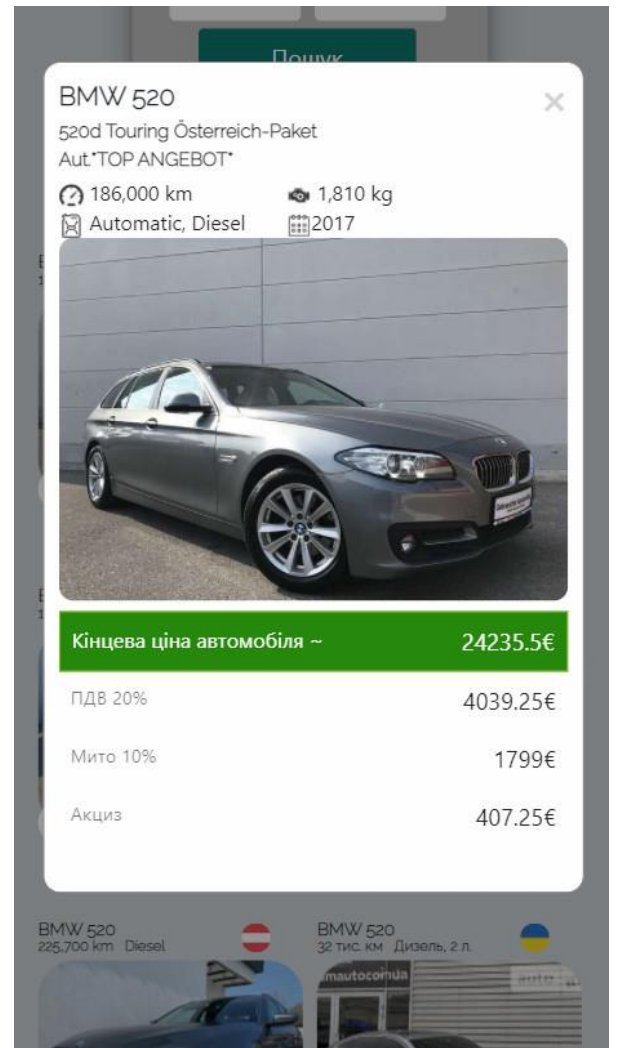


Рис. 4.5.5 Вікно результатів моб.

Передбачення ціни автомобіля

Марка
BMW 320

Пальне
бензин

Рік випуску автомобіля
2006

Коробка передач
Ручна

Пробіг у км
240000

Об'єм двигуна
2.2

Передбачити

- UAH: 302142
- USD: 7432

Рис. 4.5.6 Оцінка вартості автомобіля моб.

Висновки до розділу

В даному розділі описано використані засоби при розробці веб-застосунку. Описано задіяні фреймворки та підходи. Оглянуто програмний код ключових методів та архітектури застосунку. Також описано підходи тестування, та показано результати відпрацювання сервісів для кінцевого користувача.

РОЗДІЛ 5. РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ

5.1 Опис ідеї проекту

Попит на вживані автомобілі з Європи зростає з кожним роком. Це зумовлено багатьма факторами, один з найбільш вагомих - це ціна автомобіля, адже в Європі вона є значно нижчою ніж в Україні. Незважаючи навіть на процедуру розмитнення пригнаного автомобіля, кінцева його ціна буде нижчою ніж на авторинку України, але це стосується далеко не всіх автомобілів, тому перед громадянином, стоїть питання, купувати авто в Україні чи приганяти з Європи. Також ще одною не менш важливою проблемою є оцінка вартості автомобіля, адже відповідно до параметрів та побажань ціна сильно відрізняється. Тому щоб знати чи задуманий автомобіль впишеться у виділений на нього бюджет, потрібно оцінити цінову політику ринку, а також конкретного автомобіля.

У цьому і полягає ідея проекту – допомогти користувачам у вирішенні цих проблем, надати можливість в реальному часі порівняти вибраний автомобіль в Україні та іншій європейській країні, та одним натиском кнопки вирахувати розмитнення цього автомобіля і прийняти подальші рішення. Надати користувачам можливість, за допомогою сервісу, власноруч розглядати переваги купівлі автомобіля в Європі, або в Україні. Також за допомогою штучного навчання прогнозувати наближену вартість автомобіля відповідно до його характеристик.

Сервіс націлений не тільки на людей які хочуть купити автомобіль, а також він стане у пригоді тим хто планує продати власний автомобіль. Адже після внесення даних про свій автомобіль, за допомогою машинного навчання, на основі автомобілів які вже продаються, сервіс покаже наближену оцінку вартості заданого автомобіля.

Основними перевагами сервісу є унікальність, адже на ринку відсутні аналогічні сервіси які в режимі реального часу дозволяють отримати дані про автомобіль в Європі та Україні, а також без додаткових калькуляторів, розрахувати ціну розмитнення Європейського автомобіля. Також відсутні сервіси які надають можливість оцінки вартості автомобіля на українському ринку. Додатком до цього сервіс буде безкоштовним. Зрозумілий та простий користувацький інтерфейс, адаптований під більшість розширень екранів, з сучасною візуалізацією даних також

буде перевагою. Швидкість роботи веб-застосунку забезпечена дотриманням усіх SOLID принципів та сучасних технологій розробки.

5.2 Аналіз технологічних можливостей реалізації ідеї проекту

Проект реалізовано на основі трирівневої архітектури, ASP.NET Core 3.1 Web API на стороні бекенду та Angular на стороні фронтенду, а також ML.Net для машинного навчання.

5.3 Аналіз ринкових можливостей запуску стартап-проекту

Купівля та продаж автомобіля завжди актуальні. Незважаючи на ситуацію в країні та інші чинники, попит на автомобілі не опускається до нуля. Зважаючи на це розроблений сервіс завжди буде мати можливості для розвитку.

До чинників які можуть вплинути на зміну вектору розіску проекту, можна віднести можливість зміни порядку розмитнення автомобілів. Проте це спричинить тільки зміну в самому проекті, через заміни формул розмитнення, що не є критичним. Розглядаючи період нульового розмитнення, коли з квітня по червень 2022 року в Україні було скасовано розмитнення на більшість видів транспорту, не враховуючи збір в Пенсійний фонд та оплати сертифікації, можна з впевненістю сказати, що ця ситуація не вплинула б на проект, адже сервіс окрім самого обрахунку розмитнення надає можливість порівняти автомобілі та оцінити наближену вартість.

Після проведення аналізу, фактори що перешкоджають ринковому впровадженню проекту, або відсутні, або їхні ризики мінімальні.

5.3.1 Аналіз проектів конкурентів

Після дослідження ринку не було виявлено прямих конкурентів проекту, проте існують сервіси які містять в собі загальні калькулятори розмитнення на основі введення даних про автомобіль. До них можна віднести:

- <https://calc.customs.gov.ua>.

Державний сайт, який надає безкоштовне користування калькулятором розрахунку розмитнення авто, відповідно до чинного законодавства.

- <https://auto.ria.com/uk/rastamozhka-avto/calculator>.

Розділ сайту Авто Ріа, (найбільша українська інтернет площадка для продажу автомобілів) також містить безкоштовний функціонал для розрахунку

розмитнення не тільки легкового автомобіля, а також мото, вантажних та автобусів..

Існують також сервіси порівняння товарів, на різних торгових площах, окрім автомобілів:

- <https://ek.ua>.

E-Katalog - це сервіс пошуку товарів на різних торгових площах, а також порівняння їх цін. Сервіс надає можливість безкоштовно здійснювати пошук та порівняння, товарів, категоріями товарів. Варто зауважити що сайт користується великою популярністю.

- <https://hotline.ua>.

Hotline.ua - це схожий за функціональністю сайт до попередньо розглянутого. Також, окрім можливості порівнювати ціни товарів різних інтернет магазинів Hotline.ua надає користувачам підбірки товарів, відповідно до запитів.

5.4 Розроблення маркетингової системи

5.4.1 Стратегія монетизації проекту

Ідеєю проекту є зробити безкоштовний сервіс для користувачів, тому монетизація проекту заключається в розміщенні рекламних банерів від рекламодавців.

Автомобільна тематика є популярною серед великих рекламодавців. Сервіс може містити в собі велику кількість пріоритизованої реклами:

- Рекламний банер перед входом користувача на сервіс
- Реклама розміщена на фоні сторінки
- Реклама перед кожним пошуком користувачем автомобіля
- Реклама під час обробки запиту на оцінку вартості автомобіля
- Вбудовані рекламні банери в тілі застосунку
- Реклама відеороликів перед використанням функцій сайту

Також корисним буде розміщення реклами для автомобільних дилерів чи підприємців, які надають послуги ремонту, пригону, транспортування автомобілів.

Розміщення реклами на даному сервісі буде користуватись популярністю, адже відвідувачами веб-застосунку буде платіжно спроможна аудиторія яка розглядає можливість покупки автомобіля.

Надання диллерам інформації про популярні запити користувачів, на основі яких вони можуть формувати списки автомобілів, які користуються популярністю серед користувачів.

5.4.2 Просування проекту

Для монетизації проекту потрібно залучити велику кількість відвідувачів.

Підходи для просування проекту:

- Зміцнення SEO
- Створення блогу
- Соціальні медіа
- Публікації на форумах
- Google ads
- Facebook реклама

Зміцнення SEO - SEO, або пошукова оптимізація, є одним із найефективніших способів просування веб-сайту, оскільки він допомагає покращити рейтинг сайту в Google та інших пошукових системах. На відміну від платної пошукової реклами, пошукова оптимізація абсолютно безкоштовна – конкуренція відкрита для всіх залежно від якості вмісту, який ви публікуєте, плавності навігації, яку ви пропонуєте своїм відвідувачам, і кількості посилань, які ви отримуєте із зовнішніх джерел.

Щоб досягнути потужного SEO, буде розміщено певні ключові слова та фрази на веб-сайті, додавши альтернативний текст до зображень і оптимізувавши заголовки, щоб веб-сайт можна було знайти в пошукових системах.

Для ефективного просування веб-застосунку, гарною практикою є **створення безкоштовного блогу**, щоб підвищити свій SEO та підвищити рейтинг сайту в пошукових системах. Ведення блогів полегшує людям пошук сайту.

Соціальні медіа є найвідомішим безкоштовним методом просування нових веб-сайтів — і не дарма. Такі платформи, як Facebook, Twitter і Instagram, мають мільярди

користувачів, і на цих платформах легко зв'язатися з іншими місцевими компаніями та просувати власну.

Один із методів - **створити тему на автомобільному форумі** і дозволити іншим ділитись відгуками про розроблений сервіс. Інший варіант – взяти участь у існуючій темі розмов і посилатися на створений сервіс, коли це необхідно.

З **Google Ads** здійснюється оплата за те, щоб веб-сайт відображався вгорі результатів пошуку за певними ключовими словами. Коли запит вводиться з використанням цих ключових слів, Google переглядає всі облікові записи, які роблять ставки на ці ключові слова, і відображатиме та ранжуватиме оплачені результати відповідно до якості та релевантності реклами. Результати оплаченого пошуку відображаються першими, у верхній частині сторінок результатів. Цей підхід є чудовим способом збільшити трафік на веб-сай зацікавленим відвідувачам.

За допомогою **Facebook** можливо настроїти таргетовану рекламу для показу у релевантних стрічках. За допомогою такої реклами ми зможемо вказати тип аудиторії, перед якою з'явиться реклама, тобто вказали людей які цікавляться автомобілями.

Висновки до розділу

В даному розділі описана основна ідея проекту. Аналіз актуальності та огляд конкурентів. Також описані сценарії та стратегія монетизації. Розглянуто основні напрямки для просування проекту.

ВИСНОВКИ

В результаті виконання дипломної роботи було створено систему підтримки прийняття рішення під час купівлі автомобіля в Україні чи з-за кордону. Було розроблено алгоритми отримання та порівняння даних про автомобілі, а також на основі штучного інтелекту створено метод для оцінки приблизної вартості автомобіля, на основі його параметрів.

Побудовано трирівневий веб-застосунок за допомогою ASP.NET Core Web API на стороні бекенду та Angular на стороні фронтенду. Розроблено алгоритм парсингу двох інтернет ресурсів. Використовуючи інструменти ML.Net було створено та натреновано модель для прогнозування ціни автомобіля відповідно до параметрів. Для створення вибірки даних було створено окремого парсинг робота на основі Robotic Process Automation за допомогою UiPath.

Створена система послужить інструментом для людей які цікавляться автомобілями та їхньою ціною політикою. Попри те що система розроблялась для користувачів які на етапі вибору автомобіля, вона також має місце для користувачів які хочуть продати свій вживаний автомобіль та вагаються при його оціненні.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Freeman A. Pro ASP.NET Core 7th ed. Edition / Adam Freeman.
2. ASP.NET Core Web Api [Електронний ресурс] – Режим доступу до ресурсу: <https://metanit.com/sharp/aspnet5/3.1.php>. – доступний станом на 22.05.2022;
3. White M. Mastering JavaScript: A Complete Programming Guide Including jQuery, AJAX, Web Design, Scripting and Mobile Application Development / Michael B White. – (Kindle Edition).
4. stackoverflow.com [Електронний ресурс] – Режим доступу до ресурсу: <https://stackoverflow.com> – доступний станом на 22.05.2022.
5. Microsoft® Azure Official Site [Електронний ресурс] – Режим доступу до ресурсу: <https://azure.microsoft.com/> – доступний станом на 22.05.2022.
6. Angular.io [Електронний ресурс] – Режим доступу до ресурсу: <https://angular.io/> – доступний станом на 22.05.2022.
7. IBM Cloud Learn Hub [Електронний ресурс] – Режим доступу до ресурсу: <https://www.ibm.com/cloud/learn/three-tier-architecture> – доступний станом на 22.05.2022.
8. Techopedia [Електронний ресурс] – Режим доступу до ресурсу: <https://www.techopedia.com/definition/24649/three-tier-architecture> – доступний станом на 22.05.2022.
9. Dev Academy [Електронний ресурс] – Режим доступу до ресурсу: <https://dev-academy.com/angular-architecture-best-practices/> – доступний станом на 22.05.2022.
10. indepth.dev [Електронний ресурс] – Режим доступу до ресурсу: <https://indepth.dev/posts/1478/designing-angular-architecture-container-presentation-pattern> – доступний станом на 22.05.2022.
11. SAS.com [Електронний ресурс] – Режим доступу до ресурсу: https://www.sas.com/ru_ru/insights/analytics/machine-learning.html – доступний станом на 22.05.2022.
12. docs.microsoft.com [Електронний ресурс] – Режим доступу до ресурсу: <https://docs.microsoft.com/ru-ru/aspnet/core/tutorials/first-web-api?view=aspnetcore-6.0&tabs=visual-studio> – доступний станом на 22.05.2022.

13. W3SCHOOLS [Электронный ресурс] – Режим доступа до ресурсу: https://www.w3schools.com/js/js_api_intro.asp – доступный станом на 22.05.2022.
14. RESTFULAPI [Электронный ресурс] – Режим доступа до ресурсу: <https://restfulapi.net/> – доступный станом на 22.05.2022.
15. MEDIUM [Электронный ресурс] – Режим доступа до ресурсу: <https://medium.com/@andr.ivas12/rest-%D0%BF%D1%80%D0%BE%D1%81%D1%82%D1%8B%D0%BC-%D1%8F%D0%B7%D1%8B%D0%BA%D0%BE%D0%BC-90a0bca0bc78> – доступный станом на 22.05.2022.
16. DIGITALOCEAN [Электронный ресурс] – Режим доступа до ресурсу: https://www.digitalocean.com/community/conceptual_articles/s-o-l-i-d-the-first-five-principles-of-object-oriented-design-ru – доступный станом на 22.05.2022.
17. TECHTARGET [Электронный ресурс] – Режим доступа до ресурсу: <https://www.techtarget.com/searchenterpriseai/definition/machine-learning-ML> – доступный станом на 22.05.2022.
18. EXPERT.AI [Электронный ресурс] – Режим доступа до ресурсу: <https://www.expert.ai/blog/machine-learning-definition/> – доступный станом на 22.05.2022.
19. GEEKSFORGEEKS [Электронный ресурс] – Режим доступа до ресурсу: <https://www.geeksforgeeks.org/machine-learning/> – доступный станом на 22.05.2022.
20. C-SHARPCORNER [Электронный ресурс] – Режим доступа до ресурсу: <https://www.c-sharpcorner.com/article/getting-started-with-machine-learning-dotnet-ml-net/> – доступный станом на 22.05.2022.
21. RUBIKSCODE [Электронный ресурс] – Режим доступа до ресурсу: <https://rubikscore.net/2018/07/09/machine-learning-in-ml-net-using-machine-learning-model-in-asp-net-mvc-application/> – доступный станом на 22.05.2022.
22. PROCODEGUIDE [Электронный ресурс] – Режим доступа до ресурсу: <https://procodeguide.com/programming/ml-net-machine-learning-with-net-core/> – доступный станом на 22.05.2022.

23. DEV.TO [Электронный ресурс] – Режим доступа до ресурсу: <https://dev.to/esdanielgomez/sentiment-analysis-in-a-asp-net-web-application-with-ml-net-3kk9> – доступный станом на 22.05.2022.
24. DOTNETTHOUGHTS [Электронный ресурс] – Режим доступа до ресурсу: <https://dotnetthoughts.net/infuse-machine-learning-into-your-aspnetcore-apps/> – доступный станом на 22.05.2022.
25. MORIOH.COM [Электронный ресурс] – Режим доступа до ресурсу: <https://morioh.com/p/e60e35b76fd6> – доступный станом на 22.05.2022.
26. HABR [Электронный ресурс] – Режим доступа до ресурсу: <https://habr.com/ru/company/microsoft/blog/436728/> – доступный станом на 22.05.2022.
27. ACCESSIBLEAI [Электронный ресурс] – Режим доступа до ресурсу: https://accessibleai.dev/post/ml_net-video-game-classification/ – доступный станом на 22.05.2022.
28. CODEMAG [Электронный ресурс] – Режим доступа до ресурсу: <https://www.codemag.com/Article/1911042/ML.NET-Machine-Learning-for-.NET-Developers> – доступный станом на 22.05.2022.
29. MKMSOFT [Электронный ресурс] – Режим доступа до ресурсу: <https://mkmssoft.ua/product/metalloprokat/subd-ms-sql-server/> – доступный станом на 22.05.2022.
30. BESTPROG [Электронный ресурс] – Режим доступа до ресурсу: <https://www.bestprog.net/uk/2020/10/17/databases-normalization-concept-and-necessity-of-application-modification-anomalies-examples-ua/> – доступный станом на 22.05.2022.
31. ITOMYCHSTUDIO [Электронный ресурс] – Режим доступа до ресурсу: <https://itomychstudio.com/ua/blog/data-normalization-and-denormalization-when-each-should-be-used> – доступный станом на 22.05.2022.

ДОДАТКИ ДОДАТОК А

Скрипт наповнення бази даних з інформацією про марки автомобілів

```
SET IDENTITY_INSERT [dbo].[Carbrands] ON
```

```
INSERT [dbo].[AspNetUsers] ([Id], [UserName], [NormalizedUserName], [Email], [NormalizedEmail], [EmailConfirmed], [PasswordHash], [SecurityStamp], [ConcurrencyStamp], [PhoneNumber], [PhoneNumberConfirmed], [TwoFactorEnabled], [LockoutEnd], [LockoutEnabled], [AccessFailedCount], [LastActive], [DayOfBirth], [DisplayName]) VALUES (1, N'fytruk14@gmail.com', N'FYTRUK14@GMAIL.COM', NULL, NULL, 0, N'AQAAAAEAAcQAAAAEaC/Cp/Zfyg2WxeBpYJe8GOOXWwxmY8UyivI9LRwuRE0XQFdG88B/Pb7+PvYVEtqQ==', N'OUX2AZ2M7BOJAOH376OEP5RQ2LIIK7KK', N'2df205e8-bb68-4286-b06d-1900aae12603', NULL, 0, 0, NULL, 1, 0, CAST(N'2021-11-15T17:20:09.3250513' AS DateTime2), CAST(N'2021-11-15T17:19:16.2171229' AS DateTime2), N'test')
SET IDENTITY_INSERT [dbo].[AspNetUsers] OFF
GO
SET IDENTITY_INSERT [dbo].[Carbrands] ON
```

```
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (1, N'Audi', 6, N'audi')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (2, N'Bmw', 9, N'bmw')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (3, N'Ford', 24, N'ford')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (4, N'Mazda', 47, N'mazda')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (5, N'Mercedes-Benz', 48, N'mercedes-benz')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (6, N'Porsche', 59, N'porsche')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (7, N'Chevrolet', 13, N'chevrolet')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (8, N'Honda', 28, N'honda')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (9, N'Hyundai', 29, N'hyundai')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (10, N'Kia', 33, N'kia')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (11, N'Lexus', 38, N'lexus')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (12, N'Mitsubishi', 52, N'mitsubishi')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (13, N'Nissan', 55, N'nissan')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (14, N'Opel', 56, N'opel')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (15, N'Peugeot', 58, N'peugeot')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (16, N'Renault', 62, N'renault')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (17, N'Skoda', 70, N'skoda')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (18, N'Toyota', 79, N'toyota')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (19, N'Volkswagen', 84, N'volkswagen')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (20, N'Acura', 98, N'acura')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (21, N'Alfa Romeo', 3, N'alfa-romeo')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (22, N'Aston Martin', 5, N'aston-martin')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (23, N'Bentley', 8, N'bentley')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (24, N'Bugatti', 109, N'bugatti')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (25, N'Cadillac', 11, N'cadillac')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (26, N'Chery', 190, N'chery')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (27, N'Chrysler', 14, N'chrysler')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (28, N'Citroen', 15, N'citroen')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (29, N'Dacia', 17, N'dacia')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (30, N'Daewoo', 18, N'daewoo')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (31, N'DAF', 115, N'daf')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (32, N'Dodge', 118, N'dodge')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (33, N'Ferrari', 22, N'ferrari')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (34, N'Fiat', 23, N'fiat')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (35, N'Geely', 185, N'geely')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (36, N'Hummer', 127, N'hummer')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (37, N'Infiniti', 128, N'infiniti')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (38, N'Veco', 175, N'veco')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (39, N'Jaguar', 31, N'jaguar')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (40, N'Jeep', 32, N'jeep')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (41, N'Lamborghini', 35, N'lamborghini')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (42, N'Lada', 5553, N'lada')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (43, N'Land Rover', 37, N'land-rover')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (44, N'Lincoln', 135, N'lincoln')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (45, N'Maserati', 45, N'maserati')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (46, N'Maybach', 46, N'maybach')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (47, N'McLaren', 3101, N'mclaren')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (48, N'Rolls-Royce', 63, N'rolls-royce')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (49, N'Smart', 71, N'smart')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (50, N'Subaru', 75, N'subaru')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (51, N'Suzuki', 76, N'suzuki')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (52, N'Tesla', 2233, N'tesla')
INSERT [dbo].[Carbrands] ([CarbrandId], [NameBrand], [NameBrandRia], [NameBrandScout]) VALUES (53, N'Volvo', 85, N'volvo')
SET IDENTITY_INSERT [dbo].[Carbrands] OFF
GO
SET IDENTITY_INSERT [dbo].[Carmodels] ON
```

```
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (1, N'A3', 46, N'a3', 1)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (2, N'TT', 1837, N'tt', 1)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (3, N'Q7', 1943, N'q7', 1)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (4, N'A7', 32378, N'a7', 1)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (5, N'320', 31611, N'320', 2)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (6, N'M3', 3292, N'm3', 2)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (7, N'M5', 3213, N'm5', 2)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (8, N'X5 M', 3158, N'x5-m', 2)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (9, N'C-Max', 2036, N'c-max', 3)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (10, N'Edge', 1945, N'edge', 3)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (11, N'Focus', 240, N'focus', 3)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (12, N'RX-8', 403, N'rx-8', 4)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (13, N'3', 1692, N'3', 4)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (14, N'5000', 61285, N'5000', 1)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (15, N'80', 43, N'80', 1)
```


INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (433, N'S 63 AMG', 41370, N's-63-amg', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (434, N'S 65 AMG', 41371, N's-65-amg', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (436, N'SK-Series', 55064, N'sk-series', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (437, N'SL 300', 39159, N'sl-300', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (438, N'SL 350', 39161, N'sl-350', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (439, N'SL 400', 48460, N'sl-400', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (440, N'SL 450', 39164, N'sl-450', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (441, N'SL 500', 39165, N'sl-500', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (442, N'SL 55 AMG', 39166, N'sl-55-amg', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (443, N'SL 600', 39169, N'sl-600', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (444, N'SL 63 AMG', 39170, N'sl-63-amg', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (445, N'SL 65 AMG', 39171, N'sl-65-amg', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (446, N'SL-Class', 432, N'sl-class', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (447, N'SLC 300', 3015, N'slc-300', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (448, N'SLC 43 AMG', 1424, N'slc-43-amg', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (449, N'SLC-Class', 48780, N'slc-class', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (450, N'SLK 200', 40673, N'slk-200', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (451, N'SLK 230', 40674, N'slk-230', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (452, N'SLK 250', 40675, N'slk-250', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (453, N'SLK 280', 40676, N'slk-280', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (454, N'SLK 300', 40677, N'slk-300', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (455, N'SLK 350', 40680, N'slk-350', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (456, N'SLK 55 AMG', 40681, N'slk-55-amg', 5)
 GO
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (457, N'SLK-Class', 433, N'slk-class', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (458, N'SLS AMG', 24925, N'sls-amg', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (459, N'Smart', 38176, N'smart', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (460, N'Sprinter', 35249, N'sprinter', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (532, N'T1', 1428, N't1', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (534, N'V 200', 38988, N'v-200', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (535, N'V 220', 39128, N'v-220', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (536, N'V 230', 29491, N'v-230', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (537, N'V 250', 45088, N'v-250', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (538, N'V 300', 58565, N'v-300', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (540, N'Vaneo', 1429, N'vaneo', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (542, N'Vito', 35252, N'vito', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (559, N'X-Class', 52183, N'x-class', 5)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (562, N'Escape', 1183, N'escape', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (563, N'Escort', 237, N'escort', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (564, N'Explorer', 238, N'explorer', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (565, N'F-150', 2103, N'f-150', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (566, N'Fiesta', 239, N'fiesta', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (568, N'Fusion', 241, N'fusion', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (569, N'Galaxy', 242, N'galaxy', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (570, N'Kuga', 2874, N'kuga', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (571, N'Mondeo', 246, N'mondeo', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (572, N'Mustang', 2047, N'mustang', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (573, N'S-Max', 2037, N's-max', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (574, N'Scorpion', 250, N'scorpion', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (575, N'Sierra', 2970, N'sierra', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (576, N'Transit', 35396, N'transit', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (580, N'Aerostar', 1178, N'aerostar', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (581, N'B-Max', 40406, N'b-max', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (582, N'Bronco', 2776, N'bronco', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (584, N'Contour', 1180, N'contour', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (585, N'Cougar', 236, N'cougar', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (586, N'Courier', 1679, N'courier', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (587, N'Crown Victoria', 1181, N'crown-victoria', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (588, N'E-350', 3244, N'e-350', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (589, N'E-series', 30783, N'e-series', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (590, N'Econoline', 1182, N'econoline', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (591, N'Econovan', 1168, N'econovan', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (592, N'EcoSport', 45099, N'ecosport', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (596, N'Escort Express', 46752, N'escort-express', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (597, N'Excursion', 1169, N'excursion', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (598, N'Expedition', 1170, N'expedition', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (601, N'F-250', 2291, N'f-250', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (602, N'F-350', 3771, N'f-350', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (603, N'F-450', 3770, N'f-450', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (604, N'F-550', 35934, N'f-550', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (605, N'F-Series', 45836, N'f-series', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (606, N'Festiva', 31824, N'festiva', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (608, N'Flex', 3111, N'flex', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (609, N'Focus', 240, N'focus', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (614, N'Galaxie', 33116, N'galaxie', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (616, N'Granada', 243, N'granada', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (617, N'Grand C-MAX', 33614, N'grand-c-max', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (618, N'KA', 244, N'ka', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (620, N'Mainline', 60327, N'mainline', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (621, N'Maverick', 245, N'maverick', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (622, N'Mercury', 28799, N'mercury', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (625, N'Mustang GT', 3112, N'mustang-gt', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (626, N'Mustang Mach-E', 60585, N'mustang-mach-e', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (627, N'Mustang Shelby', 1190, N'mustang-shelby', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (628, N'Orion', 247, N'orion', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (629, N'Probe', 248, N'probe', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (630, N'Puma', 249, N'pu ma', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (631, N'Ranger', 1171, N'ranger', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (635, N'StreetKa', 253, N'streetka', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (636, N'Taurus', 1174, N'taurus', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (637, N'Taurus', 1193, N'taurus', 3)

INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (638, N'Tempo', 1175, N'tempo', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (639, N'Tourneo', 60507, N'tourneo', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (655, N'V8', 41753, N'v8', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (656, N'Windstar', 1682, N'windstar', 3)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (657, N'2', 391, N'2', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (659, N'323F', 29497, N'323f', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (660, N'5', 1912, N'5', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (661, N'6', 393, N'6', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (662, N'626', 394, N'626', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (663, N'929', 395, N'929', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (664, N'CX-3', 48041, N'cx-3', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (665, N'CX-30', 60223, N'cx-30', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (666, N'CX-5', 37381, N'cx-5', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (667, N'MPV', 2009, N'mpv', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (668, N'CX-9', 2010, N'cx-9', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (669, N'E2200', 41758, N'e2200', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (670, N'MPV', 397, N'mpv', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (671, N'MX-5', 399, N'mx-5', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (672, N'MX-6', 400, N'mx-6', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (673, N'Premacy', 401, N'premac', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (674, N'RX-7', 402, N'rx-7', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (676, N'Tribute', 404, N'tribute', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (677, N'121', 390, N'121', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (680, N'323', 392, N'323', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (686, N'Atenza', 1402, N'atenza', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (687, N'AZ-3', 1404, N'az-3', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (688, N'B-series', 1406, N'b-series', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (689, N'Biante', 1408, N'biante', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (690, N'Bongo', 1407, N'bongo', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (691, N'BT-50', 3761, N'bt-50', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (697, N'Demio', 396, N'demio', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (698, N'E-series', 1414, N'e-series', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (700, N'Familia', 2867, N'familia', 4)
 GO
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (701, N'Millenia', 2865, N'millenia', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (703, N'MX-3', 398, N'mx-3', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (710, N'Xedos', 406, N'xedos', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (711, N'Xedos 6', 35814, N'xedos-6', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (712, N'Xedos 9', 405, N'xedos-9', 4)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (713, N'911', 539, N'911', 6)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (714, N'944', 542, N'944', 6)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (715, N'Boxster', 544, N'boxster', 6)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (716, N'Cayenne', 546, N'cayenne', 6)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (717, N'Cayenne Coupe', 60099, N'cayenne-coupe', 6)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (718, N'Cayman', 1543, N'cayman', 6)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (719, N'Macan', 43158, N'macan', 6)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (720, N'Panamera', 2968, N'panamera', 6)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (721, N'Taycan', 58244, N'taycan', 6)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (722, N'Aveo', 1038, N'aveo', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (723, N'Bolt EV', 51091, N'bolt-ev', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (724, N'Camaro', 111, N'camaro', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (725, N'Captiva', 2544, N'captiva', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (726, N'Volt', 112, N'volt', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (727, N'Cruze', 3588, N'cruze', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (728, N'Epica', 2145, N'epica', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (729, N'Equinox', 1049, N'equinox', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (730, N'Evanda', 1050, N'evanda', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (731, N'Lacetti', 1052, N'lacetti', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (732, N'Malibu', 1054, N'malibu', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (733, N'Niva', 1059, N'niva', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (734, N'Nubira', 1060, N'nubira', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (735, N'Orlando', 35248, N'orlando', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (736, N'Spark', 33345, N'spark', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (737, N'Suburban', 1855, N'suburban', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (738, N'Tacuma', 1663, N'tacuma', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (739, N'Tahoe', 2070, N'tahoe', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (740, N'Trax', 49443, N'trax', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (741, N'Volt', 33783, N'volt', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (742, N'Astro', 62143, N'astro', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (743, N'Astro ???', 1037, N'astro-???'', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (744, N'Avalanche', 1854, N'avalanche', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (746, N'Blazer', 110, N'blazer', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (749, N'Caprice', 1042, N'caprice', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (751, N'Cavalier', 1043, N'cavalier', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (752, N'City Express', 57731, N'city-express', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (753, N'Cobalt', 1046, N'cobalt', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (754, N'Colorado', 1856, N'colorado', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (760, N'Explorer', 35741, N'explorer', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (761, N'Express', 62145, N'express', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (762, N'Express ??????', 31727, N'express-?????', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (763, N'Express ??????-??', 62146, N'express-?????-??', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (764, N'Express ???', 2457, N'express-???'', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (765, N'HHR', 1857, N'hhr', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (766, N'Impala', 1051, N'impala', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (767, N'Kalos', 49145, N'kalos', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (769, N'LANOS', 3658, N'lanos', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (770, N'Lumina', 1053, N'lumina', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (772, N'Matiz', 1055, N'matiz', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (776, N'Rezzo', 45178, N'rezzo', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (777, N'S-10', 2619, N's-10', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (778, N'Silverado', 1853, N'silverado', 7)

INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (779, N'Sonic', 49816, N'sonic', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (781, N'SSR', 1848, N'ssr', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (785, N'Tracker', 42012, N'tracker', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (786, N'TrailBlazer', 32143, N'trailblazer', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (787, N'Traverse', 3229, N'traverse', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (789, N'Uplander', 37651, N'uplander', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (790, N'Venture', 2714, N'venture', 7)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (791, N'Clarity', 54839, N'clarity', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (792, N'CR-V', 269, N'cr-v', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (793, N'CR-X', 270, N'cr-x', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (794, N'CR-Z', 30566, N'cr-z', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (795, N'Crosstour', 3896, N'crosstour', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (796, N'Element', 1234, N'element', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (797, N'Elision', 47418, N'elision', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (798, N'Fit', 1236, N'fit', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (799, N'FR-V', 1238, N'fr-v', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (800, N'HR-V', 271, N'hr-v', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (801, N'Insight', 272, N'insight', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (802, N'Jazz', 274, N'jazz', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (803, N'Legend', 275, N'legend', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (804, N'Odyssey', 1243, N'odyssey', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (805, N'Pilot', 1247, N'pilot', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (806, N'Prelude', 278, N'prelude', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (807, N'S2000', 280, N's2000', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (808, N'Stream', 282, N'stream', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (809, N'Accord', 262, N'accord', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (810, N'Acty', 34307, N'acty', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (811, N'Ascot', 1251, N'ascot', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (812, N'City', 1232, N'city', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (813, N'Civic', 265, N'civic', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (814, N'Civic Coupe', 42139, N'civic-coupe', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (816, N'Concerto', 268, N'concerto', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (819, N'CR-X Del Sol', 62183, N'cr-x-del-sol', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (826, N'Freed', 62469, N'freed', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (829, N'Inspire', 1239, N'inspire', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (832, N'Logo', 276, N'logo', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (833, N'NSX', 277, N'nsx', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (835, N'Passport', 1246, N'passport', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (838, N'Ridgeline', 2724, N'ridgeline', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (840, N'Shuttle', 281, N'shuttle', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (842, N'X-NV', 62480, N'x-nv', 8)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (843, N'Accent', 1258, N'accent', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (844, N'Avante', 2386, N'avante', 9)
 GO
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (845, N'Coupe', 286, N'coupe', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (846, N'Creta', 49259, N'creta', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (847, N'Elantra', 3086, N'elantra', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (848, N'Genesis', 2834, N'genesis', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (849, N'Getz', 35223, N'getz', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (850, N'Grand Santa Fe', 43651, N'grand-santa-fe', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (851, N'Grandeur', 2186, N'grandeur', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (852, N'i10', 2770, N'i10', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (853, N'i20', 2771, N'i20', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (854, N'i30', 2772, N'i30', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (855, N'i40', 38715, N'i40', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (856, N'ix35', 3901, N'ix35', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (857, N'Kona', 51933, N'kona', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (858, N'Matrix', 290, N'matrix', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (859, N'Santa FE', 293, N'santa-fe', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (860, N'Sonata', 295, N'sonata', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (861, N'Tucson', 1268, N'tucson', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (862, N'Veloster', 36447, N'veloster', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (864, N'Atos', 285, N'atos', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (866, N'Azera', 2497, N'azera', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (870, N'Equus', 36796, N'equus', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (871, N'Excel', 33898, N'excel', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (872, N'Gallop', 1261, N'gallop', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (874, N'Genesis Coupe', 3082, N'genesis-coupe', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (877, N'Grand Starex', 41439, N'grand-starex', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (879, N'H 100 ????', 31532, N'h-100-????', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (880, N'H 100 ????.-????', 56681, N'h-100-????-????', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (881, N'H 100 ????.', 1263, N'h-100-????', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (882, N'H 200 ????.', 31533, N'h-200-????', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (883, N'H 200 ????.-????', 55769, N'h-200-????-????', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (884, N'H 200 ????.', 1686, N'h-200-????', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (885, N'H 350 ????.', 40188, N'h-350-????', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (886, N'H 350 ????.', 53099, N'h-350-????', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (887, N'H-1', 43940, N'h-1', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (888, N'H-Series', 39068, N'h-series', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (889, N'H1 ????.', 31531, N'h1-????', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (890, N'H1 ????.', 1262, N'h1-????', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (895, N'Ioniq', 50018, N'ioniq', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (896, N'ix20', 48040, N'ix20', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (898, N'ix55', 3670, N'ix55', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (900, N'Lafesta EV', 62243, N'lafesta-ev', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (901, N'Lantra', 289, N'lantra', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (903, N'Maxcruz', 43519, N'maxcruz', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (904, N'Palisade', 60031, N'palisade', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (905, N'Pony', 291, N'pony', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (906, N'S-Coupe', 294, N's-coupe', 9)
 INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (908, N'Solaris', 35041, N'solaris', 9)

```

INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (910, N'Starex', 28531, N'starex', 9)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (911, N'Terracan', 1267, N'terracan', 9)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (912, N'Tiburon', 2824, N'tiburon', 9)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (913, N'Trajet', 297, N'trajet', 9)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (915, N'Tuscani', 36174, N'tuscani', 9)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (917, N'Ve nue', 58455, N've nue', 9)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (918, N'Ve racruz', 36010, N've racruz', 9)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (919, N'XG', 1269, N'xg', 9)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (920, N'Carens', 317, N'carens', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (921, N'Carnival', 1310, N'carnival', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (922, N'Ce ed', 2033, N'ceed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (923, N'Ce rato', 1311, N'cerato', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (924, N'Ce rato Kou p', 3660, N'cerato-koup', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (925, N'Ce ed', 49491, N'forte', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (926, N'K5', 58260, N'k5', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (927, N'K7', 61476, N'k7', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (928, N'Magentis', 319, N'magentis', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (929, N'Mohave', 3586, N'mohave', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (930, N'Ni ro', 50175, N'niro', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (931, N'Optima', 1840, N'optima', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (932, N'Picanto', 1306, N'picanto', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (933, N'ProCe ed', 2794, N'proceed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (934, N'Ce ed', 49491, N'forte', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (935, N'Sedona', 324, N'sedona', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (936, N'Sorento', 326, N'sorento', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (937, N'Soul', 3488, N'soul', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (938, N'Sportage', 327, N'sportage', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (939, N'Stonic', 51968, N'stonic', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (940, N'Amanti', 31946, N'amanti', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (941, N'Ve nue', 1307, N've nue', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (942, N'Besta', 1308, N'besta', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (943, N'Ce dENZA', 3897, N'cedENZA', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (946, N'Ce star', 32961, N'ce star', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (950, N'Ce larus', 318, N'ce larus', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (951, N'Ce ed', 33200, N'ce ed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (953, N'Ce ed', 1314, N'ce ed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (954, N'Ce ed', 60866, N'ce ed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (957, N'Ce ed', 58971, N'ce ed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (960, N'Ce ed', 321, N'ce ed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (962, N'Ce ed', 1316, N'ce ed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (965, N'Ce ed', 59587, N'ce ed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (966, N'Ce ed', 31560, N'ce ed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (967, N'Ce ed', 1318, N'ce ed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (968, N'Ce ed', 322, N'ce ed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (970, N'Ce ed', 54680, N'ce ed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (971, N'Ce ed', 1319, N'ce ed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (973, N'Ce ed', 54062, N'ce ed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (975, N'Ce ed', 31591, N'ce ed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (976, N'Ce ed', 1321, N'ce ed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (977, N'Ce ed', 325, N'ce ed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (980, N'Ce ed', 3410, N'ce ed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (982, N'Ce ed', 51264, N'ce ed', 10)
INSERT [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (984, N'Ce ed', 57410, N'ce ed', 10)
GO [dbo].[Carmodels] ([CarmodelId], [NameModel], [NameModelRia], [NameModelScout], [CarbrandId]) VALUES (111, N'420', 42498, N'420', 2)
GO

```

ДОДАТОК Б

Вибіркові файли програмного коду бекенду

Startup.cs

```
using System.IO;
using AutoFromAbroad.WebUI.Extensions;
using AutoFromAbroad.WebUI.Middleware;
using AutoFromAbroad.WebUI.SignalR;
using Microsoft.AspNetCore.Builder;
using Microsoft.AspNetCore.Hosting;
using Microsoft.AspNetCore.SpaServices.AngularCli;
using Microsoft.Extensions.Configuration;
using Microsoft.Extensions.DependencyInjection;
using Microsoft.Extensions.Hosting;
using NLog;

namespace AutoFromAbroad.WebUI
{
    public class Startup
    {
        readonly string MyAllowSpecificOrigins = "_myAllowSpecificOrigins";

        public Startup(IConfiguration configuration)
        {
            LogManager.LoadConfiguration(string.Concat(Directory.GetCurrentDirectory(), "nlog.config"));
            Configuration = configuration;
        }

        public IConfiguration Configuration { get; }

        // This method gets called by the runtime. Use this method to add services to the container.
        public void ConfigureServices(IServiceCollection services)
        {
            services.AddApplicationServices(Configuration);//ApplicationServiceExtensions
            services.AddControllers();
            //AddCors here with http://localhost:4200 domain of angular
            services.AddCors(options =>
            {
                options.AddPolicy(name: MyAllowSpecificOrigins,
                    builder =>
                    {
                        builder.WithOrigins("http://localhost:4200")
                            .AllowAnyHeader()
                            .AllowAnyMethod()
                            .AllowCredentials();
                    }
                );
            });

            services.AddIdentityServices(Configuration);
            services.AddSignalR();
            services.AddControllers().AddNewtonsoftJson(options =>
                options.SerializerSettings.ReferenceLoopHandling = Newtonsoft.Json.ReferenceLoopHandling.Ignore
            );

            //-----
            #region MyRegion
            //services.ConfigureCors();
            //services.AddControllersWithViews();
            //services.ConfigureLoggerService();
            //services.ConfigureRepositoryManager();
            //services.ConfigureEmployeeService();
            ///TODO:
            ///services.ConfigureCarbrandService();
            ///services.ConfigureCarmodelService();

            //services.AddAutoMapper(typeof(Startup));

            //services.Configure<DataProtectionTokenProviderOptions>(opt =>
            //    opt.TokenLifespan = TimeSpan.FromHours(2));

            /// In production, the Angular files will be served from this directory
            //services.AddSpaStaticFiles(configuration =>
            //{
```

```

// configuration.RootPath = "ClientApp/dist";
//});

//var mapperConfig = new MapperConfiguration(mc =>
//{
//    mc.AddProfile(new MappingProfile());
//});
//IMapper mapper = mapperConfig.CreateMapper();
//services.AddSingleton(mapper);
//services.AddControllers();
//services.AddScoped<JwtHandler>();
//services.AddScoped<IUnitOfWork, UnitOfWork>();
//services.AddScoped<ICarBrandService, CarBrandService>();
//services.AddScoped<ICarModelService, CarModelService>();
//services.AddScoped<IEmailSender, EmailSender>();
//services.AddScoped<IAllCarsService, ParseCars>();
//services.AddDataServices(Configuration);
//var jwtSettings = Configuration.GetSection("JwtSettings");
//services.AddAuthentication(opt =>
//{
//    opt.DefaultAuthenticateScheme = JwtBearerDefaults.AuthenticationScheme;
//    opt.DefaultChallengeScheme = JwtBearerDefaults.AuthenticationScheme;
//}).AddJwtBearer(options =>
//{
//    options.TokenValidationParameters = new TokenValidationParameters
//    {
//        ValidateIssuer = true,
//        ValidateAudience = true,
//        ValidateLifetime = true,
//        ValidateIssuerSigningKey = true,
//        ValidIssuer = jwtSettings.GetSection("validIssuer").Value,
//        ValidAudience = jwtSettings.GetSection("validAudience").Value,
//        IssuerSigningKey = new SymmetricSecurityKey(Encoding.UTF8.GetBytes(jwtSettings.GetSection("securityKey").Value))
//    };
//});

//var emailConfig = Configuration
//    .GetSection("EmailConfiguration")
//    .Get<EmailConfiguration>();
//services.AddSingleton(emailConfig);

}
#endregion

// This method gets called by the runtime. Use this method to configure the HTTP request pipeline.
public void Configure(IApplicationBuilder app, IWebHostEnvironment env)
{
    if (env.IsDevelopment())
    {
        app.UseDeveloperExceptionPage();
    }
    else
    {
        app.UseExceptionHandler("/Error");
        // The default HSTS value is 30 days. You may want to change this for production scenarios, see https://aka.ms/aspnetcore-hsts.
        app.UseHsts();
    }

    app.UseHttpsRedirection();
    app.UseStaticFiles();
    if (!env.IsDevelopment())
    {
        app.UseSpaStaticFiles();
    }

    //-----
    app.UseMiddleware<ExceptionMiddleware>();

    app.UseHttpsRedirection();

    app.UseRouting();
    app.UseCors(MyAllowSpecificOrigins);
    app.UseAuthentication();
    app.UseAuthorization();
    //-----
    app.UseEndpoints(endpoints =>
    {
        endpoints.MapControllerRoute(

```

```

        name: "default",
        pattern: "{controller}/{action=Index}/{id?}";

//-----
endpoints.MapControllers();
endpoints.MapHub<PresenceHub>("hubs/presence");
//-----
});

app.UseSpa(spa =>
{
// To learn more about options for serving an Angular SPA from ASP.NET Core,
// see https://go.microsoft.com/fwlink/?linkid=864501

spa.Options.SourcePath = "ClientApp";

if (env.IsDevelopment())
{
spa.UseAngularCliServer(npmScript: "start");
}
});
}
}
}
CarsController.cs
using AutoFromAbroad.BLL.Interfaces;
using AutoFromAbroad.WebUI.Model;
using Microsoft.AspNetCore.Mvc;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;

namespace AutoFromAbroad.WebUI.Controllers
{
[Route("api/getcars")]
[ApiController]
public class CarsController : Controller
{
private readonly IAllCarsService _allCarsService;
private readonly ICarBrandService _brandService;
private readonly ICarModelService _carModelService;

public CarsController(IAllCarsService allCarsService, ICarBrandService brandService,
ICarModelService carModelService)
{
_allCarsService = allCarsService;
_brandService = brandService;
_carModelService = carModelService;
}

[HttpGet("GetRia")]
public IActionResult ListRia([FromQuery] string brand, string model, string year)
{
var getcarsRia = _allCarsService.ParseRia(brand, model, year);
return Ok(getcarsRia);
}

[HttpGet("GetScout")]
public IActionResult ListScout([FromQuery] string brand, string model, string year, string country)
{
var getcarsScout = _allCarsService.ParseScout(brand, model, year, country);
return Ok(getcarsScout);
}

[HttpGet("GetBrands")]
public async Task<ActionResult> CarBrandList()
{
return Ok(await _brandService.GetAll());
}

[HttpGet("GetModels")]
public IActionResult GetModels(int id)
{
var models = _carModelService.GetById(id);
if (models.Count() == 0)
{
return BadRequest();
}
}
}
}
}

```

```

        return Ok(models);
    }
}

```

CarBrandService.cs

```

using System;
using System.Collections.Generic;
using System.Text;
using System.Threading.Tasks;
using AutoFromAbroad.BLL.EntitiesDTO.Carbrand;
using AutoFromAbroad.BLL.Interfaces;
using AutoFromAbroad.DAL.Entities;
using AutoFromAbroad.DAL.Interfaces;
using AutoMapper;

namespace AutoFromAbroad.BLL.Services
{
    public class CarBrandService:ICarBrandService
    {
        private readonly IUnitOfWork unitOfWork;
        private readonly IMapper mapper;

        public CarBrandService(IUnitOfWork unitOfWork,IMapper mapper)
        {
            this.unitOfWork = unitOfWork;
            this.mapper = mapper;
        }

        public async Task<IEnumerable<CarbrandDto>>GetAll()
        {
            var data = await unitOfWork.CarbrandRepository.ListItemsAsync();
            return mapper.Map<IEnumerable<Carbrand>,IEnumerable<CarbrandDto>>(data);
        }
    }
}

```

Clearance.cs

```

using AutoFromAbroad.BLL.EntitiesDTO.Car;
using AutoFromAbroad.BLL.Interfaces;
using System;

namespace AutoFromAbroad.BLL.Services
{
    public class Clearance : IClearanceService
    {
        public ClearanceInfo CalcClearance(int year, int typeoffuel, double cm3, double price)
        {
            ClearanceInfo clearanceInfo = new ClearanceInfo();
            if ((2020 - year) >= 15)
            {
                year = 15;
            }
            else
            {
                year = 2020 - year;
            }
            if (typeoffuel == 1)
            {
                if (cm3 <= 3000)
                {
                    clearanceInfo.Akcuz = Math.Round(50 * (cm3 / 1000) * year, 2);
                }
                else if (cm3 >= 3000)
                {
                    clearanceInfo.Akcuz = Math.Round(100 * (cm3 / 1000) * year, 2);
                }
            }
            else if (typeoffuel == 2)
            {
                if (cm3 <= 3500)
                {
                    clearanceInfo.Akcuz = Math.Round(75 * (cm3 / 1000) * year, 2);
                }
                else if (cm3 >= 3500)
                {

```

```

        clearanceInfo.Akucz = Math.Round(150 * (cm3 / 1000) * year, 2);
    }
}
else if (typeoffuel == 3)
{
    throw new Exception("Need electro car calculator");
}
clearanceInfo.Muto = Math.Round((price * 0.1), 2);
clearanceInfo.Pdv = Math.Round(((clearanceInfo.Muto + price + clearanceInfo.Akucz) * 0.2), 2);
clearanceInfo.Finalprice = Math.Round((price + clearanceInfo.Muto + clearanceInfo.Akucz + clearanceInfo.Pdv), 2);

return clearanceInfo;
}
}
}
}

```

Вибіркові файли програмного коду фронтенду

search-bar.component.html

```

<div class="search-sec">
  <div class="form-inline">
    <div class="cols">
      <select id="brand" class="selectpicker" title="Brand" (change)="getModels($event.target.value)" data-live-search="true" data-size="10">
        <option *ngFor="let car of carBrands" [value]="car.carbrandId" [attr.data-ria]="car.nameBrandRia" [attr.data-
scout]="car.nameBrandScout">{{ car.nameBrand }}</option>
      </select>
    </div>
    <div class="cols">
      <select id="model" class="selectpicker" title="Model" data-live-search="true" data-size="10">
        <option *ngFor="let model of carModels" [value]="model.carmodelId" [attr.data-modelria]="model.nameModelRia" [attr.data-
modelscout]="model.nameModelScout">{{ model.nameModel }}</option>
      </select>
    </div>
    <div class="cols" id="country">
      <select id="country" class="selectpicker" title="Country" data-live-search="true" data-size="10">
        <option *ngFor="let country of Countries" [attr.data-id]="country.id">{{ country.name }}</option>
      </select>
    </div>
    <div class="cols">
      <select id="year" class="selectpicker" title="Year" data-live-search="true" data-size="10">
        <option *ngFor="let number of numbers">{{ number }}</option>
      </select>
    </div>
    <div class="cols">
      <input id="search" class="btnsearch" (click)="findCars()" [disabled]="!searchBtn" value="Пошук" type="button" [ngStyle]="{'background-color':
searchBtn ? '#0e746c': 'grey'}">
    </div>
  </div>
</div>
<div class="cars">
  <div class="itemscolumn" id="Scout">
    <div *ngIf="scoutEmpty" class="card">
      <div class="countryicon"></div>
      <div class="card_image">
        <div class="empty-result">
          <p><em>Не знайдено в {{selectedCountry}}</em></p>
          <p>Спробуйте шукати іншу Модель</p>
        </div>
      </div>
    </div>
  </div>

```

```

</div>
<div *ngFor="let car of carsScout" class="card">
  <div class="textbox">
    <p class="carname">{{ car.name }}</p>
    <p class="additionaltext"> {{ car.kilometers}} &nbsp;{{ car.typeoffuel}}</p>
  </div>
  <div class="countryicon">
    
  </div>
  <div class="card_image">
    <a href="{{ car.link }}" target="_blank">
      
      <div class="middle">
        <div class="hovertext">Натисніть щоб перейти на сайт</div>
      </div>
    </a>
  </div>
  <div class="footercard">
    <div class="cardprice">
      <p>${ { car.price }}</p>
    </div>
  </div>
</div>
</div>
<div class="itemsrow">
  <div class="itemscolumn" id="Ria">
    <div *ngIf="riaEmpty" class="card">
      <div class="countryicon"></div>
      <div class="card_image">
        <div class="empty-result">
          <p><em>Не знайдено в Україні</em></p>
          <p>Спробуйте шукати іншу Модель</p>
        </div>
      </div>
    </div>
  </div>
  <div *ngFor="let car of carsRia" class="card">
    <div class="textbox">
      <p class="carname">{{ car.name }}</p>
      <p class="additionaltext"> {{ car.kilometers}} &nbsp;{{ car.typeoffuel}}</p>
    </div>
    <div class="countryicon">
      
    </div>
    <div class="card_image">
      <a href="{{ car.link }}" target="_blank">
        
        <div class="middle">
          <div class="hovertext">Натисніть щоб перейти на сайт</div>
        </div>
      </a>
    </div>
  </div>
  <div class="footercard">

```

```

        <div class="cardprice">
            <p>${ { car.price } }</p>
        </div>
    </div>
</div>
</div>
</div>
</div>
</div>

```

search-bar.component.ts

```

import { Component, ElementRef, OnInit, ViewChild } from '@angular/core';
import { first } from 'rxjs/operators';
import { FindCarsService } from '../shared/services/findcars.service';
import { CarBrand } from '../_interfaces/car/brandDto.model';
import { Car } from '../_interfaces/car/findCarDto.model';
import { CarModel } from '../_interfaces/car/modelDto.model';
import { Country } from '../_models/Country';
declare var $: any;

@Component({
  selector: 'app-search-bar',
  templateUrl: './search-bar.component.html',
  styleUrls: ['./search-bar.component.css']
})
export class SearchBarComponent implements OnInit {
  constructor(private _findcarservice: FindCarsService) { }
  carBrands: CarBrand[] = [];
  carModels: CarModel[] = [];
  numbers = [];
  carsRia: Car[] = [];
  carsScout: Car[] = [];
  Countries: Country[] = [
    { id: "D", name: "Німеччина", icon: "" },
    { id: "F", name: "Франція", icon: "" },
    { id: "B", name: "Бельгія", icon: "" },
    { id: "A", name: "Австрія", icon: "" },
    { id: "S", name: "Іспанія", icon: "" },
    { id: "I", name: "Італія", icon: "" },
    { id: "NL", name: "Люксембург", icon: "" },
    { id: "N", name: "Нідерланди", icon: "" }
  ];
  searchBtn = true;
  riaEmpty = false;
  scoutEmpty = false;
  selectedCountry=null;

  ngOnInit(): void {
    this.getStartBrands();
    this.numbers = this.range(1980, 2021);

    $('#brand').on('show.bs.select', this.selectRefresh);
  }

```

```

$('#model').on('show.bs.select', this.selectRefresh);

window.onresize = function () {
  if (window.innerWidth < 992) {
    $(".selectpicker").attr("style", "width: 90px");
    $(".dropdown.bootstrap-select").attr("style", "width: 90px");
    $(".dropdown-menu.show").attr("style", "width: 90px");
    $('.selectpicker').selectpicker("render");
  }
  else {
    $(".selectpicker").attr("style", "width: 150px");
    $(".dropdown.bootstrap-select").attr("style", "width: 150px");
    $('.selectpicker').selectpicker("render");
  }
}

window.onload = function () {
  if (window.innerWidth < 992) {
    $(".selectpicker").attr("style", "width: 90px");
    $(".dropdown.bootstrap-select").attr("style", "width: 90px");
    $(".dropdown-menu.show").attr("style", "width: 90px");
    $('.selectpicker').selectpicker("render");
  }
  else {
    $(".selectpicker").attr("style", "width: 150px");
    $(".dropdown.bootstrap-select").attr("style", "width: 150px");
    $('.selectpicker').selectpicker("render");
  }
}

getStartBrands() {
  this._findcarservice.getBrandList().pipe(first()).subscribe(data => {
    if (data !== undefined) {
      this.carBrands = data;
    }
  });
}

getModels(id: number) {
  this._findcarservice.getModelLis(id).pipe(first()).subscribe(data => {
    this.carModels = data;
  });
}

range(start, end) {
  return Array(end - start + 1).fill(0).map( (_, idx) => start + idx)
}

findCars() {
  this.searchBtn = false;
  this.selectedCountry=$('#country').find(':selected').text();
  var Country = $('#country').find(':selected').data('id');
  var BrandScout = $('#brand').find(':selected').data('scout');
  var ModelScout = $('#model').find(':selected').data('modelscout');
}

```

```

var BrandRia = $('#brand').find(':selected').data('ria');
var ModelRia = $('#model').find(':selected').data('modelria');
var Year = $('#year').find(':selected').text();
this._findcarservice.findCarsRia(BrandRia, ModelRia, Year).subscribe(riaCars => {
  this._findcarservice.findCarsScout(BrandScout, ModelScout, Year, Country).subscribe(scoutCars => {
    if (riaCars.length == 0) {
      this.riaEmpty = true;
    } else { this.riaEmpty = false; }
    if (scoutCars.length == 0) {
      this.scoutEmpty = true;
    } else { this.scoutEmpty = false; }
    this.carsScout = scoutCars;
    this.carsRia = riaCars;
    this.searchBtn = true;
  });
});

}

selectRefresh = function () {
$('#selectpicker').selectpicker("refresh");
if (window.innerWidth < 992) {
  $(".selectpicker").attr("style", "width: 90px");
  $(".dropdown.bootstrap-select").attr("style", "width: 90px");
  $(".dropdown-menu.show").attr("style", "width: 100px");
  $('#selectpicker').selectpicker("render");
}
else {
  $(".selectpicker").attr("style", "width: 150px");
  $(".dropdown.bootstrap-select").attr("style", "width: 150px");
  $('#selectpicker').selectpicker("render");
}
}
}

```