

Team Members

1. Rahul Googikoll [1AM19CS158]
2. Rahul Mishra [1AM19CS159]
3. Saurav Kumar [1AM19CS195]

Introduction

1.1 Project outline

This application is going to be used by:

1. Employees
2. Customer

This application is meant for users (Employees and Customers) who can access the information from database about car deals. Our objective is so that executives and administrators can easily automate their work with less effort. For example they can generate quotations for any enquiry for a car in no time. They can also have the customer's feedback to the show room authority regarding show room services.

1.2 Project Goals

The System should be capable of performing the following:

- Store basic information regarding cars, employees, customers, accessories and services provided by the organization.
- Store salary information of employees (entered by the team leaders in each city) such as, working hours, salary per hour, salary before tax, tax percentage, total amount of tax paid , salary after tax, social security fee, on monthly basis
- Help employees automate the billing process by quickly fetching car/accessory details from the database.
- System should be able to generate invoices and maintain transaction details.
- Should be able to list the inventory of cars currently available in a dealership.
- Should help customers to access car details like model, make, engine no., price etc.

- Help the organization to maintain customer details like customer name, transaction/purchase details, car purchased, purchase medium, purchase date etc.
- Display details of various dealerships of the same franchise/organization like, profits, dealership name, number of cars in dealership, number of employees etc.

System Specifications

2.1 Hardware requirements

Minimum requirements :

- **processor:** Intel Pentium 4 / AMD Athlon series
- **RAM:** 512MB
- **Storage:** 100MB

Recommended requirements:

- **processor:** Intel 11th generation I-series/AMD Ryzen Threadripper series
- **RAM:** 8GB
- **Storage:** 500GB

2.2 Software requirements

OS: Windows, GNU/Linux Distributions, Mac OS, BSD, 64-bit

Tech stack used:

Front end: React-JS, Tailwind-CSS

- **React-JS:**

React is a free and open-source front-end JavaScript library for building user interfaces based on UI components. It is maintained by Meta and a community of individual developers and companies. React can be used as a base in the development of single-page or mobile applications

- **Tailwind-css:**

Tailwind CSS is basically **a utility-first CSS framework for rapidly building**

custom user interfaces. It is a highly customizable, low-level CSS framework that gives you all of the building blocks you need to build bespoke designs without any annoying opinionated styles you have to fight to override.

Back end: Django, SQLite3

- **Django:**

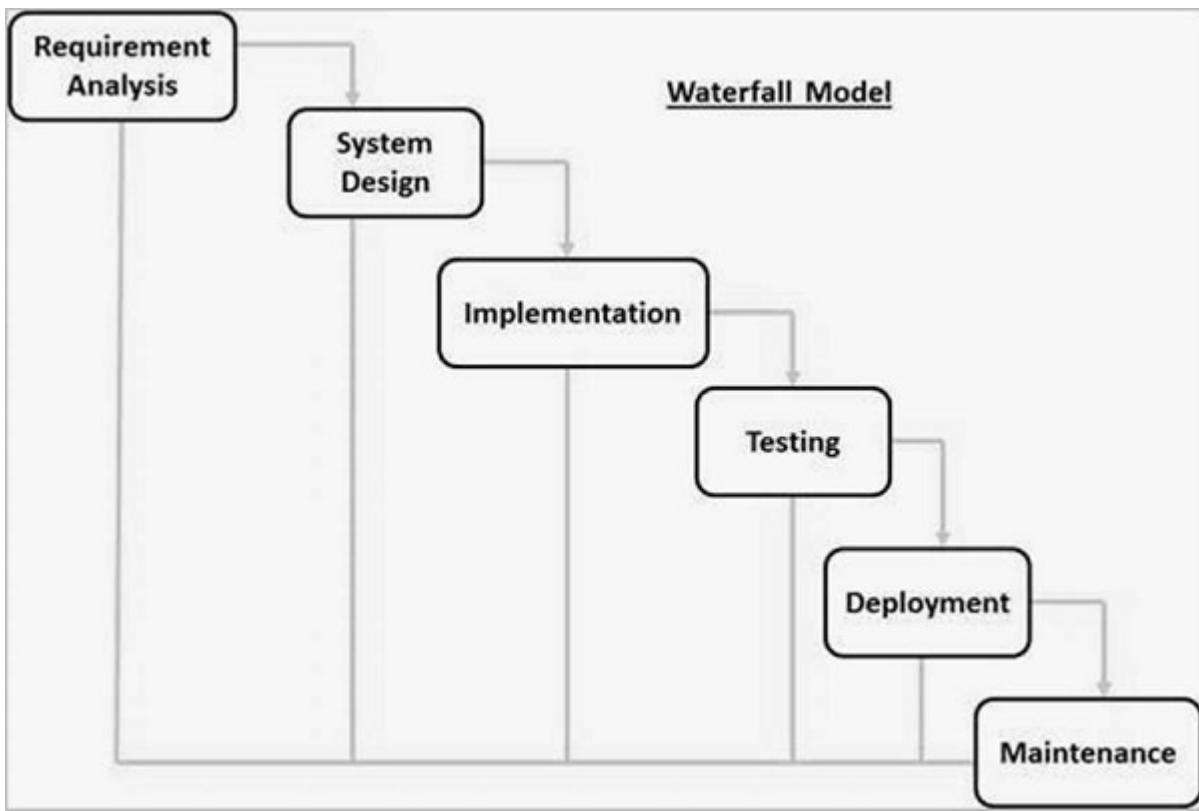
Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel. It's free and open source.

- **SQLite3:**

SQLite is a C-language library that implements a [small](#), [fast](#), [self-contained](#), [high-reliability](#), [full-featured](#), SQL database engine. SQLite is the [most used](#) database engine in the world. SQLite is built into all mobile phones and most computers and comes bundled inside countless other applications that people use every day.

Design

3.1 Development model:



- **Requirement Analysis:**

All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document.

- **Design stage:**

The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.

- **Implementation:**

With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as Unit Testing.

- **Integration and Testing:**

All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.

- **Deployment of System:**

Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market.

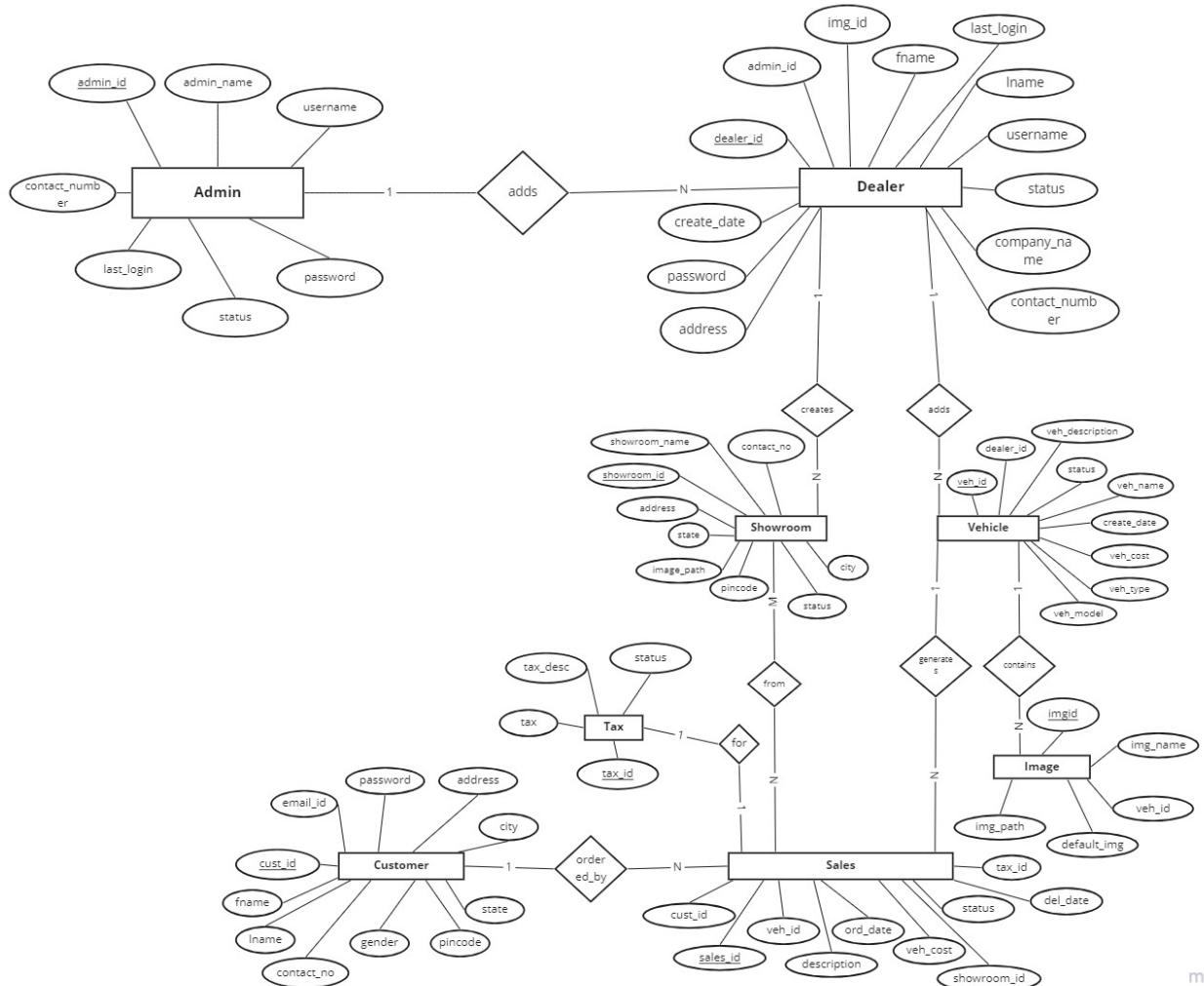
- **Maintenance:**

There are some issues which come up in the client environment. To fix those issues, patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

All these phases are cascaded to each other in which progress is seen as flowing steadily downwards (like a waterfall) through the phases. The next phase is started only after the defined set of goals are achieved for previous phase and it is signed off, so the name "Waterfall Model". In this model, phases do not overlap.

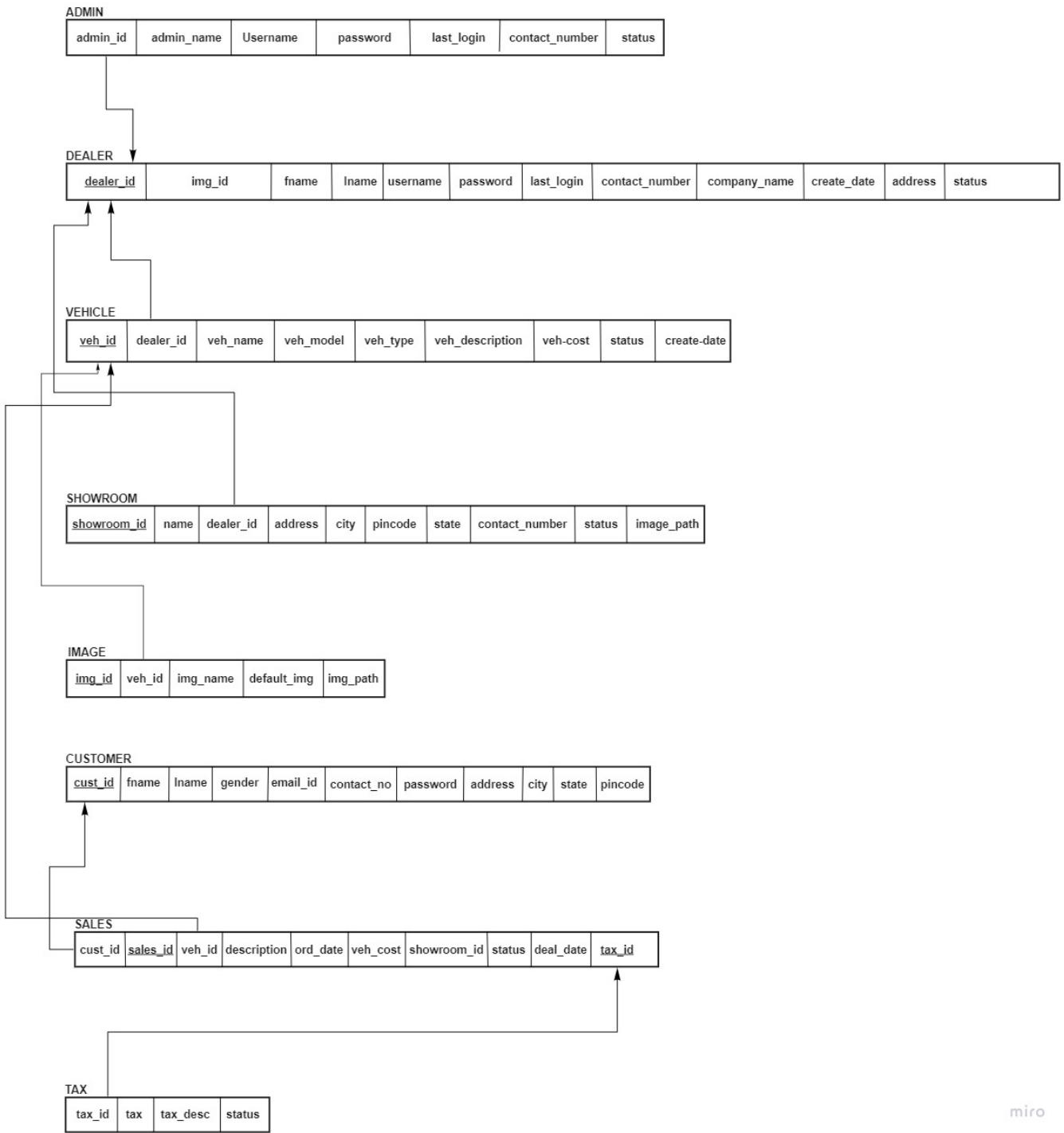
3.2 Database description:

ER Diagram



miro

Schema Diagram



miro

Documentation

4. Backend

4.1 Models (Tables)

Dealer Model

Contains the Dealer table definition and attribute list, based and derived from the schema diagram

```
# src/models.py
class Dealer(models.Model) => (attrs)
```

The Dealer Table hereby referred to as the "Dealer Object" further into this section of the documentation consists of the following attributes

- dealer_id

```
(variable) dealer_id: IntegerField => (primary_key: bool, unique: bool)
```

```
dealer_id = models.IntegerField(primary_key=True, unique=True)
```

Refers to the identification number of the dealer , `primary_key=True` parameter is setting the attribute to a primary key. The `unique=True` parameter is useful in referencing this particular attribute via **Foreign Keys**.

- dealer_img

Attribute definition:

```
(variable) dealer_img: ImageField => (blank: bool)
```

source code:

```
dealer_img = models.ImageField(blank=False)
```

Refers to the dealer's image field the `ImageField` method stores uploaded image in the room `/media/` folder. `blank = False`

- dealer_name

Attribute definition:

```
(variable) dealer_name: CharField =>(max_length: int, blank: bool, null: bool)
```

source code:

```
dealer_name = models.CharField(max_length=100, blank=False, null=False)
```

Refers to the dealer's name field the `NameField` method stores the name of the dealer class, `null=False` parameter setting to the null attribute. `max_length` is set to (100) and it's restricts the number of characters used by the user by (100).

`Blank=False`

- dealer_username

Attribute definition:

```
(variable) dealer_username: ForeignKey => ((class) User, (parameter) on_delete: (...) -> None, (parameter) related_name: str | None)
```

Source code:

```
dealer_username = models.ForeignKey(User, on_delete=models.CASCADE, related_name='d_username')
```

refers to the username of the dealer, as apparent here this attribute is a foreign key that refers to `(class) User`, it has a parameter `(parameter) on_delete: (...) -> None` set to `models.CASCADE` which means on deletion of a record from this table it reflects by committing deletion on the referenced table as well. The `(parameter) related_name: str | None` parameter sets up an alternate name or view to avoid arising naming conflicts.

- last_login

Attribute definition:

```
(variable) last_login: DateTimeField => ((parameter) auto_now: bool)
```

Source code:

```
last_login = models.DateTimeField(auto_now=True)
```

The `last_login` attribute refers to the last time the dealer had logged in. the `(parameter) auto_now: bool` is set to `True`, the `auto_now` parameter automatically sets the login time based on server timezone.

- contact

Attribute definition:

```
(variable) contact: IntegerField => (null: bool, blank: bool)
```

Source code:

```
contact = models.IntegerField(null=False, blank=False)
```

refers to the contact details of the dealer class.the `(parameter) null: bool` is set to `False`,`Blank=False`

- company_name

Attribute definition:

```
(variable) company_name: CharField => ((parameter) max_length: int  
| None, (parameter) null: bool, (parameter) blank: bool)
```

Source code:

```
company_name = models.CharField(max_length=100, null=False,  
blank=False)
```

The `company_name` attribute refers to the name of the company that the dealer belongs to, since it is a `CharField` it consists of the `(parameter) max_length` that

is set to `100`. This field cannot be left `null` or `blank`.

- address

Attribute definition:

```
(variable) address: TextField => ((parameter) max_length: int |  
None, (parameter) null: bool, (parameter) blank: bool)
```

Source code:

```
address = models.TextField(max_length=300, blank=False, null=False)
```

The `Address` attribute refers to the Dealer's address, `(parameter) max_length: int | None` is set to `300`, and `(parameter) blank: bool` and `(parameter) null: bool` are set to `False`.

- created

Attribute definition:

```
(variable) created: DateTimeField => ((parameter) auto_created:  
bool)
```

Source code:

```
created = models.DateTimeField(auto_created=True)
```

The `created` attribute refers to the date the dealer's profile was created, `(parameter) auto_created: bool` is set to `True`, this is a method passed in as a parameter which sets the value to current date and time based on the server region and timezone.

- status

Attribute definition:

```
(variable) status: CharField => ((parameter) max_length: int | None,  
(parameter) choices: _FieldChoices | None, (parameter) default: Any)
```

here the choices are defined in a list of choice sets,

```
(constant) STATUS_CHOICE: list
```

Source code:

```
STATUS_CHOICE = [('Active', 'Active'), ('Inactive', 'Inactive')]
```

```
status = models.CharField(max_length=20, choices=STATUS_CHOICE,  
default= 'Active')
```

refers to the status in the dealer class. Since it is a `CharField` it consists of the `(parameter) max_length` that is set to `20`.
by `default= Any` the `STATUS_CHOICE` is set to `Active`.

Vehicle Model

Contains the Vehicle table definition and attribute list, based and derived from the schema diagram

- vehicle_id

Attribute definition:

```
(variable) vehicle_id: IntegerField => ((parameter) primary_key:  
bool, (parameter) unique: bool)
```

Source code:

```
vehicle_id = models.IntegerField(primary_key=True, unique=True)
```

The `vehicle_id` attribute refers to the numeric id of the vehicle entries present in the table, it is a `primary_key` and it is set to be `unique`.

- `dealer_id`

Attribute definition:

```
(variable) dealer_id: ForeignKey => ((class) Dealer, (parameter)
on_delete: (...) -> None, (parameter) to_field: str | None)
```

Source code:

```
dealer_id = models.ForeignKey(Dealer, to_field='dealer_id',
on_delete=models.CASCADE)
```

refers to the id of the dealer, as apparent here this attribute is a foreign key that refers to `(class) dealer`, it has a parameter `(parameter) on_delete: (...) -> None` set to `models.CASCADE` which means on deletion of a record from this table it reflects by committing deletion on the referenced table as well. the `to_field` is set to `dealer_id`

- `vehicle_img`

Attribute definition:

```
(variable) vehicle_img: ImageField => (blank: Any)
```

Source code:

```
vehicle_img = models.ImageField(blank=False)
```

The `vehicle_img` attribute contains the vehicle images uploaded to the server stored at `/media/` folder in the `MEDIA_DIR` directory of the server, `blank` is set to `False` so an image is required for every record uploaded.

- `name`

Attribute definition:

```
(variable) name: CharField => ((parameter) max_length: int | None,  
(parameter) blank: bool)
```

Source code:

```
name = models.CharField(max_length=100, blank=False)
```

The `name` attribute refers to the vehicle name, it's `max_length` is set to `100` and `blank` is `False` so it cannot be left empty.

- type

Attribute definition:

```
(variable) type: CharField => ((parameter) max_length: int | None,  
(parameter) choices: _FieldChoices | None, (parameter) blank: bool)
```

here the choices are defined in a list of choice sets,

```
(constant) VEHICLE_TYPE: list
```

```
VEHICLE_TYPE = [  
    ("Hatchback", "Hatchback"),  
    ("Sedan", "Sedan"),  
    ("MPV", "MPV"),  
    ("SUV", "SUV"),  
    ("Crossover", "Crossover"),  
    ("Coupe", "Coupe"),  
    ("Convertible", "Convertible")  
]
```

Source code:

```
type = models.CharField(max_length=100, choices=VEHICLE_TYPE,  
blank=False)
```

The `type` attribute refers to the vehicle type, it's `max_length` is set `100` and `blank=False` so it cannot be left empty. The `choices` parameter is set to `VEHICLE_TYPE`

- description

Attribute definition:

```
(variable) description: TextField => ((parameter) max_length: int | None, (parameter) blank: bool)
```

Source code:

```
description = models.TextField(max_length=100, blank=False)
```

The `description` attribute refers to vehicle description, it's `max_length` is `200` and `blank` is set to `False`.

- cost

Attribute definition:

```
(variable) cost: IntegerField => ((parameter) blank: bool)
```

Source code:

```
cost = models.IntegerField(blank=False)
```

The `cost` attribute refers to the vehicle cost, it cannot be blank since `blank` is set to `False`.

- status

Attribute definition:

```
(variable) status: CharField => ((parameter) max_length: int | None, (parameter) choices: _FieldChoices | None, (parameter) blank: bool)
```

here the choices are defined in a list of choice sets,

```
(constant) VEH_STATUS: list
```

```
VEH_STATUS = [
    ("For Sale", "For Sale"),
    ("Sold", "Sold")
]
```

Source code:

```
status = models.CharField(max_length=20, choices=VEH_STATUS,
blank=False, default='For Sale')
```

The `status` attribute refers to the vehicle sale status, the choice sets are defined in `VEH_STATUS`, and `default` choice is `For Sale`.

- created

Attribute definition:

```
(variable) created: DateTimeField => ((parameter) auto_now_add:
bool)
```

Source code:

```
created = models.DateTimeField(auto_now_add=True)
```

The `created` attribute refers to the date on which the vehicle was created. The `auto_now_add:bool` is set to `True`.

Customer Model

Contains the Customer table definition and attribute list, based and derived from the schema diagram

- customer_id

Attribute definition:

```
(variable) customer_id: IntegerField => ((parameter) primary_key:  
bool, (parameter) unique: bool)
```

Source code:

```
customer_id = models.IntegerField(primary_key=True, unique=True)
```

The `customer_id` attribute refers to numeric customer id, it is a `primary_key`, and it is `unique`.

- first_name

Attribute definition:

```
(variable) first_name: CharField => ((parameter) max_length: int |  
None, (parameter) blank: bool)
```

Source code:

```
first_name = models.CharField(max_length=100, blank=False)
```

The `first_name` attribute stores the customer first name. The `max_length` is set to `100`. It cannot be left `blank`.

- last_name

Attribute definition:

```
(variable) last_name: CharField => ((parameter) max_length: int |  
None, (parameter) blank: bool)
```

Source code:

```
last_name = models.CharField(max_length=100, blank=False)
```

The `last_name` attribute refers to the customer's last name, it cannot be left `blank`.

- gender

Attribute definition:

```
(variable) gender: CharField => ((parameter) max_length: int | None,  
(parameter) blank: bool, (parameter) choices: _FieldChoices | None)
```

here the choices are defined in a list of choice sets,

```
(constant) GENDER: list
```

```
GENDER = [  
    ("Male", "Male"),  
    ("Female", "Female"),  
    ("Other", "Other")  
]
```

Source code:

```
gender = models.CharField(max_length=100, blank=False,  
choices=GENDER)
```

The `gender` attribute refers to the customer's gender, the gender field has `choices` defined in a list of sets defined in `GENDER`.

- email

Attribute definition:

```
(variable) email: CharField => ((parameter) max_length: int | None,  
(parameter) blank: bool)
```

Source code:

```
email = models.CharField(max_length=100, blank=False)
```

The `email` attribute refers to the customer's email. It's `max_length` is set to `100` and it cannot be left `blank`.

- contact

Attribute definition:

```
(variable) contact: IntegerField => ((parameter) blank: bool)
```

Source code:

```
contact = models.IntegerField(blank=False)
```

The `contact` attribute refers to the contact information of the customer, it cannot be left `blank`.

- city

Attribute definition:

```
(variable) city: CharField => ((parameter) max_length: int | None,  
(parameter) blank: bool)
```

Source code:

```
city = models.CharField(max_length=100, blank=False)
```

The `city` attribute refers to the customer's city name in which he/she resides. It's `max_length` is set to `100` and it cannot be left. `blank`.

- state

Attribute definition:

```
(variable) state: CharField => ((variable) state: CharField,  
(parameter) blank: bool)
```

Source code:

```
state = models.CharField(max_length=100, blank=False)
```

The `state` attribute refers to the customer's state, it cannot be `blank`.

- address

Attribute definition:

```
(variable) address: TextField => ((parameter) max_length: int |  
None)
```

Source code:

```
address = models.TextField(max_length=100, blank=False)
```

The `address` attribute refers to the customer's address, it cannot be left `blank`.

Sale Model

Contains the Sale table definition and attribute list, based and derived from the schema diagram

- customer_id

Attribute definition:

```
(variable) customer_id: ForeignKey => ((class) Customer, (parameter)  
to_field: str | None, (parameter) on_delete: (...) -> None)
```

Source code:

```
customer_id = models.ForeignKey(Customer, to_field='customer_id',  
on_delete=models.CASCADE)
```

The `customer_id` attribute refers to the numeric id of the referenced customer.

- [sale_id](#)

Attribute definition:

```
(variable) sale_id: IntegerField => ((parameter) primary_key: bool,  
(parameter) unique: bool)
```

Source code:

```
sale_id = models.IntegerField(primary_key=True, unique=True)
```

The `sale_id` attribute refers to the numeric id of a sale made.

- [vehicle_id](#)

Attribute definition:

```
(variable) vehicle_id: ForeignKey => ((class) Vehicle, (parameter)  
to_field: str | None, (parameter) on_delete: (...) -> None)
```

Source code:

```
vehicle_id = models.ForeignKey(Vehicle, to_field='vehicle_id',  
on_delete=models.CASCADE)
```

The `vehicle_id` attribute refers to the numeric id of the referenced vehicle object.

- [description](#)

Attribute definition:

```
(variable) description: TextField =>((parameter) max_length: int |  
None, (parameter) blank: bool)
```

Source code:

```
description = models.TextField(max_length=100, blank=False)
```

The `description` attribute describes the sale made.

- order_date

Attribute definition:

```
(variable) order_date: DateTimeField =>((variable) order_date:  
DateTimeField)
```

Source code:

```
order_date = models.DateTimeField(auto_created=True)
```

The `order_date` attribute refers to the date the sale was made.

- cost

Attribute definition:

```
(variable) cost: IntegerField => ((parameter) blank: bool)
```

Source code:

```
cost = models.IntegerField(blank=False)
```

The `cost` refers to the cost paid by the customer to the dealership.

- deal_date

Attribute definition:

```
(variable) deal_date: DateTimeField => ((parameter) auto_created:  
bool)
```

Source code:

```
deal_date = models.DateTimeField(auto_created=True)
```

The `deal_date` attribute refers to the date the car deal was made.

- [status](#)

Attribute definition:

```
(variable) status: CharField => ((parameter) max_length: int | None,  
(parameter) blank: bool, (parameter) choices: _FieldChoices | None)
```

here the choice list is given in a list of sets,

```
(constant) TAX_STATUS: list
```

```
SALE_STATUS = [  
    ("Sold", "Sold"),  
    ("On hold", "On hold"),  
    ("Rejected", "Rejected")  
]
```

Source code:

```
status = models.CharField(max_length=20, blank=False,  
choices=SALE_STATUS)
```

The `status` attribute refers to the deal status.

- tax_id

Attribute definition:

```
(variable) tax_id: IntegerField => ((parameter) unique: bool,  
(parameter) blank: bool)
```

Source code:

```
tax_id = models.IntegerField(unique=True, blank=False)
```

The `tax` attribute refers to the id of the tax paid for the vehicle. The `unique` is set to `True` and it cannot be left blank.

Tax Model

Contains the Tax table definition and attribute list, based and derived from the schema diagram

- tax_id

Attribute definition:

```
(variable) tax_id: ForeignKey => ((class) Sale, (parameter)  
to_field: str | None, (parameter) on_delete: (...) -> None)
```

Source code:

```
tax_id = models.ForeignKey(Sale, to_field='tax_id',  
on_delete=models.CASCADE)
```

The `tax_id` attribute refers to the numeric id referenced in the Sale Model/Table.

- tax

Attribute definition:

```
(variable) tax: IntegerField =>((variable) tax: IntegerField)
```

Source code:

```
tax = models.IntegerField(blank=False)
```

The `tax` attribute refers to the amount of tax applied to the bill of sale.

- [description](#)

Attribute definition:

```
(variable) description: TextField=>((parameter) max_length: int | None, (parameter) blank: bool)
```

Source code:

```
description = models.TextField(max_length=100, blank=False)
```

The `description` attribute refers to the description of the tax paid.

- [status](#)

Attribute definition:

```
(variable) status: CharField => ((parameter) max_length: int | None, (parameter) blank: bool, (parameter) choices: _FieldChoices | None)
```

here the choice list is given as,

```
(constant) TAX_STATUS: list
```

```
TAX_STATUS = [  
    ("Approved", "Approved"),
```

```
("Pending", "Pending"),  
("Rejected", "Rejected")
```

Source code:

```
status = models.CharField(max_length=20, blank=False,  
choices=TAX_STATUS)
```

The `status` attribute refers to the status of the tax payment, it refers to `TAX_STATUS` list for choice sets.

Snapshots

5.1 Frontend w/backend admin panel

127.0.0.1:8000

127.0.0.1:8000

Apps View Passwords (17) YouTube Google Online Courses - A... GitHub Programming articles Get AWS Certified S... Problemset - Codef... workouts megathread/games... Reading list

Buzzcars Home Vehicles Services About Us Contact Us Login SignUp

FIND YOUR CAR



Explore Cars

127.0.0.1:8000

127.0.0.1:8000

Apps View Passwords (17) YouTube Google Online Courses - A... GitHub Programming articles Get AWS Certified S... Problemset - Codef... workouts megathread/games... Reading list

Buzzcars Home Vehicles Services About Us Contact Us Login SignUp

150+ Branches

4770+ Cars Sold

320+ Happy Clients

1500+ News Cars

Popular Vehicles



New Model

Price: **Rs.30,62,000/-**

New • 2021 • Automatic • Petrol • 183mph

Book A Quote

The screenshot shows a web browser window for the 'Buzzcars' website at 127.0.0.1:8000. The top navigation bar includes links for Home, Vehicles, Services, About Us, Contact Us, Login, and SignUp. The main heading 'Our Services' is centered above six service cards arranged in two rows of three. Each card features a teal circular icon with a white car-related symbol (e.g., steering wheel, battery, oil can) and a title followed by a short lorem ipsum text block and a 'Read More' button.

The screenshot shows a web browser window for the URL 127.0.0.1:8000. The page title is "Buzzcars". The main content area features a large heading "Contact Us" above a map of Mumbai's northern suburbs. The map highlights areas like Jogeshwari West, Goregaon, Oshiwara, Andheri West, and Thane. Key landmarks marked on the map include Umiya Mata Mandir, SRV Hospital, Movietime: The Hub Mall, PVR Goregaon West, City Centre Mall, Carnival Cinemas, Imagica Tickets, Infiniti Mall, 1BHK, Oves, and various temples and railway stations. Below the map is a "Get In Touch" form with fields for First Name, Last Name, Gender, Email, Contact, City, and a message input field.

Buzzcars

Home Vehicles Services About Us Contact Us

Login

SignUp

Contact Us

Jogeshwari West
Mumbai, Maharashtra 400047
Directions
View larger map

H NAGAR
Umiya Mata Mandir
GOREGAON
SRV Hospital
Movietime: The Hub Mall
PVR Goregaon West
City Centre Mall
Carnival Cinemas
Moviestar
Oves
1BHK
okhandwala Market Road
SS NAGAR
ANDHERI - WEST
Infiniti Mall
Imagica Tickets
Jogeshwari Caves
Matoshri N. Thackeray
Kokan Hospital

Get In Touch

First Name:

Last Name:

Gender:

Email:

Contact:

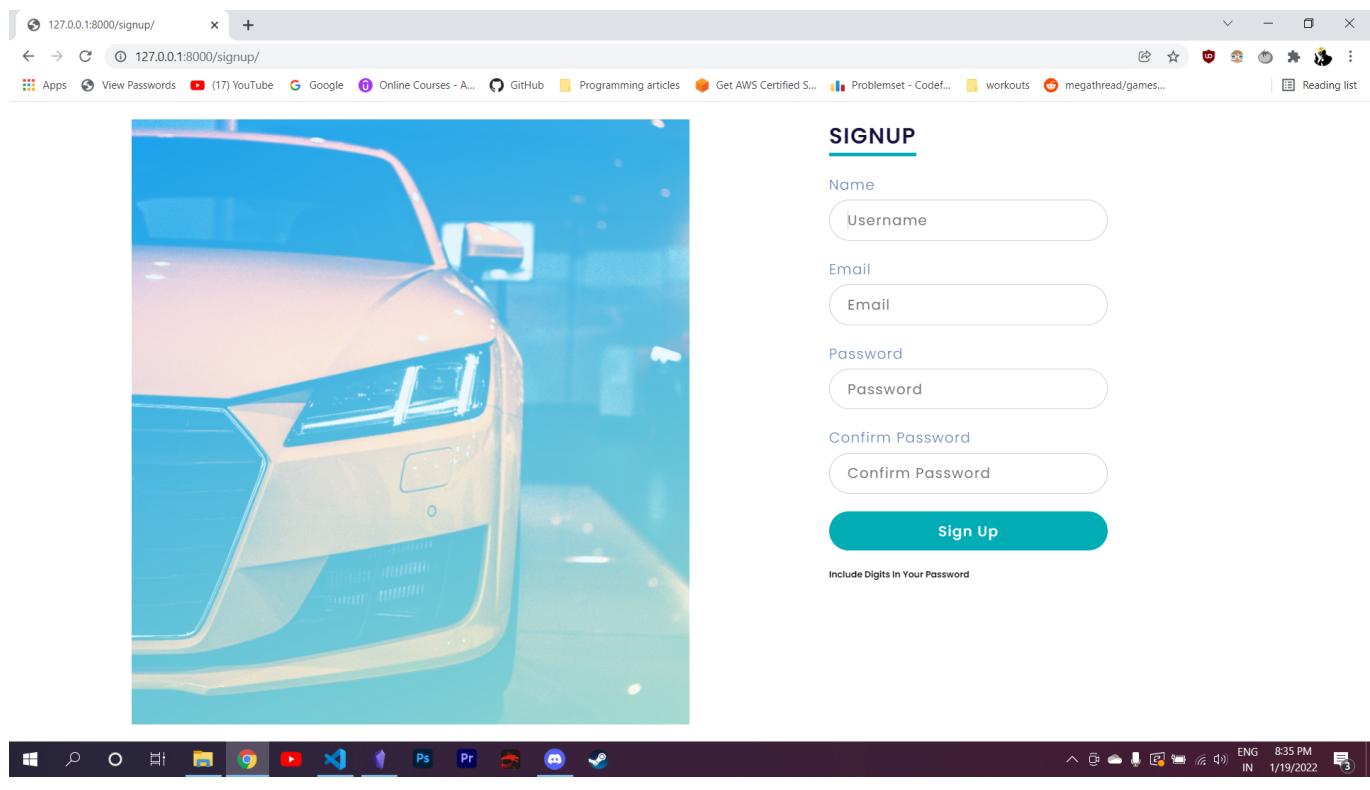
City:

City

The screenshot shows a web browser displaying a car listing page from 'Buzzcars'. The page features four cars in a grid format: a white Mercedes-Benz SLS AMG, a blue BMW M3 (1990), a red KIA Sportage, and a blue Lotus Emira. Each car card includes a thumbnail image, the car's name, its category (e.g., Coupe, Sedan, SUV, Crossover), and a 'For Sale' status. Below each car's name is a brief description. At the bottom of each card is a 'Book A Quote' button.

Car Model	Category	Status
Mercedes SLS AMG	Coupe	For Sale
BMW M3 (1990)	Sedan	For Sale
KIA Sportage	SUV	For Sale
Lotus Emira	Crossover	For Sale

The screenshot shows a web browser window with the URL `127.0.0.1:8000/login/`. The page features a large, blurry background image of a blue sports car. To the right of the image is a login form titled "LOGIN". The form includes fields for "Username" and "Password", a "Remember Me" checkbox, and a large teal "Sign In" button. Below the sign-in area are buttons for "Or Login With Google" and "Facebook". The browser's address bar also displays the URL `127.0.0.1:8000/login/`.



A screenshot of the Django administration site. The top navigation bar shows "Site administration | Django site" and the URL "127.0.0.1:8000/admin/". The main content area is divided into two sections: "AUTHENTICATION AND AUTHORIZATION" and "SRC". The "AUTHENTICATION AND AUTHORIZATION" section contains "Groups" and "Users" with "Add" and "Change" buttons. The "SRC" section contains "Customers", "Dealers", "Sales", "Tax", and "Vehicles" with "Add" and "Change" buttons. To the right of these sections is a sidebar titled "Recent actions" which lists recent model instances like "Mercedes EQE Vehicle", "Ferrari 296 GTB Tax", and "Saurav Kumar Customer".



Source code

File Edit Selection View Go Run Terminal Help

models.py - Buzzcars - Visual Studio Code

BUZZCARS

- src
 - __pycache__
 - migrations
 - _init_.py
 - admin.py
 - apps.py
 - forms.py
 - models.py
 - tests.py
 - urls.py
 - views.py
- static
- templates
- db.sqlite3
- manage.py
- env
- Synopsis.md

```
from email.headerregistry import Address
from re import A
from django.db import models
from django.contrib.auth.models import User
# Create your models here.

STATUS_CHOICE = [ ('Active', 'Active'), ('Inactive', 'Inactive')]

class Dealer(models.Model):
    dealer_id = models.IntegerField(primary_key=True, unique=True)
    dealer_img = models.ImageField(blank=False)
    dealer_name = models.CharField(max_length=100, blank=False, null=False)
    dealer_username = models.ForeignKey(User, on_delete=models.CASCADE, related_name='d_username')
    last_login = models.DateTimeField(auto_now=True)
    contact = models.IntegerField(null=False, blank=False)
    company_name = models.CharField(max_length=100, null=False, blank=False)
    address = models.TextField(max_length=300, blank=False, null=False)
    created = models.DateTimeField(auto_created=True)
    status = models.CharField(max_length=20, choices=STATUS_CHOICE, default='Active')

    def __str__(self):
        return str(self.dealer_id)

VEHICLE_TYPE = [
    ("Hatchback", "Hatchback"),
    ("Sedan", "Sedan"),
    ("MPV", "MPV"),
    ("SUV", "SUV"),
    ("Crossover", "Crossover"),
    ("Coupe", "Coupe"),
    ("Convertible", "Convertible")
]

VEH_STATUS = [
    ("For Sale", "For Sale"),
    ("Sold", "Sold")
]

class Vehicle(models.Model):
    vehicle_id = models.IntegerField(primary_key=True, unique=True)
    dealer_id = models.ForeignKey(Dealer, to_field='dealer_id', on_delete=models.CASCADE)
    vehicle_img = models.ImageField(blank=False)
    name = models.CharField(max_length=100, blank=False)
    type = models.CharField(max_length=100, choices=VEHICLE_TYPE, blank=False)
    description = models.TextField(max_length=100, blank=False)
    cost = models.IntegerField(blank=False)
    status = models.CharField(max_length=20, choices=VEH_STATUS, blank=False, default='For Sale')
    created = models.DateTimeField(auto_now_add=True)

    def __str__(self):
        return self.name

GENDER = [
    ("Male", "Male"),
    ("Female", "Female"),
    ("Other", "Other")
]

class Customer(models.Model):
    customer_id = models.IntegerField(primary_key=True, unique=True)
```

Ln 48, Col 98 Spaces: 4 UTF-8 CRLF Python ⚡ Prettier ⌂ ENG 8:37 PM IN 1/19/2022 (3)

File Edit Selection View Go Run Terminal Help

models.py - Buzzcars - Visual Studio Code

BUZZCARS

- src
 - __pycache__
 - migrations
 - _init_.py
 - admin.py
 - apps.py
 - forms.py
 - models.py
 - tests.py
 - urls.py
 - views.py
- static
- templates
- db.sqlite3
- manage.py
- env
- Synopsis.md

```
VEHICLE_TYPE = [
    ("Hatchback", "Hatchback"),
    ("Sedan", "Sedan"),
    ("MPV", "MPV"),
    ("SUV", "SUV"),
    ("Crossover", "Crossover"),
    ("Coupe", "Coupe"),
    ("Convertible", "Convertible")
]

VEH_STATUS = [
    ("For Sale", "For Sale"),
    ("Sold", "Sold")
]

class Vehicle(models.Model):
    vehicle_id = models.IntegerField(primary_key=True, unique=True)
    dealer_id = models.ForeignKey(Dealer, to_field='dealer_id', on_delete=models.CASCADE)
    vehicle_img = models.ImageField(blank=False)
    name = models.CharField(max_length=100, blank=False)
    type = models.CharField(max_length=100, choices=VEHICLE_TYPE, blank=False)
    description = models.TextField(max_length=100, blank=False)
    cost = models.IntegerField(blank=False)
    status = models.CharField(max_length=20, choices=VEH_STATUS, blank=False, default='For Sale')
    created = models.DateTimeField(auto_now_add=True)

    def __str__(self):
        return self.name

GENDER = [
    ("Male", "Male"),
    ("Female", "Female"),
    ("Other", "Other")
]

class Customer(models.Model):
    customer_id = models.IntegerField(primary_key=True, unique=True)
```

Ln 48, Col 98 Spaces: 4 UTF-8 CRLF Python ⚡ Prettier ⌂ ENG 8:37 PM IN 1/19/2022 (3)

File Edit Selection View Go Run Terminal Help

models.py - Buzzcars - Visual Studio Code

EXPLORER BUZZCARS Buzzcars media src _pycache_ migrations __init__.py admin.py apps.py forms.py models.py tests.py urls.py views.py static templates db.sqlite3 manage.py env Synopsis.md

```
Buzzcars > src > models.py > Vehicle
    ('other', 'other')
]
]

class Customer(models.Model):
    customer_id = models.IntegerField(primary_key=True, unique=True)
    first_name = models.CharField(max_length=100, blank=False)
    last_name = models.CharField(max_length=100, blank=False)
    gender = models.CharField(max_length=100, blank=False, choices=GENDER)
    email = models.CharField(max_length=100, blank=False)
    contact = models.IntegerField(blank=False)
    city = models.CharField(max_length=100, blank=False)
    state = models.CharField(max_length=100, blank=False)
    address = models.TextField(max_length=100, blank=False)

    def __str__(self):
        return self.first_name + " " + self.last_name

SALE_STATUS = [
    ("Sold", "Sold"),
    ("On hold", "On hold"),
    ("Rejected", "Rejected")
]

class Sale(models.Model):
    customer_id = models.ForeignKey(Customer, to_field='customer_id', on_delete=models.CASCADE)
    sale_id = models.IntegerField(primary_key=True, unique=True)
    vehicle_id = models.ForeignKey(Vehicle, to_field='vehicle_id', on_delete=models.CASCADE)
    description = models.TextField(max_length=100, blank=False)
    order_date = models.DateTimeField(auto_created=True)
    cost = models.IntegerField(blank=False)
    deal_date = models.DateTimeField(auto_created=True)
    status = models.CharField(max_length=20, blank=False, choices=SALE_STATUS)
    tax_id = models.IntegerField(unique=True, blank=False)

    def __str__(self):
        return str(self.vehicle_id)

TAX_STATUS = [
```

Ln 48, Col 98 Spaces: 4 UTF-8 CRLF Python Prettier ENG 8:37 PM IN 1/19/2022

File Edit Selection View Go Run Terminal Help

models.py - Buzzcars - Visual Studio Code

EXPLORER BUZZCARS Buzzcars media src _pycache_ migrations __init__.py admin.py apps.py forms.py models.py tests.py urls.py views.py static templates db.sqlite3 manage.py env Synopsis.md

```
Buzzcars > src > models.py > Vehicle
    ('other', 'other')
]
]

class Customer(models.Model):
    customer_id = models.ForeignKey(Customer, to_field='customer_id', on_delete=models.CASCADE)
    sale_id = models.IntegerField(primary_key=True, unique=True)
    vehicle_id = models.ForeignKey(Vehicle, to_field='vehicle_id', on_delete=models.CASCADE)
    description = models.TextField(max_length=100, blank=False)
    order_date = models.DateTimeField(auto_created=True)
    cost = models.IntegerField(blank=False)
    deal_date = models.DateTimeField(auto_created=True)
    status = models.CharField(max_length=20, blank=False, choices=SALE_STATUS)
    tax_id = models.IntegerField(unique=True, blank=False)

    def __str__(self):
        return str(self.vehicle_id)

TAX_STATUS = [
    ("Approved", "Approved"),
    ("Pending", "Pending"),
    ("Rejected", "Rejected")
]

class Tax(models.Model):
    tax_id = models.ForeignKey(Sale, to_field='tax_id', on_delete=models.CASCADE)
    tax = models.IntegerField(blank=False)
    description = models.TextField(max_length=100, blank=False)
    status = models.CharField(max_length=20, blank=False, choices=TAX_STATUS)

    def __str__(self):
        return str(self.tax_id)
```

Ln 48, Col 98 Spaces: 4 UTF-8 CRLF Python Prettier ENG 8:37 PM IN 1/19/2022

File Edit Selection View Go Run Terminal Help

forms.py - Buzzcars - Visual Studio Code

```
Buzzcars > src > forms.py > ContactForm > __init___.py
```

```
1 from django import forms
2 from django.forms import ModelForm
3 from django.contrib.auth.forms import UserCreationForm
4 from django.contrib.auth.models import User
5 from .models import Customer
6
7 class SignUpForm(UserCreationForm):
8     def __init__(self, *args, **kwargs):
9         super().__init__(*args, **kwargs)
10
11         self.fields['username'].widget.attrs.update({
12             'type': 'text',
13             'name': 'username',
14             'placeholder': 'Username',
15             'class': 'box',
16             'required': ''
17         })
18
19         self.fields['email'].widget.attrs.update({
20             'type': 'email',
21             'name': 'email',
22             'placeholder': 'Email',
23             'class': 'box',
24             'required': ''
25         })
26
27         self.fields['password1'].widget.attrs.update({
28             'type': 'password',
29             'name': 'password1',
30             'placeholder': 'Password',
31             'class': 'box',
32             'required': ''
33         })
34
35         self.fields['password2'].widget.attrs.update({
36             'type': 'password',
37             'name': 'password2',
38             'placeholder': 'Confirm Password'
39         })
```

Ln 51, Col 42 Spaces: 5 UTF-8 CRLF Python ⚡ Prettier ⚡ ENG 8:37 PM IN 1/19/2022 (3)

File Edit Selection View Go Run Terminal Help

forms.py - Buzzcars - Visual Studio Code

```
Buzzcars > src > forms.py > ContactForm > __init__.py
```

```
1 ...
2
3 class Meta:
4     model = User
5     fields = ['username', 'email', 'password1', 'password2']
6
7 class ContactForm(ModelForm):
8     def __init__(self, *args, **kwargs):
9         super().__init__(*args, **kwargs)
10
11         self.fields['first_name'].widget.attrs.update({
12             'type': 'text',
13             'name': 'first_name',
14             'placeholder': 'First Name',
15             'class': 'box',
16             'required': ''
17         })
18
19         self.fields['last_name'].widget.attrs.update({
20             'type': 'text',
21             'name': 'last_name',
22             'placeholder': 'Last Name',
23             'class': 'box',
24             'required': ''
25         })
26
27         self.fields['gender'].widget.attrs.update({
28             'type': 'text',
29             'name': 'first_name',
30             'placeholder': 'First Name',
31             'class': 'box',
32             'required': ''
33         })
34
35         self.fields['email'].widget.attrs.update({
36             'type': 'text',
37             'name': 'email',
38             'placeholder': 'Email',
39             'class': 'box',
40             'required': ''
41         })
```

Ln 63, Col 33 Spaces: 5 UTF-8 CRLF Python ⚡ Prettier ⚡ ENG 8:38 PM IN 1/19/2022 (3)

File Edit Selection View Go Run Terminal Help

forms.py - Buzzcars - Visual Studio Code

EXPLORER BUZZCARS

forms.py > Buzzcars > src > forms.py > ContactForm > __init__

```
74     'required': ''
75   })
76
77   self.fields['email'].widget.attrs.update({
78     'type': 'text',
79     'name': 'email',
80     'placeholder': 'Email',
81     'class': 'box',
82     'required': ''
83   })
84
85   self.fields['contact'].widget.attrs.update({
86     'type': 'text',
87     'name': 'contact',
88     'placeholder': 'contact',
89     'class': 'box',
90     'required': ''
91   })
92
93   self.fields['city'].widget.attrs.update({
94     'type': 'text',
95     'name': 'city',
96     'placeholder': 'city',
97     'class': 'box',
98     'required': ''
99   })
100
101  self.fields['state'].widget.attrs.update({
102    'type': 'text',
103    'name': 'state',
104    'placeholder': 'State',
105    'class': 'box',
106    'required': ''
107  })
108
109  self.fields['address'].widget.attrs.update({
110    'type': 'text',
111    'name': 'address'
112  })
113
114
115
116
117
118
119
120
121
```

Python 3.9.6 64-bit ('env': venv) 0 0 0 Rahul Live Share

Ln 63, Col 33 Spaces: 5 UTF-8 CRLF Python Prettier ENG 8:38 PM IN 1/19/2022

File Edit Selection View Go Run Terminal Help

forms.py - Buzzcars - Visual Studio Code

EXPLORER BUZZCARS

forms.py > Buzzcars > src > forms.py > ContactForm > __init__

```
97     'class': 'box',
98     'required': ''
99   })
100
101  self.fields['state'].widget.attrs.update({
102    'type': 'text',
103    'name': 'state',
104    'placeholder': 'State',
105    'class': 'box',
106    'required': ''
107  })
108
109  self.fields['address'].widget.attrs.update({
110    'type': 'text',
111    'name': 'address',
112    'placeholder': 'Address',
113    'class': 'box',
114    'required': ''
115  })
116
117
118
119
120
121
```

```
class Meta:
    model = Customer
    fields = ['first_name', 'last_name', 'gender', 'email', 'contact', 'city', 'state', 'address']
```

Python 3.9.6 64-bit ('env': venv) 0 0 0 Rahul Live Share

Ln 63, Col 33 Spaces: 5 UTF-8 CRLF Python Prettier ENG 8:38 PM IN 1/19/2022

File Edit Selection View Go Run Terminal Help

admin.py - Buzzcars - Visual Studio Code

EXPLORER BUZZCARS admin.py

```
BUZZCARS
  Buzzcars
  media
  src
    _pycache_
    migrations
    __init__.py
    admin.py
    apps.py
    forms.py
    models.py
    tests.py
    urls.py
    views.py
  static
  templates
  db.sqlite3
  manage.py
  env
  Synopsis.md
```

admin.py

```
from django.contrib import admin
from .models import Dealer, Vehicle, Customer, Sale, Tax

# Register your models here.
admin.site.register(Dealer)
admin.site.register(Vehicle)
admin.site.register(Customer)
admin.site.register(Sale)
admin.site.register(Tax)
```

OUTLINE

Python 3.9.6 64-bit ('env': venv) 0 0 0 Rahul Live Share

Ln 9, Col 25 Spaces: 4 UTF-8 CRLF Python Prettier ENG 8:38 PM IN 1/19/2022

File Edit Selection View Go Run Terminal Help

urls.py - Buzzcars - Visual Studio Code

EXPLORER BUZZCARS urls.py

```
BUZZCARS
  Buzzcars
  media
  src
    _pycache_
    migrations
    __init__.py
    admin.py
    apps.py
    forms.py
    models.py
    tests.py
    urls.py
    views.py
  static
  templates
  db.sqlite3
  manage.py
  env
  Synopsis.md
```

urls.py

```
from django.contrib import admin
from django.urls import path, include
from . import views
from django.conf import settings
from django.conf.urls.static import static

urlpatterns = [
    path('', views.Home, name='home'),
    path('about/', views.About, name='about'),
    path('login/', views.loginPage, name='login'),
    path('signup/', views.Signup, name='signup'),
    path('logout/', views.logoutUser, name='logout'),
    path('vehicle/', views.vehiclePage, name='vehicle'),
]
] + static(settings.MEDIA_URL, document_root=settings.MEDIA_ROOT)
```

OUTLINE

Python 3.9.6 64-bit ('env': venv) 0 0 0 Rahul Live Share

Ln 8, Col 36 (4 selected) Spaces: 4 UTF-8 CRLF Python Prettier ENG 8:38 PM IN 1/19/2022

views.py - Buzzcars - Visual Studio Code

```
BUZZCARS
  Buzzcars
  media
  src
    _pycache_
    migrations
    __init__.py
    admin.py
    apps.py
    forms.py
    models.py
    tests.py
    urls.py
    views.py
  static
  templates
  db.sqlite3
  manage.py
env
Synopsis.md
```

```
1  from django.shortcuts import render, redirect
2  from django.contrib.auth.models import User
3  from django.contrib import messages
4  from django.contrib.auth import authenticate, login, logout
5  from django.contrib.auth.forms import UserCreationForm
6  from .forms import SignUpForm, ContactForm
7  from .models import Vehicle
8
9  # Create your views here.
10
11
12
13 def home(request):
14     form = ContactForm()
15
16     if request.method == 'POST':
17         form = ContactForm(request.POST)
18
19         if form.is_valid():
20             form.save()
21
22     context = {'form': form}
23     return render(request, 'index.html', context)
24
25
26
27 def loginpage(request):
28
29     if request.method == 'POST':
30         username = request.POST.get('username')
31         password = request.POST.get('password')
32
33         try:
34             user = User.objects.get(username=username)
35         except:
36             messages.error(request, 'User does not exist!')
37
38         user = authenticate(request, username=username, password=password)
39
40         if user is not None:
41
42             login(request, user)
43             return redirect('home')
44
45     else:
46         messages.error(request, 'Username or password does not exist!')
47
48
49 def logoutuser(request):
50     logout(request)
51     return redirect('home')
52
53 def about(request):
54     return render(request, 'about.html')
55
56
57 def signup(request):
58     form = SignUpForm()
59
60     if request.method == 'POST':
61         form = SignUpForm(request.POST)
62
63         if form.is_valid():
64             user = form.save(commit=False)
65             user.save()
66             login(request, user)
67             return redirect('home')
68         else:
69             messages.error(request, 'User not created!')
70
71
72 def vehiclepage(request):
73     vehicles = Vehicle.objects.all()
74     context = {'vehicles': vehicles}
75
76     return render(request, 'vehicles.html', context)
```

Python 3.9.6 64-bit ('env': venv) 0 0 0 Rahul Live Share

LN 21, Col 1 Spaces: 4 UTF-8 CRLF Python Prettier ENG 8:38 PM IN 1/19/2022

views.py - Buzzcars - Visual Studio Code

```
BUZZCARS
  Buzzcars
  media
  src
    _pycache_
    migrations
    __init__.py
    admin.py
    apps.py
    forms.py
    models.py
    tests.py
    urls.py
    views.py
  static
  templates
  db.sqlite3
  manage.py
env
Synopsis.md
```

```
1  else:
2     messages.error(request, 'Username or password does not exist!')
3
4
5     return render(request, 'login.html')
6
7
8
9
10 def logoutuser(request):
11     logout(request)
12     return redirect('home')
13
14 def about(request):
15     return render(request, 'about.html')
16
17
18 def signup(request):
19     form = SignUpForm()
20
21     if request.method == 'POST':
22         form = SignUpForm(request.POST)
23
24         if form.is_valid():
25             user = form.save(commit=False)
26             user.save()
27             login(request, user)
28             return redirect('home')
29         else:
30             messages.error(request, 'User not created!')
31
32
33 def vehiclepage(request):
34     vehicles = Vehicle.objects.all()
35     context = {'vehicles': vehicles}
36
37     return render(request, 'vehicles.html', context)
```

Python 3.9.6 64-bit ('env': venv) 0 0 0 Rahul Live Share

LN 21, Col 1 Spaces: 4 UTF-8 CRLF Python Prettier ENG 8:38 PM IN 1/19/2022

File Edit Selection View Go Run Terminal Help

about.html - Buzzcars - Visual Studio Code

EXPLORER BUZZCARS

- Buzzcars
- index.html
- login.html
- signup.html
- vehicles.html
- db.sqlite3
- manage.py
- env
- Synopsis.md

about.html

```
1  <% load static %>
2  <!DOCTYPE html>
3  <html lang="en">
4  <head>
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8
9      <link rel="stylesheet" href="https://unpkg.com/swiper@7/swiper-bundle.min.css" />
10
11     <!-- font awesome cdn Link -->
12     <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css">
13
14     <!-- custom css file Link -->
15     <link rel="stylesheet" href="<% static 'css/style.css' %>">
16
17 </head>
18 <body>
19
20     <header class="header">
21         <div id="menu-btn" class="fas fa-bars"></div>
22
23         <a href="#" class="logo"> maxwheels </a>
24
25         <nav class="navbar">
26             <a href="<% url 'home' %>">home</a>
27             <a href="<% url 'about' %>">About Us</a>
28             <a href="<% url 'vehicle' %>">Vehicles</a>
29         </nav>
30     </header>
31
32     <div class="about-section">
33         <div class="about-containers">
34             <div class="content-section">
35                 <div class="title">
36                     <h1>About Us</h1>
37                     <div class="border"></div>
38                 </div>
39             </div>
40         </div>
41
42         <div class="content">
43             <p>Lorem ipsum dolor, sit amet consectetur adipisicing elit. Corrupti quod molestiae quae accusantium fuga? Consectetur aperi
44             </div>
45             <div class="skills-title">
46                 <h2>We Serve!</h2>
47             </div>
48             <div class="skills">
49                 <span>Quality services</span>
50                 <span>Premium Cars</span>
51                 <span>Quality Parts</span>
52             </div>
53             <div class="image-section">
54                 
55             </div>
56         </div>
57     </div>
58     </body>
59 </html>
```

Python 3.9.6 64-bit ('env': venv) 0 0 0 Rahul Live Share

Ln 21, Col 1 Spaces: 4 UTF-8 CRLF HTML Go Live Prettier ENG 8:39 PM IN 1/19/2022

File Edit Selection View Go Run Terminal Help

about.html - Buzzcars - Visual Studio Code

EXPLORER BUZZCARS

- Buzzcars
- index.html
- login.html
- signup.html
- vehicles.html
- db.sqlite3
- manage.py
- env
- Synopsis.md

about.html

```
1  </header>
2  <div class="about-section">
3      <div class="about-containers">
4          <div class="content-section">
5              <div class="title">
6                  <h1>About Us</h1>
7                  <div class="border"></div>
8              </div>
9              <div class="content">
10                 <p>Lorem ipsum dolor, sit amet consectetur adipisicing elit. Corrupti quod molestiae quae accusantium fuga? Consectetur aperi
11                 </div>
12                 <div class="skills-title">
13                     <h2>We Serve!</h2>
14                 </div>
15                 <div class="skills">
16                     <span>Quality services</span>
17                     <span>Premium Cars</span>
18                     <span>Quality Parts</span>
19                 </div>
20             </div>
21             <div class="image-section">
22                 
23             </div>
24         </div>
25     </div>
26     </body>
27 </html>
```

Python 3.9.6 64-bit ('env': venv) 0 0 0 Rahul Live Share

Ln 21, Col 1 Spaces: 4 UTF-8 CRLF HTML Go Live Prettier ENG 8:39 PM IN 1/19/2022

index.html - Buzzcars - Visual Studio Code

```
BUZZCARS > templates > index.html > ...
index.html
1 1% load static %
2 <!DOCTYPE html>
3 <html lang="en">
4 <head>
5   <meta charset="UTF-8">
6   <meta http-equiv="X-UA-Compatible" content="IE=edge">
7   <meta name="viewport" content="width=device-width, initial-scale=1.0">
8
9   <link rel="stylesheet" href="https://unpkg.com/swiper@7/swiper-bundle.min.css" />
10
11  <!-- font awesome cdn Link -->
12  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css">
13
14  <!-- custom css file Link -->
15  <link rel="stylesheet" href="{% static 'css/style.css' %}">
16
17 </head>
18 <body>
19
20 <header class="header">
21
22   <div id="menu-btn" class="fas fa-bars"></div>
23
24   <a href="#" class="logo"> <span>Buzz</span>cars </a>
25
26   <nav class="navbar">
27     <a href="{% url 'home' %}">home</a>
28     <a href="{% url 'vehicle' %}">vehicles</a>
29     <a href="#services">services</a>
30     <a href="{% url 'about' %}">About Us</a>
31     <a href="#contact">Contact Us</a>
32   </nav>
33
34   <div id="login-btn">
35     <div class="user-container">
36
37       1% if request.user.is_authenticated %>
38
39       <p class="user"><span>Hello {{request.user}} !</span></p>
40       <button class="btn"><a href="{% url 'logout' %}"><span>Logout</span></a></button>
41
42       &% else %
43       <button class="btn"><a href="{% url 'login' %}"><span>Login</span></a></button>
44
45       &% endif %
46     </div>
47
48     <i class="far fa-user"></i>
49   </div>
50
51   <div id="signup-btn">
52     <button class="btn"><a href="{% url 'signup' %}"><span>SignUp</span></a></button>
53     <i class="far fa-user"></i>
54   </div>
55
56 </header>
57
58 <section class="home" id="home">
59   <h3 data-speed="-2" class="home-parallax">find your car</h3>
60
61   
62
63   <a data-speed="7" href="{% url 'vehicle' %}" class="btn home-parallax">explore cars</a>
64
65 </section>
66
67 <section class="icons-container">
68   <div class="icons">
69     <i class="fas fa-home"></i>
70     <div class="content">
71       <h3>150+</h3>
72       <p>branches</p>
73     </div>
74   </div>
75
76 </section>
```

index.html - Buzzcars - Visual Studio Code

```
BUZZCARS > templates > index.html > ...
index.html
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
```

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer (Left):** Shows the project structure under "BUZZCARS". The "index.html" file is selected and highlighted.
- Editor (Center):** Displays the content of the "index.html" file. The code includes sections for icons, content, and a vehicle slider.
- Terminal (Bottom Left):** Shows the command "Rahul" entered.
- Status Bar (Bottom):** Provides information about the Python environment ("Python 3.9.6 64-bit (Env: venv)"), the current file ("index.html - Buzzcars - Visual Studio Code"), and the terminal session ("Rahul").
- Activity Bar (Bottom Right):** Includes icons for Share, Live Share, and Prettier.

```
<div class="icons">
    <i class="fas fa-car"></i>
    <div class="content">
        <h3>4770+</h3>
        <p>cars sold</p>
    </div>
</div>

<div class="icons">
    <i class="fas fa-users"></i>
    <div class="content">
        <h3>20+</h3>
        <p>happy clients</p>
    </div>
</div>

<div class="icons">
    <i class="fas fa-car"></i>
    <div class="content">
        <h3>1500+</h3>
        <p>news cars</p>
    </div>
</div>

</section>

<section class="vehicles" id="vehicles">

    <h1 class="heading"> popular <span>vehicles</span> </h1>

    <div class="swiper vehicles-slider">
        <div class="swiper-wrapper">
            <div class="swiper-slide box">
                
                <div class="content">
```

The screenshot shows the Visual Studio Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Explorer:** BUZZCARS folder containing: Buzzcars, media, src, static, templates (about.html, index.html, login.html, signup.html, vehicles.html), db.sqlite3, env, Synopsis.md.
- Editor:** The index.html file is open, displaying three car models with their details and a "Book a Quote" button.
- Bottom Status Bar:** Python 3.9.6 64-bit (env: venv), Rahul, Live Share, L1, Col 12, Spaces: 4, UTF-8, CRLF, HTML, Go Live, IN 1/19/2022, ENG 8:39 PM.

```

<div class="content">
  <h3>new model</h3>
  <div class="price"> <span>price : </span> Rs.28,62,000/- </div>
  <p>
    new
    <span class="fas fa-circle"></span> 2021
    <span class="fas fa-circle"></span> automatic
    <span class="fas fa-circle"></span> petrol
    <span class="fas fa-circle"></span> 183mph
  </p>
  <a href="#contact" class="btn">Book a Quote</a>
</div>

<div class="swiper-slide box">
  
  <div class="content">
    <h3>new model</h3>
    <div class="price"> <span>price : </span> Rs.30,62,000/- </div>
    <p>
      new
      <span class="fas fa-circle"></span> 2021
      <span class="fas fa-circle"></span> automatic
      <span class="fas fa-circle"></span> petrol
      <span class="fas fa-circle"></span> 183mph
    </p>
    <a href="#contact" class="btn">Book a Quote</a>
  </div>
</div>

<div class="swiper-slide box">
  
  <div class="content">
    <h3>new model</h3>
    <div class="price"> <span>price : </span> Rs.34,62,000/- </div>
    <p>
      new
    </p>
  </div>
</div>
```

File Edit Selection View Go Run Terminal Help index.html - Buzzcars - Visual Studio Code

EXPLORER BUZZCARS ... index.html x

Buzzcars > templates > index.html > ...

```
180     <span class="fas fa-circle">2021
181     <span class="fas fa-circle">automatic
182     <span class="fas fa-circle">petrol
183     <span class="fas fa-circle">183mph
184   </p>
185   <a href="#contact" class="btn">Book a Quote</a>
186 </div>
187 </div>

188 <div class="swiper-slide box">
189   
190   <div class="content">
191     <h3>new model</h3>
192     <div class="price"> <span>price : </span> Rs.29,62,000/- </div>
193     <p>
194       new
195       <span class="fas fa-circle">2021
196       <span class="fas fa-circle">automatic
197       <span class="fas fa-circle">petrol
198       <span class="fas fa-circle">183mph
199     </p>
200     <a href="#contact" class="btn">Book a Quote</a>
201   </div>
202 </div>

203 <div class="swiper-pagination"></div>
204 </div>
205 </section>
206
207 <section class="services" id="services">
208   <h1 class="heading"> our <span>services</span> </h1>
209   <div class="box-container">
210     <div class="swiper-pagination"></div>
211   </section>
212
213 <section class="services" id="services">
214   <h1 class="heading"> our <span>services</span> </h1>
215   <div class="box-container">
216     <div class="box">
217       <i class="fas fa-car"></i>
218       <h3>car selling</h3>
219       <p>Lorem, ipsum dolor sit amet consectetur adipisicing elit. Corporis, nisi.</p>
220       <a href="#" class="btn">read more</a>
221     </div>
222     <div class="box">
223       <i class="fas fa-tools"></i>
224       <h3>parts repair</h3>
225       <p>Lorem, ipsum dolor sit amet consectetur adipisicing elit. Corporis, nisi.</p>
226       <a href="#" class="btn">read more</a>
227     </div>
228     <div class="box">
229       <i class="fas fa-car-crash"></i>
230       <h3>car insurance</h3>
231       <p>Lorem, ipsum dolor sit amet consectetur adipisicing elit. Corporis, nisi.</p>
232       <a href="#" class="btn">read more</a>
233     </div>
234     <div class="box">
235       <i class="fas fa-car-battery"></i>
236       <h3>battery replacement</h3>
237       <p>Lorem, ipsum dolor sit amet consectetur adipisicing elit. Corporis, nisi.</p>
238       <a href="#" class="btn">read more</a>
239     </div>
240   </div>
241 </section>
242
243 <div class="box">
244   <h3>About Us</h3>
245   <p>We are a leading automobile company providing various services like car selling, parts repair, insurance, and battery replacement. Our mission is to provide quality services at competitive prices. We are committed to customer satisfaction and strive to exceed expectations. Join us today and experience the difference!</p>
246   <a href="#" class="btn">Read More</a>
247 </div>
```

Ln 1, Col 12 Spaces: 4 UTF-8 CRLF HTML ⚡ Go Live ✨ Prettier ⌂ ENG 8:39 PM IN 1/19/2022

File Edit Selection View Go Run Terminal Help index.html - Buzzcars - Visual Studio Code

EXPLORER BUZZCARS ... index.html x

Buzzcars > templates > index.html > ...

```
207   <div class="swiper-pagination"></div>
208 </div>
209 </section>
210
211 <section class="services" id="services">
212   <h1 class="heading"> our <span>services</span> </h1>
213   <div class="box-container">
214     <div class="box">
215       <i class="fas fa-car"></i>
216       <h3>car selling</h3>
217       <p>Lorem, ipsum dolor sit amet consectetur adipisicing elit. Corporis, nisi.</p>
218       <a href="#" class="btn">read more</a>
219     </div>
220     <div class="box">
221       <i class="fas fa-tools"></i>
222       <h3>parts repair</h3>
223       <p>Lorem, ipsum dolor sit amet consectetur adipisicing elit. Corporis, nisi.</p>
224       <a href="#" class="btn">read more</a>
225     </div>
226     <div class="box">
227       <i class="fas fa-car-crash"></i>
228       <h3>car insurance</h3>
229       <p>Lorem, ipsum dolor sit amet consectetur adipisicing elit. Corporis, nisi.</p>
230       <a href="#" class="btn">read more</a>
231     </div>
232     <div class="box">
233       <i class="fas fa-car-battery"></i>
234       <h3>battery replacement</h3>
235       <p>Lorem, ipsum dolor sit amet consectetur adipisicing elit. Corporis, nisi.</p>
236       <a href="#" class="btn">read more</a>
237     </div>
238   </div>
239 </section>
240
241 <div class="box">
242   <h3>About Us</h3>
243   <p>We are a leading automobile company providing various services like car selling, parts repair, insurance, and battery replacement. Our mission is to provide quality services at competitive prices. We are committed to customer satisfaction and strive to exceed expectations. Join us today and experience the difference!</p>
244   <a href="#" class="btn">Read More</a>
245 </div>
```

Ln 1, Col 12 Spaces: 4 UTF-8 CRLF HTML ⚡ Go Live ✨ Prettier ⌂ ENG 8:39 PM IN 1/19/2022

File Edit Selection View Go Run Terminal Help

index.html

```
Buzzcars > templates > index.html ...  
263 </section>  
264  
265  
266 <section class="newsletter">  
267  
268 <h3>subscribe for latest updates</h3>  
269 <p>Lorem ipsum dolor sit amet consectetur adipisicing elit, voluptatum, suscipit.</p>  
270  
271 <form action="">  
272 <input type="email" placeholder="enter your email">  
273 <input type="submit" value="subscribe">  
274 </form>  
275 </section>  
276  
277 <section class="contact" id="contact">  
278 <h1 class="heading"><span>contact</span> us</h1>  
279  
280 <div class="row">  
281  
282 <iframe class="map" src="https://www.google.com/maps/embed?pb=1m18!1m12!1m3!1d30153.788252261566!2d72.82321484621745!3d19.141690214" ...  
283  
284 <form method="POST" action="">  
285 <h3>Get in touch</h3>  
286 <% csrf_token %>  
287 <!-- <input type="text" placeholder="your name" class="box">  
288 <input type="email" placeholder="your email" class="box">  
289 <input type="tel" placeholder="subject" class="box">  
290 <textarea placeholder="your message" class="box" cols="30" rows="10"></textarea>  
291 <input type="submit" value="send message" class="btn"> -->  
292 {{form}}  
293 <input type="submit" value="send message" class="btn">  
294 </form>  
295  
296  
297  
298  
299 </div>  
300 </section>
```

Ln 1, Col 12 Spaces: 4 UTF-8 CRLF HTML ⚡ Go Live ✨ Prettier 🔍 ENG IN 8:40 PM 1/19/2022

File Edit Selection View Go Run Terminal Help

index.html

```
Buzzcars > templates > index.html ...  
303 <div class="box-container">  
304  
305  
306  
307 <div class="box">  
308 <h3>our branches</h3>  
309 <a href="#">  india </a>  
310 <a href="#">  japan </a>  
311 <a href="#">  france </a>  
312 <a href="#">  russia </a>  
313 <a href="#">  USA </a>  
314 </div>  
315  
316 <div class="box">  
317 <h3>quick links</h3>  
318 <a href="{% url 'home' %}">  home </a>  
319 <a href="{% url 'vehicle' %}">  vehicles </a>  
320 <a href="{% url 'services' %}">  services </a>  
321 <a href="{% url 'about' %}">  About Us </a>  
322 <a href="#contact">  Contact Us </a>  
323 </div>  
324  
325 <div class="box">  
326 <h3>contact info</h3>  
327 <a href="#">  9843273637 </a>  
328 <a href="#">  9637536687 </a>  
329 <a href="#">  saurav123@gmail.com </a>  
330 <a href="#">  Bangalore, India - 560083 </a>  
331 </div>  
332  
333 <div class="box">  
334 <h3>contact info</h3>  
335 <a href="#">  facebook </a>  
336 <a href="#">  twitter </a>  
337 <a href="#">  instagram </a>  
338 <a href="#">  linkedin </a>  
339 <a href="#">  pinterest </a>  
340 </div>
```

Ln 1, Col 12 Spaces: 4 UTF-8 CRLF HTML ⚡ Go Live ✨ Prettier 🔍 ENG IN 8:40 PM 1/19/2022

File Edit Selection View Go Run Terminal Help index.html - Buzzcars - Visual Studio Code

EXPLORER BUZZCARS

- Buzzcars
- media
- src
- static
- templates
- about.html
- index.html
- login.html
- signup.html
- vehicles.html

index.html x

```
266 <a href="#"> </i> 9843273637 </a>
267 <a href="#"> </i> 9637536687 </a>
268 <a href="#"> </i> saurav123@gmail.com </a>
269 <a href="#"> </i> Bangalore,India-560083 </a>
270 </div>
271 <div class="box">
272   <h3>contact info</h3>
273   <a href="#"> </i> facebook </a>
274   <a href="#"> </i> twitter </a>
275   <a href="#"> </i> instagram </a>
276   <a href="#"> </i> linkedin </a>
277   <a href="#"> </i> pinterest </a>
278 </div>
279 <div class="credit"> Front-end by Saurav Kumar | all rights reserved </div>
280 </section>
281 <script src="https://unpkg.com/swiper@7/swiper-bundle.min.js"></script>
282 <script src="{% static 'JS/script.js' %}"></script>
283 </body>
284 </html>
```

Python 3.9.6 64-bit ('env': venv) 0 0 0 Rahul Live Share

Ln 1, Col 12 Spaces: 4 UTF-8 CRLF HTML Go Live Prettier ENG IN 8:40 PM 1/19/2022 (3)

File Edit Selection View Go Run Terminal Help login.html - Buzzcars - Visual Studio Code

EXPLORER BUZZCARS

- Buzzcars
- media
- src
- static
- templates
- about.html
- index.html
- login.html
- signup.html
- vehicles.html

login.html x

```
1  {% load static %}
2
3  <!DOCTYPE html>
4  <html lang="en">
5    <head>
6      <meta charset="UTF-8">
7      <meta http-equiv="X-UA-Compatible" content="IE=edge">
8      <meta name="viewport" content="width=device-width, initial-scale=1.0">
9
10     <link rel="stylesheet" href="https://unpkg.com/swiper@7/swiper-bundle.min.css" />
11
12     <!-- font awesome cdn Link -->
13     <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css">
14
15     <!-- custom css file link -->
16     <link rel="stylesheet" href="{% static 'css/style.css' %}">
17
18   </head>
19   <body>
20     <section class="login_page">
21       <div class="imgbx">
22         
23       </div>
24       <div class="contentbx">
25         <div class="formbx">
26           <h2>Login</h2>
27           {% if messages %}
28             <ul class="messages">
29               {% for message in messages %}
30                 <p>{{ message }}</p>
31               {% endfor %}
32             </ul>
33           {% endif %}
34           <form method="POST" action="">
35             {% csrf_token %}
36             <div class="inputbx">
37               <span>Username</span>
38               <input type="text" name="username" placeholder="username" class="box" required>
39             </div>
40           </form>
41         </div>
42       </div>
43     </section>
44   </body>
45 </html>
```

Python 3.9.6 64-bit ('env': venv) 0 0 0 Rahul Live Share

Ln 29, Col 54 Spaces: 4 UTF-8 CRLF HTML Go Live Prettier ENG IN 8:40 PM 1/19/2022 (3)

File Edit Selection View Go Run Terminal Help

login.html - Buzzcars - Visual Studio Code

EXPLORER login.html

```
Buzzcars > templates > login.html > html > body > section.login_page > div.contentbx > div.formbx > ul.messages
  27
  28      <ul class="messages">
  29          {% for message in messages %}
  30              <p>{{ message }}</p>
  31          {% endfor %}
  32      </ul>
  33  {% endif %}
  34  <form method="POST" action="">
  35      {% csrf_token %}
  36      <div class="inputbx">
  37          <span>Username</span>
  38          <input type="text" name="username" placeholder="username" class="box" required>
  39      </div>
  40      <div class="inputbx">
  41          <span>Password</span>
  42          <input type="password" name="password" placeholder="password" class="box" required>
  43      </div>
  44      <div class="remember">
  45          <label><input type="checkbox" name="" id="">Remember me</label>
  46      </div>
  47      <div class="inputbx">
  48          <input type="submit" value="Sign in">
  49      </div>
  50      <div class="inputbx">
  51          <p>Or login with</p>
  52          <div class="buttons">
  53              <a href="#" class="btn">google</a>
  54              <a href="#" class="btn">facebook</a>
  55          </div>
  56      </div>
  57  </form>
  58  </div>
  59  </section>
  60  </body>
  61  </html>
```

Python 3.9.6 64-bit ('env': venv) 0 △ 0 Rahul Live Share

Ln 29, Col 54 Spaces: 4 UTF-8 CRLF HTML Go Live Prettier

File Edit Selection View Go Run Terminal Help

signup.html - Buzzcars - Visual Studio Code

EXPLORER signup.html

```
Buzzcars > templates > signup.html > html > body > section.login_page > div.contentbx > div.formbx
  1  {% load static %}
  2
  3  <!DOCTYPE html>
  4  <html lang="en">
  5      <head>
  6          <meta charset="UTF-8">
  7          <meta http-equiv="X-UA-Compatible" content="IE=edge">
  8          <meta name="viewport" content="width=device-width, initial-scale=1.0">
  9
 10         <link rel="stylesheet" href="https://unpkg.com/swiper@7/swiper-bundle.min.css" />
 11
 12         <!-- font awesome cdn link -->
 13         <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css" />
 14
 15         <!-- custom css file link -->
 16         <link rel="stylesheet" href="{% static 'css/style.css' %}" />
 17
 18     </head>
 19     <body>
 20         <section class="login_page">
 21             <div class="imgbx">
 22                 
 23             </div>
 24             <div class="contentbx">
 25                 <div class="formbx">
 26                     <h2>SIGNUP</h2>
 27                     {% if messages %}
 28                         <ul class="messages">
 29                             {% for message in messages %}
 30                                 <p>{{ message }}</p>
 31                             {% endfor %}
 32                         </ul>
 33                     {% endif %}
 34                     <form method="POST" action="">
 35                         {% csrf_token %}
 36                         <div class="inputbx">
 37                             <span>Name</span>
 38                             <input type="text" name="name" class="box" required>
 39                         </div>
 40                         <div class="inputbx">
 41                             <span>Email</span>
 42                             <input type="email" name="email" class="box" required>
 43                         </div>
 44                         <div class="inputbx">
 45                             <span>Password</span>
 46                             <input type="password" name="password" class="box" required>
 47                         </div>
 48                         <div class="inputbx">
 49                             <span>Confirm Password</span>
 50                             <input type="password" name="confirm_password" class="box" required>
 51                         </div>
 52                         <div class="remember">
 53                             <label><input type="checkbox" name="" id="">Remember me</label>
 54                         </div>
 55                         <div class="inputbx">
 56                             <input type="submit" value="Sign Up">
 57                         </div>
 58                     </form>
 59                 </div>
 60             </div>
 61         </section>
 62
 63
 64
 65
 66
 67
 68
 69
 70
 71
 72
 73
 74
 75
 76
 77
 78
 79
 80
 81
 82
 83
 84
 85
 86
 87
 88
 89
 90
 91
 92
 93
 94
 95
 96
 97
 98
 99
 100
 101
 102
 103
 104
 105
 106
 107
 108
 109
 110
 111
 112
 113
 114
 115
 116
 117
 118
 119
 120
 121
 122
 123
 124
 125
 126
 127
 128
 129
 130
 131
 132
 133
 134
 135
 136
 137
 138
 139
 140
 141
 142
 143
 144
 145
 146
 147
 148
 149
 150
 151
 152
 153
 154
 155
 156
 157
 158
 159
 160
 161
 162
 163
 164
 165
 166
 167
 168
 169
 170
 171
 172
 173
 174
 175
 176
 177
 178
 179
 180
 181
 182
 183
 184
 185
 186
 187
 188
 189
 190
 191
 192
 193
 194
 195
 196
 197
 198
 199
 200
 201
 202
 203
 204
 205
 206
 207
 208
 209
 210
 211
 212
 213
 214
 215
 216
 217
 218
 219
 220
 221
 222
 223
 224
 225
 226
 227
 228
 229
 230
 231
 232
 233
 234
 235
 236
 237
 238
 239
 240
 241
 242
 243
 244
 245
 246
 247
 248
 249
 250
 251
 252
 253
 254
 255
 256
 257
 258
 259
 260
 261
 262
 263
 264
 265
 266
 267
 268
 269
 270
 271
 272
 273
 274
 275
 276
 277
 278
 279
 280
 281
 282
 283
 284
 285
 286
 287
 288
 289
 290
 291
 292
 293
 294
 295
 296
 297
 298
 299
 300
 301
 302
 303
 304
 305
 306
 307
 308
 309
 310
 311
 312
 313
 314
 315
 316
 317
 318
 319
 320
 321
 322
 323
 324
 325
 326
 327
 328
 329
 330
 331
 332
 333
 334
 335
 336
 337
 338
 339
 340
 341
 342
 343
 344
 345
 346
 347
 348
 349
 350
 351
 352
 353
 354
 355
 356
 357
 358
 359
 360
 361
 362
 363
 364
 365
 366
 367
 368
 369
 370
 371
 372
 373
 374
 375
 376
 377
 378
 379
 380
 381
 382
 383
 384
 385
 386
 387
 388
 389
 390
 391
 392
 393
 394
 395
 396
 397
 398
 399
 400
 401
 402
 403
 404
 405
 406
 407
 408
 409
 410
 411
 412
 413
 414
 415
 416
 417
 418
 419
 420
 421
 422
 423
 424
 425
 426
 427
 428
 429
 430
 431
 432
 433
 434
 435
 436
 437
 438
 439
 440
 441
 442
 443
 444
 445
 446
 447
 448
 449
 450
 451
 452
 453
 454
 455
 456
 457
 458
 459
 460
 461
 462
 463
 464
 465
 466
 467
 468
 469
 470
 471
 472
 473
 474
 475
 476
 477
 478
 479
 480
 481
 482
 483
 484
 485
 486
 487
 488
 489
 490
 491
 492
 493
 494
 495
 496
 497
 498
 499
 500
 501
 502
 503
 504
 505
 506
 507
 508
 509
 510
 511
 512
 513
 514
 515
 516
 517
 518
 519
 520
 521
 522
 523
 524
 525
 526
 527
 528
 529
 530
 531
 532
 533
 534
 535
 536
 537
 538
 539
 540
 541
 542
 543
 544
 545
 546
 547
 548
 549
 550
 551
 552
 553
 554
 555
 556
 557
 558
 559
 560
 561
 562
 563
 564
 565
 566
 567
 568
 569
 570
 571
 572
 573
 574
 575
 576
 577
 578
 579
 580
 581
 582
 583
 584
 585
 586
 587
 588
 589
 590
 591
 592
 593
 594
 595
 596
 597
 598
 599
 600
 601
 602
 603
 604
 605
 606
 607
 608
 609
 610
 611
 612
 613
 614
 615
 616
 617
 618
 619
 620
 621
 622
 623
 624
 625
 626
 627
 628
 629
 630
 631
 632
 633
 634
 635
 636
 637
 638
 639
 640
 641
 642
 643
 644
 645
 646
 647
 648
 649
 650
 651
 652
 653
 654
 655
 656
 657
 658
 659
 660
 661
 662
 663
 664
 665
 666
 667
 668
 669
 670
 671
 672
 673
 674
 675
 676
 677
 678
 679
 680
 681
 682
 683
 684
 685
 686
 687
 688
 689
 690
 691
 692
 693
 694
 695
 696
 697
 698
 699
 700
 701
 702
 703
 704
 705
 706
 707
 708
 709
 710
 711
 712
 713
 714
 715
 716
 717
 718
 719
 720
 721
 722
 723
 724
 725
 726
 727
 728
 729
 730
 731
 732
 733
 734
 735
 736
 737
 738
 739
 740
 741
 742
 743
 744
 745
 746
 747
 748
 749
 750
 751
 752
 753
 754
 755
 756
 757
 758
 759
 760
 761
 762
 763
 764
 765
 766
 767
 768
 769
 770
 771
 772
 773
 774
 775
 776
 777
 778
 779
 780
 781
 782
 783
 784
 785
 786
 787
 788
 789
 789
 790
 791
 792
 793
 794
 795
 796
 797
 798
 799
 800
 801
 802
 803
 804
 805
 806
 807
 808
 809
 8010
 8011
 8012
 8013
 8014
 8015
 8016
 8017
 8018
 8019
 8020
 8021
 8022
 8023
 8024
 8025
 8026
 8027
 8028
 8029
 80210
 80211
 80212
 80213
 80214
 80215
 80216
 80217
 80218
 80219
 80220
 80221
 80222
 80223
 80224
 80225
 80226
 80227
 80228
 80229
 80230
 80231
 80232
 80233
 80234
 80235
 80236
 80237
 80238
 80239
 80240
 80241
 80242
 80243
 80244
 80245
 80246
 80247
 80248
 80249
 80250
 80251
 80252
 80253
 80254
 80255
 80256
 80257
 80258
 80259
 80260
 80261
 80262
 80263
 80264
 80265
 80266
 80267
 80268
 80269
 80270
 80271
 80272
 80273
 80274
 80275
 80276
 80277
 80278
 80279
 80280
 80281
 80282
 80283
 80284
 80285
 80286
 80287
 80288
 80289
 80290
 80291
 80292
 80293
 80294
 80295
 80296
 80297
 80298
 80299
 802100
 802101
 802102
 802103
 802104
 802105
 802106
 802107
 802108
 802109
 802110
 802111
 802112
 802113
 802114
 802115
 802116
 802117
 802118
 802119
 8021100
 8021110
 8021120
 8021130
 8021140
 8021150
 8021160
 8021170
 8021180
 8021190
 80211000
 80211100
 80211200
 80211300
 80211400
 80211500
 80211600
 80211700
 80211800
 80211900
 80211001
 80211101
 80211201
 80211301
 80211401
 80211501
 80211601
 80211701
 80211801
 80211901
 80211002
 80211102
 80211202
 80211302
 80211402
 80211502
 80211602
 80211702
 80211802
 80211902
 80211003
 80211103
 80211203
 80211303
 80211403
 80211503
 80211603
 80211703
 80211803
 80211903
 80211004
 80211104
 80211204
 80211304
 80211404
 80211504
 80211604
 80211704
 80211804
 80211904
 80211005
 80211105
 80211205
 80211305
 80211405
 80211505
 80211605
 80211705
 80211805
 80211905
 80211006
 80211106
 80211206
 80211306
 80211406
 80211506
 80211606
 80211706
 80211806
 80211906
 80211007
 80211107
 80211207
 80211307
 80211407
 80211507
 80211607
 80211707
 80211807
 80211907
 80211008
 80211108
 80211208
 80211308
 80211408
 80211508
 80211608
 80211708
 80211808
 80211908
 80211009
 80211109
 80211209
 80211309
 80211409
 80211509
 80211609
 80211709
 80211809
 80211909
 80211010
 80211110
 80211210
 80211310
 80211410
 80211510
 80211610
 80211710
 80211810
 80211910
 80211011
 80211111
 80211211
 80211311
 80211411
 80211511
 80211611
 80211711
 80211811
 80211911
 80211012
 80211112
 80211212
 80211312
 80211412
 80211512
 80211612
 80211712
 80211812
 80211912
 80211013
 80211113
 80211213
 80211313
 80211413
 80211513
 80211613
 80211713
 80211813
 80211913
 80211014
 80211114
 80211214
 80211314
 80211414
 80211514
 80211614
 80211714
 80211814
 80211914
 80211015
 80211115
 80211215
 80211315
 80211415
 80211515
 80211615
 80211715
 80211815
 80211915
 80211016
 80211116
 80211216
 80211316
 80211416
 80211516
 80211616
 80211716
 80211816
 80211916
 80211017
 80211117
 80211217
 80211317
 80211417
 80211517
 80211617
 80211717
 80211817
 80211917
 80211018
 80211118
 80211218
 80211318
 80211418
 80211518
 80211618
 80211718
 80211818
 80211918
 80211019
 80211119
 80211219
 80211319
 80211419
 80211519
 80211619
 80211719
 80211819
 80211919
 80211020
 80211120
 80211220
 80211320
 80211420
 80211520
 80211620
 80211720
 80211820
 80211920
 80211021
 80211121
 80211221
 80211321
 80211421
 80211521
 80211621
 80211721
 80211821
 80211921
 80211022
 80211122
 80211222
 80211322
 80211422
 80211522
 80211622
 80211722
 80211822
 80211922
 80211023
 80211123
 80211223
 80211323
 80211423
 80211523
 80211623
 80211723
 80211823
 80211923
 80211024
 80211124
 80211224
 80211324
 80211424
 80211524
 80211624
 80211724
 80211824
 80211924
 80211025
 80211125
 80211225
 80211325
 80211425
 80211525
 80211625
 80211725
 80211825
 80211925
 80211026
 80211126
 80211226
 80211326
 80211426
 80211526
 80211626
 80211726
 80211826
 80211926
 80211027
 80211127
 80211227
 80211327
 80211427
 80211527
 80211627
 80211727
 80211827
 80211927
 80211028
 80211128
 80211228
 80211328
 80211428
 80211528
 80211628
 80211728
 80211828
 80211928
 80211029
 80211129
 80211229
 80211329
 80211429
 80211529
 80211629
 80211729
 80211829
 80211929
 80211030
 80211130
 80211230
 80211330
 80211430
 80211530
 80211630
 80211730
 80211830
 80211930
 80211031
 80211131
 80211231
 80211331
 80211431
 80211531
 80211631
 80211731
 80211831
 80211931
 80211032
 80211132
 80211232
 80211332
 80211432
 80211532
 80211632
 80211732
 80211832
 80211932
 80211033
 80211133
 80211233
 80211333
 80211433
 80211533
 80211633
 80211733
 80211833
 80211933
 80211034
 80211134
 80211234
 80211334
 80211434
 80211534
 80211634
 80211734
 80211834
 80211934
 80211035
 80211135
 80211235
 80211335
 80211435
 80211535
 80211635
 80211735
 80211835
 80211935
 80211036
 80211136
 80211236
 80211336
 80211436
 80211536
 80211636
 80211736
 80211836
 80211936
 80211037
 80211137
 80211237
 80211337
 80211437
 80211537
 80211637
 80211737
 80211837
 80211937
 80211038
 80211138
 80211238
 80211338
 80211438
 80211538
 80211638
 80211738
 80211838
 80211938
 80211039
 80211139
 80211239
 80211339
 80211439
 80211539
 80211639
 80211739
 80211839
 80211939
 80211040
 80211140
 80211240
 80211340
 80211440
 80211540
 80211640
 80211740
 80211840
 80211940
 80211041
 80211141
 80211241
 80211341
 80211441
 80211541
 80211641
 80211741
 80211841
 80211941
 80211042
 80211142
 80211242
 80211342
 80211442
 80211542
 80211642
 80211742
 80211842
 80211942
 80211043
 80211143
 80211243
 80211343
 80211443
 80211543
 80211643
 80211743
 80211843
 80211943
 80211044
 80211144
 80211244
 80211344
 80211444
 80211544
 80211644
 80211744
 80211844
 80211944
 80211045
 80211145
 80211245
 80211345
 80211445
 80211545
 80211645
 80211745
 80211845
 80211945
 80211046
 80211146
 80211246
 80211346
 80211446
 80211546
 80211646
 80211746
 80211846
 80211946
 80211047
 80211147
 80211247
 80211347
 80211447
 80211547
 80211647
 80211747
 80211847
 80211947
 80211048
 80211148
 80211248
 80211348
 80211448
 80211548
 80211648
 80211748
 80211848
 80211948
 80211049
 80211149
 80211249
 80211349
 80211449
 80211549
 80211649
 80211749
 80211849

```

File Edit Selection View Go Run Terminal Help

signup.html - Buzzcars - Visual Studio Code

BUZZCARS

- BUZZCARS
 - > Buzzcars
 - > media
 - > src
 - > static
 - templates
 - > about.html
 - > index.html
 - > login.html
 - > signup.html
 - > vehicles.html
 - db.sqlite3
 - manage.py
 - > env
 - Synopsis.md

signup.html

```
28     <ul class="messages">
29         {% for message in messages %}
30             <p>{{ message }}</p>
31         {% endfor %}
32     {% endif %}
33     <form method='POST' action="">
34         {% csrf_token %}
35         <div class="inputbx">
36             <span>Name</span>
37             {{ form.username }}
38         </div>
39         <div class="inputbx">
40             <span>Email</span>
41             {{ form.email }}
42         </div>
43         <div class="inputbx">
44             <span>Password</span>
45             {{ form.password1 }}
46         </div>
47         <div class="inputbx">
48             <span>Confirm Password</span>
49             {{ form.password2 }}
50         </div>
51         <div class="inputbx">
52             <input type="submit" value="Sign Up">
53         </div>
54     <div class="txt"><h4>include digits in your password</h4></div>
55
56     </form>
57
58 </div>
59 </section>
60 </body>
61 </html>
```

Python 3.9.6 64-bit ('env': venv) 0 0 0 Rahul Live Share

File Edit Selection View Go Run Terminal Help

vehicles.html - Buzzcars - Visual Studio Code

BUZZCARS

- BUZZCARS
 - > Buzzcars
 - > media
 - > src
 - > static
 - templates
 - > about.html
 - > index.html
 - > login.html
 - > signup.html
 - > vehicles.html
 - db.sqlite3
 - manage.py
 - > env
 - Synopsis.md

vehicles.html

```
1  {% load static %}
2  <!DOCTYPE html>
3  <html lang="en">
4      <head>
5          <meta charset="UTF-8">
6          <meta http-equiv="X-UA-Compatible" content="IE=edge">
7          <meta name="viewport" content="width=device-width, initial-scale=1.0">
8              Follow link (ctrl + click)
9          <link rel="stylesheet" href="https://unpkg.com/swiper@7/swiper-bundle.min.css" />
10
11         <!-- font awesome cdn link -->
12         <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css" />
13
14         <!-- custom css file link -->
15         <link rel="stylesheet" href="{% static 'css/style.css' %}" />
16
17     </head>
18     <body>
19         <header class="header">
20             <div id="menu-btn" class="fas fa-bars"></div>
21             <a href="#" class="logo"> Buzz <span>cars </span> </a>
22
23             <nav class="navbar">
24                 <a href="{% url 'home' %}"> home </a>
25                 <a href="{% url 'vehicle' %}"> vehicles </a>
26                 <a href="{% url 'home' %}#services"> services </a>
27                 <a href="{% url 'about' %}"> About Us </a>
28                 <a href="{% url 'home' %}#contact"> Contact Us </a>
29             </nav>
30
31         </header>
32
33         <div class="grid-container">
34             {% for vehicle in vehicles %}
35                 <div class="grid-item">
36                     <div class="card">
37                         
38                         <div class="card__content">
```

Python 3.9.6 64-bit ('env': venv) 0 0 0 Rahul Live Share

File Edit Selection View Go Run Terminal Help

EXPLORER BUZZCARS

vehicles.html

```
31     </nav>
32   </header>
33   <div class="grid-container">
34     {% for vehicle in vehicles %}
35       <div class="grid-item">
36         <div class="card">
37           
38           <div class="card__content">
39             <h1 class="card__title">{{vehicle.name}}</h1>
40             <h3 class="card__title"><span class="fas fa-circle"></span> {{vehicle.type}} <span class="fas fa-circle"></span> {{vehicle.year}}</h3>
41             <p>{{vehicle.description}}</p>
42             <div class="card__price">Rs.{{vehicle.cost}}</div>
43             <div class="card__content"><a href="{{ url 'home' }}#contact" class="btw">Book a Quote</a></div>
44           </div>
45         </div>
46       </div>
47     {% endfor %}
48   </div>
49 <script src="https://unpkg.com/swiper@7/swiper-bundle.min.js"></script>
50 <script src="{{ static 'JS/script.js' }}"></script>
51 </body>
52 </html>
```

OUTLINE

Python 3.9.6 64-bit ('env': venv) 0 0 0 Live Share

Ln 13, Col 1 Spaces: 4 UTF-8 CRLF HTML Go Live Prettier ENG 8:40 PM IN 1/19/2022