## Web Application Revision Practice 5 [YIJC/2023/Prelim/P2/Task 4]

A taxi company rents its vehicles to the drivers. It uses a database Taxi.db that has four tables: a table to store data about the drivers, a table for the vehicle data, a table for the data on vehicle rental, and a table for the drivers' feedback.

The fields in each table are:

#### Driver:

- ID unique identification number, for example, 1012
- Name the driver's name
- Type the type of driver, for example, Full Time or Part Time.

#### Vehicle:

- License the unique license plate number of the vehicle
- Model model of the vehicle
- Cost the cost to rent the vehicle per day
- MaxPassenger the maximum number of passengers it can carry.

#### Rent:

- DriverID the driver's unique identification number
- License the unique license plate number of the vehicle
- Date the date of rental
- Paid the status of payment for the rental, for example, Yes or No.

#### Feedback:

- ID an auto-generared unique identification number
- DriverID the driver's unique identification number
- Date the date when the feedback was received
- Comment the feedback from the driver.

#### **Task 4.1**

Write a Python program that uses SQL code to create the table Feedback in the database Taxi.db. Define the primary key and foreign key for the table.

The text file Feeback.txt stores the comma-separated values for the Feedback table in the database. Write a Python program to read in the data from the text file and store them in the database.

[6]

## Save your program code as:

Task4 1 <your name> <centre number> <index number>.py

#### **Task 4.2**

Write a Python program and the necessary files to create a web application.

The program in the default route renders the index.html to display the following form for the taxi driver to enter and submit his driver's ID.

# Taxi Rental Database

To book a vehicle, enter Driver ID:	
Submit	<u> </u>

### Save your Python program as:

Task4 2 <your name> <centre number> <index number>.py

## with any additional files/subfolders as needed in a folder named

Task4 2 <your name> <centre number> <index number>.

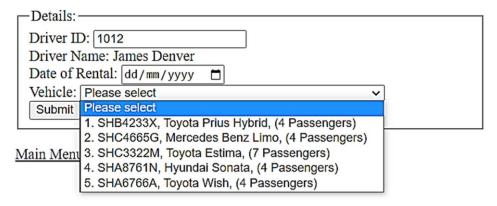
[3]

### **Task 4.3**

Write Python program for the /check route to:

- read the driver's ID from the form submitted by the driver
- perform a query check on the database for the driver's name
- render the rental.html to display a form for the driver to select the rental date and vehicle as shown in the following:

# **Taxi Rental:**



#### Save your Python program as:

Task4\_3\_<your name>\_<centre number>\_<index number>.py

## with any additional files/subfolders as needed in a folder named

Task4 3 <your name> <centre number> <index number>.

Run the web application with the driver's ID 1012.

### Save the webpage output as:

Task4 3 <your name> <centre number> <index number>.html

#### **Task 4.4**

Write Python program for the /rental route to:

- · read the driver's ID, the date and the selected vehicle from the form submitted by the driver
- perform a query check on the database for the vehicle's availability on the selected date
  - if it is unavailable, render a html page to display the message:

"Vehicle unavailable for the selected date."

• if it is available, record the details in the database's table Rent with the field Paid indicated as "No"; thereafter, render a html page to display the message:

"Rental Successful".

## Save your Python program as:

Task4 4 <your name> <centre number> <index number>.py

# with any additional files/subfolders as needed in a folder named

Task4 4 <your name> <centre number> <index number>.

[5]

### **Task 4.5**

Modify the program code for the / rental route in Task 4.4 to:

- perform a query on the database for the driver's outstanding unpaid rentals
- compute the outstanding total rental owed
- render the success.html to display a table as shown in the following:

#### Rental Successful

#### **Current Rental Records**

Date	Vehicle Model	Rental	Paid
2023-08-26	Toyota Estima	105.0	Yes
2023-08-27	Toyota Estima	105.0	Yes
2023-08-28	Toyota Prius Hybrid	90.0	No
2023-08-30	Toyota Prius Hybrid	90.0	No

Outstanding amount of rental owed: \$180.00 Main Menu

# Save your Python program as:

Task4\_5\_<your name>\_<centre number>\_<index number>.py

## with any additional files/subfolders as needed in a folder named

Task4 5 <your name> <centre number> <index number>.

Run the web application with the driver (ID 1012) renting the vehicle SHB4233X Toyota Prius Hybrid on 30th August 2023.

#### Save the webpage output as:

Task4 5 <your name> <centre number> <index number>.html

[7]