IE5600 – Applied Programming for Industrial Systems

AY 2022/23 Semester 1

Individual Assignment 1 - PyCar

Objectives

At the completion of the individual assignment, you should:

1. Develop a better understanding of how to apply the computational problem solving process to a moderately complex problem.

2. Implement imperative and structured programming paradigms in Python.

3. Apply appropriate control flow statements to implement algorithm.

4. Apply appropriate multi-dimensional data structures to implement algorithm.

General Requirements

You are required to develop a Python program known as **PyCar** to manage the business operations of a car rental company with three outlets. Some of the essential business operations include fleet management, rental reservations and rental operations (i.e., pickup and return of rental cars).

The company currently operates the following three outlets that is opened daily but only during the respective operating hours:

- Outlet A – 9 am to 6 pm
- Outlet B – 9 am to 6 pm
- Outlet C – 8 am to 8 pm

Each car in the company's fleet is associated with a particular make and model, e.g., Toyota Corolla, and categorised into one of three categories – Sedan, SUV and MPV. A car is available for rental during a particular period if it is not allocated to an existing reservation or not under maintenance. Rental fees are typically quoted on a daily basis for a contiguous rental period of 24 hours. That is, if the rental duration is 24 hours or any part thereof, the customer will be charged for one day of rental. The daily rental rate for each category of cars is fixed as follows:

- Sedan – $100/day
- SUV – $150/day
- MPV – $200/day

When renting a car, customers can only specify the required car category, pickup date/time, return date/time, pickup outlet and return outlet. In other words, customers can be allocated any car of any make and model as long as the car is of the correct category. For the location options, customers can specify the return outlet to be the same as the pickup outlet or a different outlet, subject to the operating hours of the respective outlet.

In order to maximise business opportunities, the company allows a customer to reserve a car that is last returned to a different outlet as long as the car can be moved to the required pickup outlet in time. To facilitate a timely pickup, a minimum transit time of 2 hours is enforced. The company will assign an employee to drive the car from the last returned outlet to the new pickup outlet. For example, it should be possible for a customer to rent a car for pickup at Outlet A on 3 October 12 pm that is last returned to outlet B on 3 October 10 am. The company will assign an employee to drive the car from Outlet B to Outlet A at an appropriate time between 10 am and 12 noon taking into consideration the travel duration and pickup time.

PyCar is also responsible for the actual allocation of cars to fulfil the rental reservations at each outlet on a daily basis. When performing car allocation, it is important to take into consideration the car category of each reservation and the pickup location. As part of the car allocation process, PyCar also needs to generate the required transit records so that the company can assign its employees to move cars from one outlet to another.

Use Cases

Implement the following use cases for PyCar without the use of any predefined Python modules other than those specified below:

- `datetime`

Your program should **NOT** contain any other `import` statement unless you are importing your own user-defined module(s).

S/N	Use Case	Use Case Description/Business Rules
1	**Load Initial Data** (1 mark)	• Auto load the initial data given in Appendix A • You required to use an appropriate data structure to represent the initial data in-memory. • It is not necessary to save the data to a file.
2	**Add Car** (2 mark)	• Add a new car. • If the license plate number already exists, print out an error message. • Perform input data validation on category, status and outlet. Output an appropriate error message whenever necessary. Sample Input: • Input license plate number • Input make • Input model

S/N	Use Case	Use Case Description/Business Rules
		• Input category – Sedan, SUV, MPV • Input status – Available, Allocated, Pickup, Maintenance • Input outlet – Outlet A, Outlet B, Outlet C

3	**Reserve Car**
	(3 mark)
		• Prompt user to input customer name and the required options.
		• Perform a check to determine whether the existing fleet of cars is able to cater to the required reservation.
		• The reservation check should take into consideration:
		  o Only cars that are not under maintenance are rentable.
		  o Current, i.e., last return, outlet of a car.
		  o Existing reservations in PyCar.
		  o Cars are only allocated on the day of pickup.
		  o The operating hours of the pickup outlet and return outlet.
		• Output the availability of the required car category.
		• If the required car category is available:
		  o Calculate and output the rental fee.
		  o Prompt user to confirm the reservation.
		• If user confirms the reservation, save the reservation data into an appropriate data structure in-memory and output a suitable numerical reservation number.
		
		Sample Input:
		• Input customer name
		• Input car category
		• Input pickup date/time
		• Input return date/time
		• Input pickup outlet
		• Input return outlet
		
		Sample Output:
		• Output availability – Yes/No
		• Output rental fees if available
		• Confirm reservation – Yes/No
		
		Sample Use Cases based on Initial Data:
		• Alice, Sedan, 03/10/22 08:00, 05/10/22 06:00, Outlet A, Outlet A – Not Available (Outside Outlet Operating Hours)
		• Bob, Sedan, 03/10/22 09:00, 05/10/22 9:00, Outlet A, Outlet A – Available, $200 and Reserved with #1
		• Charles, SUV, 03/10/22 09:00, 05/10/22 16:00, Outlet C, Outlet C – Available, $450 and Reserved with #2
		• Deborah, SUV, 04/10/22 09:00, 05/10/22 09:00, Outlet C, Outlet C – Not Available

S/N	Use Case	Use Case Description/Business Rules
		• Edward, SUV, 05/10/22 18:00, 07/10/22 18:00, Outlet A, Outlet A – Available (Sufficient Transit Time), $300 and Reserved with #3 • Felix, Sedan, 03/10/22 09:00, 04/10/22 9:00, Outlet A, Outlet A – Available, $100 and Reserved with #4

4	**Allocate Car**
	(2 mark)
		• In real-world business operation, this use case would perform car allocation for reservations that are scheduled for current day pickup.
		• For the purpose of this assignment, PyCar should prompt user to input a required date and then perform car allocation for reservations that are scheduled for pickup on that date.
		• The status of allocated cars should be set to Allocated. If a car's status is currently Pickup, the allocation status should be set to "Pickup and Allocated".
		• Car allocation results should be saved in an appropriate data structure in-memory.
		• You may assume that this use case is executed in strict chronological order once started, i.e., no intermediate date would be skipped and no date earlier than the first date would be inputted.
		• Output a list of car allocations scheduled for pickup on the inputted date.
		
		Sample Input:
		• Input date
		
		Sample Output:
		• Output reservation number.
		• Output pickup location.
		• Output car license plate number.
		• Output transit information if required:
		  ○ Source outlet.
		
		Sample Use Cases based on Reserve Car:
		• Input 03/10/22:
		  ○ #1, Outlet A, SE001A
		  ○ #2, Outlet C, SU002C
		  ○ #4, Outlet A, SE002A
		• Input 04/10/22:
		  ○ No allocation is required
		• Input 05/10/22:
		  ○ #3, Outlet A, SU002C, Transit from Outlet C

5	**Pickup Car** (1 mark)	• Prompt user to input reservation number. • If the reservation number does not exist or a car has not been allocated, output an error message. • Otherwise, output the license plate number of the allocated car and set its status to Pickup.

S/N	Use Case	Use Case Description/Business Rules
		Sample Use Cases based on Allocate Car: • #1: ○ SE001A
6	**Return Car** (1 mark)	• Prompt user to input reservation number. • If the reservation number does not exist, a car has not been allocated, or the allocated car has not been pickup, output an error message. • Otherwise, output the license plate number of the allocated car and set its status to Available. Sample Use Cases based on Pickup Car: • #1 ○ SE001A

Deliverable Submission

The assignment deliverable to be submitted to the Canvas Assignments tool are to be placed in a single zip archive file with the following folders structure:

- **source** subfolder containing:
  - All Python source files that constitute your program.
  - The main source file containing the program entry point should be named as pycar.py, i.e., your program should be runnable with the command `python pycar.py`

Upload this zip archive file to the designated Canvas Assignment: Assignments > Individual Assignment 1.

Your deliverables must be submitted latest by **Sunday, 23 October 2022, 11:59 pm**. No submission will be accepted for assessment after this date/time and you will be awarded **0** marks.

-- End of Assignment Specification --

Appendix A – Initial Data

Outlet

OutletName, OpeningHour, ClosingHour
Outlet A, 9:00, 18:00
Outlet B, 9:00, 18:00
Outlet C, 8:00, 20:00

Car

LicensePlateNumber, Make, Model, Category, Status, Outlet
SE001A, Toyota, Corolla, Sedan, Available, Outlet A
SE002A, Toyota, Corolla, Sedan, Available, Outlet A
SE003A, Toyota, Corolla, Sedan, Maintenance, Outlet A
SE001B, Honda, Civic, Sedan, Available, Outlet B
SE002B, Honda, Civic, Sedan, Available, Outlet B
SE003B, Honda, Civic, Sedan, Maintenance, Outlet B
SE001C, Kia, Cerato, Sedan, Available, Outlet C
SU002C, Subaru, Forrester, SUV, Available, Outlet C
MP003C, Honda, Odyssey, MPV, Available, Outlet C