

PARKING LOT SYSTEM

Contents:

- Description and Features
- Technology and concepts used
- Requirements
- Setup
- Instructions on how to run the program

Description and Features:

This project contains basic implementation of Parking lot system using Python. It creates parking lot with given number of slots. The program has following features:

- Create a parking lot with given capacity and name
- Park a car
- Leave parking lot
- Check the status of parking lot
- Get the nearest empty slot
- Get registration numbers of all cars of a particular colour
- Get the slot numbers of all cars of a particular colour
- Get the slot number of car by registration number
- Get total number of empty slots available at the moment

Inputs to run the program can be provided in three different mode,

1. Using a text file
2. User interactive – by typing the commands manually
3. User interactive – by choosing options in inbuilt Menu

Technology and concepts used:

- Python 3.8 (Code is made compatible to run in python 2 as well)
- OOPS
- User defined Modules

Requirements:

- User needs to install Python in their system, use the following link to download and install the latest version of python: <https://www.python.org/downloads/>

Setup:

Follow the below steps to execute or run this project in your system:

1. Clone the repository (<https://github.com/attainu/project-akshay-sahu-au13.git>)
2. Install Python in your computer
3. Open the folder using shell or VS code/IDE and type: `python main.py` to run it as a user interactive program
 - Press 1 for menu based interactive interface
 - Press 2 or any other key to type the commands manually
4. To give inputs from a file, type following command in shell:
`python main.py -run input_file1.txt`

Instructions on how to run the program:

➤ To run with file input:

- Open the folder containing source code files in shell and type:
`python main.py -run input_file1.txt` (or)
`python main.py -run input_file2.txt`
- Input files have the all the commands to test the features of Parking lot program, it will generate following output the in the shell window:

```
C:\Users\Akshay\OneDrive\Desktop\My Projects\Python Project\Attainu_Py_Project>python main.py -run input_file1.txt
Parking lot Park_Safe created with capacity of 6 vehicles
Please park your car in allocated slot : 1
Please park your car in allocated slot : 2
Please park your car in allocated slot : 3
Please park your car in allocated slot : 4
Please park your car in allocated slot : 5
Please park your car in allocated slot : 6
Vacated slot number 4
Slot No.      Registration No.      Colour
1             KA-01-HH-1234          White
2             KA-01-HH-9999          White
3             KA-01-BB-0001          Black
4             ((EMPTY))              ((EMPTY))
5             KA-01-HH-2701          Blue
6             KA-01-HH-3141          Black
Please park your car in allocated slot : 4
Parking lot is full, can't accomodate more vehicles... SORRY!
KA-01-HH-1234, KA-01-HH-9999, KA-01-P-3331
1, 2, 4
The car with registration no. KA-01-HH-3141 is parked in slot 6
No such car is Parked here
```

➤ To run with User driven input:

- Open the folder containing source code files in shell and type following command: `python main.py`, then **press any key except 1** after the prompt.
- Now user has to type the commands manually in following format:
 - To create parking lot of size N and name XYZ:
→ `create_parking_lot N XYZ`
 - To park a car:
→ `park reg_no color`
 - To leave/vacate the slot:
→ `leave slot_no`
 - To see the status of parking lot:
→ `view_parked_cars`
 - To get Registration numbers of cars of same color:
→ `registration_numbers_for_cars_with_colour White`
 - To get slot numbers for cars of same color:
→ `slot_numbers_for_cars_with_colour White`
 - To get the nearest empty slot:
→ `nearest_empty_slot`
 - To check the total available empty slots:
→ `remaining_slots`
 - To get slot number for car by registration no.:
→ `slot_number_for_registration_number KA-01-HH-7777`
 - To exit the program:
`exit`

```
C:\Windows\system32\cmd.exe
C:\Users\Akshay\OneDrive\Desktop\My Projects\Python_Project\Attainu_Py_Project>python main.py
Press 1 for menu driven inputs:
Press any other key for normal user driven inputs: ②
==> create_parking_lot 6 Park Safe
Parking lot Park Safe created with capacity of 6 vehicles
==> park KA01AA7788 Red
Please park your car in allocated slot : 1
==> park KA01AA1010 White
Please park your car in allocated slot : 2
==> park KA01AA2555 White
Please park your car in allocated slot : 3
==> view_parked_cars
Slot No.      Registration No.      Colour
1             KA01AA7788            Red
2             KA01AA1010            White
3             KA01AA2555            White
4             ((EMPTY))             ((EMPTY))
5             ((EMPTY))             ((EMPTY))
6             ((EMPTY))             ((EMPTY))
==> leave 1
Vacated slot number 1
==> slot_numbers_for_cars_with_colour White
2, 3
==> registration_numbers_for_cars_with_colour White
KA01AA1010, KA01AA2555
==> nearest_empty_slot
Nearest available slot is 1
==> exit
Thank you, for using our Parking station, Visit Again!!!
```

➤ To run with Menu driven input:

- Open the folder containing source code files in shell and type:
`python main.py`, then **press 1** after the prompt to see the **menu**.

```
C:\Windows\system32\cmd.exe - python main.py
C:\Users\Akshay\OneDrive\Desktop\My Projects\Python_Project\Attainu_Py_Project>python main.py
Press 1 for menu driven inputs:
Press any other key for normal user driven inputs: 1

Please find the MENU below:

Press 0 to create parking lot
Press 1 to park your car
Press 2 to leave the parking lot
Press 3 to know car slot by reg_no.
Press 4 to know all the slots for cars of particular color
press 5 to get the nearest empty slot
Press 6 to view the current status of Parking lot
Press 7 to view reg_nos for cars of particular color
Press 8 to know the no. of empty slots
Type MENU/menu to see the Menu at any point of time
Type Exit or Quit to end the program

==>
```

- Read the options and press the keys accordingly, for eg.
Press 0 to create a parking lot, the program will ask you to enter the **capacity** as well as **name** of the parking lot

```
==>0
Enter the size of parking lot: 6
Enter name of parking lot: Park Safe
Parking lot Park Safe created with capacity of 6 vehicles
```

- Similarly, you can check the menu and press the respective key/number to perform the operation mentioned in the menu.