Parking Lot

**Name: NILESHWAR PAUL**

**Batch: A.P.J Kalam**

**Contents:-**

1. Introduction
2. Module
3. SetUp
4. Key Features
5. Language Used
6. Future Aspect

**Introduction**

This Project is built to handle vehicle parking management .The Objective of the project is to hold up to given number of cars at any point of time. This system keep track of record of parked vehicle in the parking. If want to create an automated ticketing system that allows his customers to use his parking lot without human intervention, he can use this system.

When a car enters his parking lot, a ticket issued to the driver. The ticket issuing process includes documenting the registration number (number plate) and the color of the car and allocating an available parking slot to the vehicle nearest to the entry point.

At the exit, the customer returns the ticket, which then marks the slot they were using as being available. The system should provide this facility to Admin

\* Registration numbers of all cars of a particular color.

\* Slot number in which a car with a given registration number is parked.

\* Slot numbers of all slots where a car of a particular color is parked.

This system is using command line interface to interact with the user. This system can be used on any type of operating system. However, you need to install python interpreter to execute the scripts.

***Module***

This parking lot system is divided in modules. Each modules is dedicated for different tasks. This system has three module:-

1. **Module I (main.py):-** This module is the responsible for successfully executing the whole system. the execution of this system started from this module. This module is handling the user interaction. User give data through command line interface and this module taking them through some specific commands which is defined in this module. After taking data from the user this module validate that data and then according to the command sent that data to different module. This parking lot system can be used by two type, first is entering data through specifically defined commands or give a bunch of data in a form of text file to this module. This module is also responsible to handle any Exception arise during the execution. This module is handling the user interaction independently but all modules are depended to each other to run the system.
2. **Module II (parking.py):-** This is the second module of this system and this consisting all the functionality of the parking system like:
   1. **Create the parking slots**
   2. **Park the car**
   3. **Retrieve the details of vehicle**

The data taken by the first module is received here in this module. According to the commands chosen by the user data sent to the methods of this module. And the function perform the task and return the control to first module. This module can not run independently, it is instantiated by the first module.

1. **Module III (file\_car.py):-** This is the third and the last module of this system**.**

This module is dedicated for storing data of vehicle entity. this module is mainly used by Module II . when a vehicle come to the parking lot it register the vehicle’s data and store the object of this module in the slots which is containing attributes of the vehicle.

***SetUp***

1. Clone the repository

2. open shell in your system and set the path of this repository in your shell

3. Run `python main.py` to run without input test case file. This opens a shell where you can write your commands like -

4. First you need to create parking slots of your parking by entering command :-

`create\_parking\_lot n `

n is the number of vehicles can be parked in your parking

5. To run with a Input file execute `python main.py -f Input\_file\_name.txt.

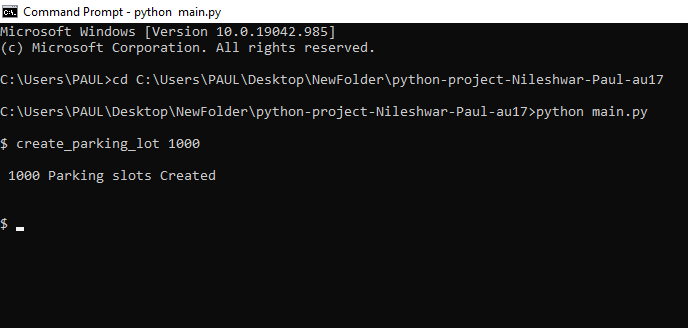
for example a input test file (run\_test.txt) is given in the repository

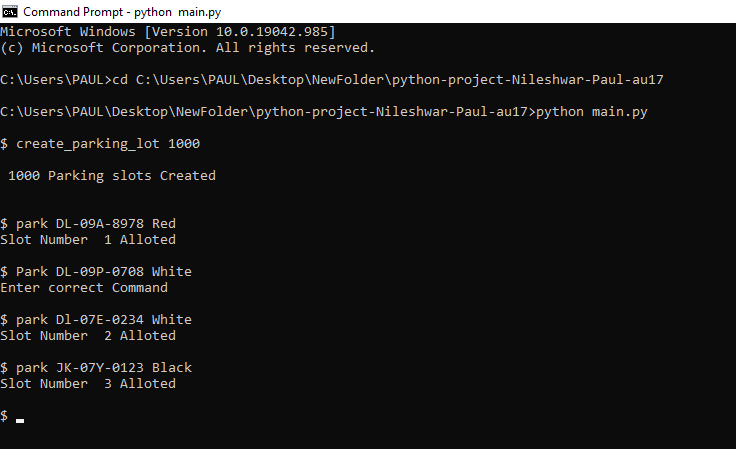
https://github.com/attainu/python-project-Nileshwar-Paul-au17/blob/dev/Pic/output\_of%20input\_text\_file.png

***Key Features***

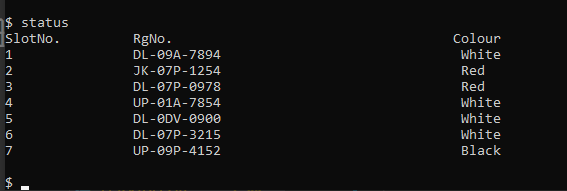
This system can perform some tasks through following commands:-

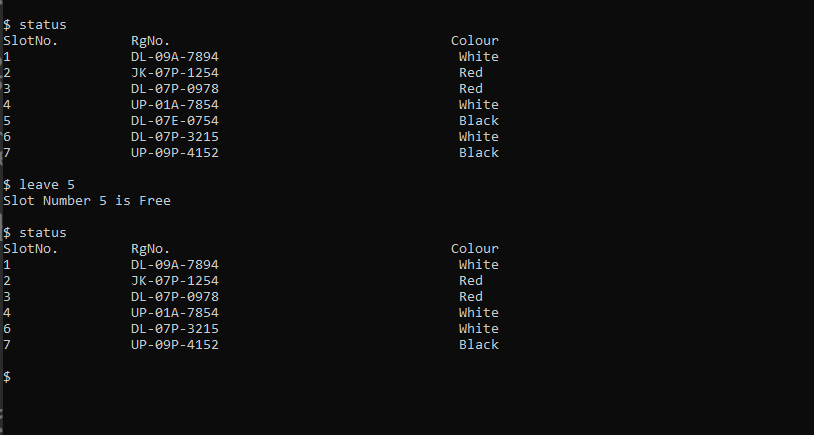
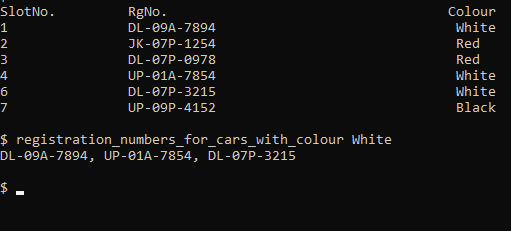
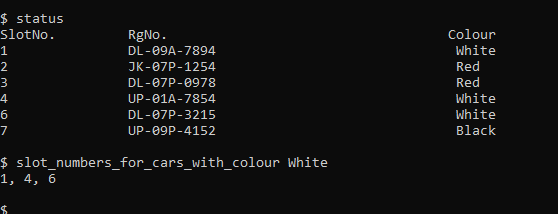
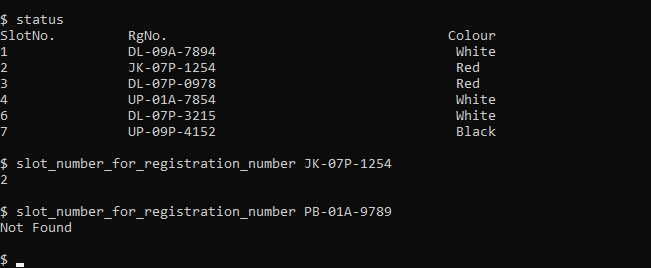
**Commands Descriptions**

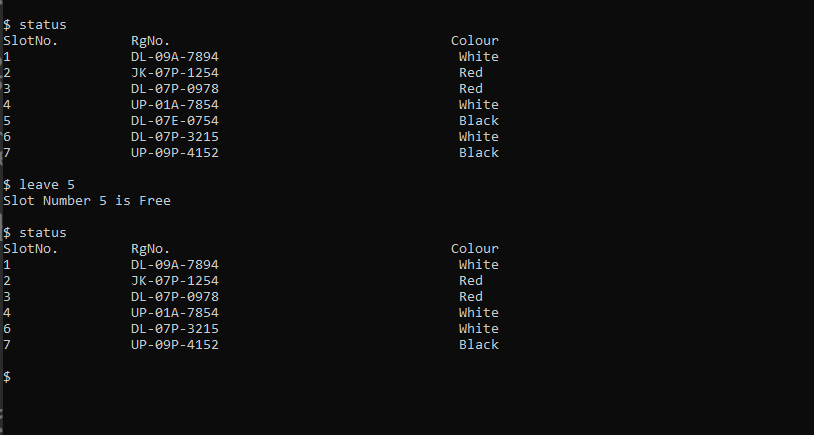
* ***‘create\_parking\_lot n’:-*** this command create parking lot of n numbers of slots.
* ***‘park car\_reg\_no car\_colour’:-***in this command you have to give vehicle registration number and the color of the vehicle separated by space.



* ***‘status’:-***This commands give you the list of vehicles parked in your parking with their registration number and colour



* ***‘leave x’:-***this command you can use for free the parking slot when a vehicle left his parking slot.in this command “x” is the parking slot number of vehicle.
* ***‘registration\_numbers\_for\_cars\_with\_colou colour \_of\_vehicle’:-*** By this command you can get the list of registration numbers of vehicles of that particular colour . 
* ***‘slot\_numbers\_for\_cars\_with\_colour colour\_name’:-*** By this command, you can get the list of parking slots number of cars of that colour.
* ***‘slot\_number\_for\_registration\_number registration\_num’:***By this command you can get the slot number of a vehicle by providing its registration number.
* ***‘Exit’:-***By using this command you can close the command at any point of time.



***Language Used***

This system is develop by using Python 3.9. Python is a user friendly high level script based on object oriented programing methodology. Python has some very useful library which can be used to make tasks easy. In the system, we follow the object orient programming principles like Encapsulation, Modularity and Abstraction.

Python Library used in this :-

1. ***argparse :-*** argparse is the “recommended command-line parsing module in the Python standard library.” It's what you use to get command line arguments into your program. i am using this library’s classes and its method to give the commands in a form of txt file through command line interface.
2. **sys:-** The sys module in Python provides various functions and variables that are used to manipulate different parts of the Python runtime environment. It allows operating on the interpreter as it provides access to the variables and functions that interact strongly with the interpreter. I am using it because I am using these library’s function to exit the system

***Future Aspects***

In addition, use Graphical user interface (GUI) to make it more user friendly

In future, I will develop the web version of these scripts and make it as web application.