## horizontal line

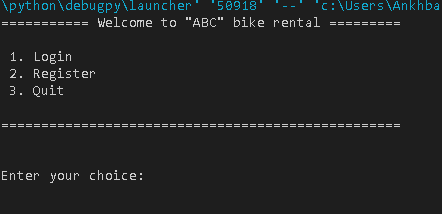


Bike Rental

# 

# Abstract

This project is a sample application that tracks rental information of the users registered to our system. Using this application in a working environment can enable our users to smoothly rent a bike, allow our users to retrieve information, such as: user details and their owing balance value to the company.



Ankhbayar Enkhlkhagva **COS340a Programming in Python**

ID: 100132497 Spring 2021

Mail address: ANE171@aubg.edu Instructor: Metodi Traykov

# 

# Introduction

Following Rental program is based on Python programming language. It is designed for anyone who would like to rent a bike from our company. Rather than going through the pages of hard copy document to rent a bike or to get information about them, a user can simply use this software to retrieve any information due to their needs.

Anyone using this program can register as a user. Using their account credentials, he/she can log into our system and start renting their products as well as checking their rentals. Small organizations need to have this software to keep the records of hundreds of users and their rental information at one place without any paperwork.

**Project Goals**

Key goals of the Rental application:

* Provide an easy interface for end users.
* Easy to store a bike detail.
* Easy to store user information.
* Easy to rent a car/motorcycle.
* Read and Write features

# 

# Class hierarchy

# Used modules and functions

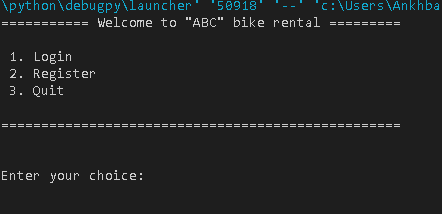
In order to get better efficiency, I used csv, os, and pandas modules. Also, I used list to traverse through different instances of the classes, in order to find and grab necessary data.

# User Manual

The following pages describe the concept of operations, procedures, information on software commands and problem resolution in respect of Rental application.

## Main menu

The following is the main menu of the program. Here the user has to enter the number written before the options. Any other entries are NOT acceptable and will print a message, asking the user to enter available selection.



*Main Menu*

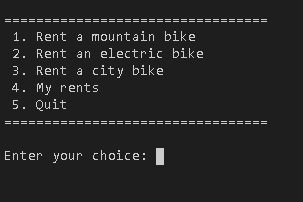
## User Login

If the user chooses the option [1] – User Login, the program will *clear the screen* and *ask user to enter* their username and password. After the user finishes entering their username and password, the program will *traverse through user\_list (a list containing instances of User class)*, where all user information is stored.

If there is a matching username and password, the user will be *redirected to User Menu*. If NOT, the program will and *ask the user to enter* their username and password, until they are valid.

## Rental menu

After successfully log in, user will be *redirected to* *rental menu*. Anything else from these options will make the program to ask you once more again.



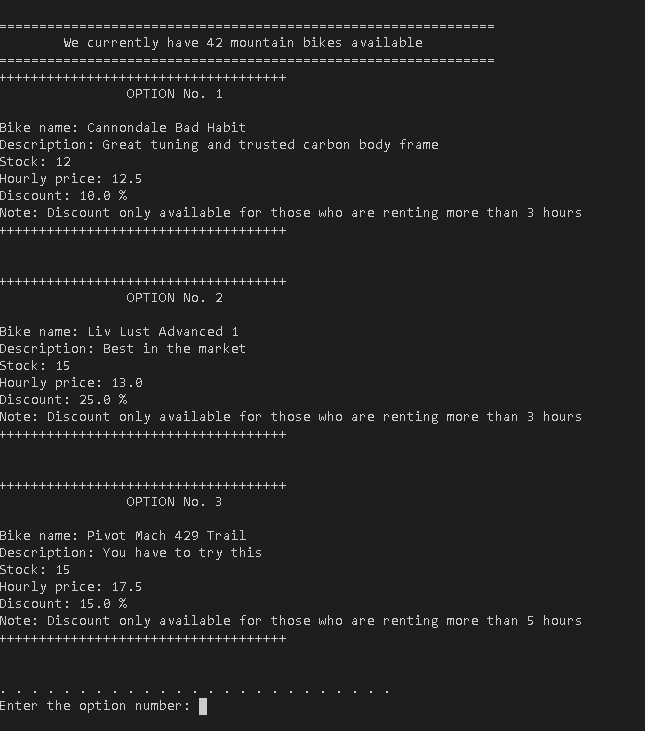
## Rental menu 🡪 Rent a mountain, electric, and city bike.

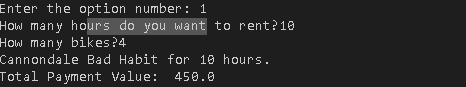
If you choose one of these, the procedure is all the same. It will print all possible options chosen from the category. For example, see the screenshot below. Following bike descriptions are made up by overloaded printIt() function in the subclasses.

After you choose your bike, the program will ask few more questions:

* + 1. Duration of rent
    2. Number of bikes

Once you fill them, the program will print you the bills and will update the csv file as well as the instances of the user.





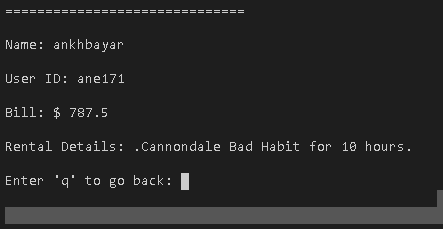
## My Rents

If the user chooses the option to check their account balance and user info, the program will print their full name, account balance (owing money to the company), and the list of their rented products.

Whenever the user wants to go back to the User menu, he/she can enter [q].

Account information menu

When the user enters wrong selection.



# Future Scope of the project

Our program will be more useful with following improvement:

●Adding a function that enables admins to add new bike into the system.

●Admin functionalities

●More products