./

GENESIS – Advanced Python Programming



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ver. Rel. No.** | **Release Date** | **Prepared. By** | **Reviewed By** | **To be Approved** | **Remarks/Revision Details** |
| 1 | 13-12-20 | Alen V George |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Details**

Contents

[Contents 3](#_Toc58757646)

[Bike Rental System 4](#_Toc58757647)

[**Day1:** 4](#_Toc58757648)

[**PEP8 Screenshot** 6](#_Toc58757649)

[**Day2:** 7](#_Toc58757650)

[**PEP8 Screenshot** 9](#_Toc58757651)

[**Day3** 10](#_Toc58757652)

[**PEP8 Screenshot** 13](#_Toc58757653)

# Bike Rental System

## **Day1:**

**Github Link**: <https://github.com/99003146/PythonProject/blob/main/Day1.py>

**Code:**

bikes = {"Pulsar": 20, "Unicorn": 30, "Royal Enfield": 10}

def display():

for (bike, no) in zip(bikes.keys(), bikes.values()):

print("Bike:", bike, "|", "Numbers available: ", no)

def book(bike\_name, num):

for values in bikes:

bikes[values] = int(bikes[values])

if bike\_name in bikes.keys():

value = bikes.get(bike\_name)

if (num <= value):

value = value - num

bikes.pop(bike\_name)

bikes[bike\_name] = value

print("You have successfully booked %d %s" % (num, bike\_name))

else:

print("Required number of bikes not available")

else:

print("Required Bike not available")

def ret\_bike(bike\_name, num, hours):

cost = num \* hours \* 50

for values in bikes:

bikes[values] = int(bikes[values])

if bike\_name in bikes.keys():

value = bikes.get(bike\_name)

value = value + num

bikes.pop(bike\_name)

bikes[bike\_name] = value

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

print("You have returned  ", bike\_name)

print("Number of bikes returned ", num)

print("Total amount payable is Rs :", cost)

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

def main():

print()

print("\*\*\*\*\*\*\*\*\*\*\*\*BIKE RENTAL SYSTEM\*\*\*\*\*\*\*\*\*\*\*")

print(" \*\*\*\*\*\*50 rs per hour for rentals\*\*\*\*\*\*\*\*")

print("1. Display details of bikes")

print("2. Rent bikes")

print("3. Return bikes")

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

choice = input("Enter choice: ")

choice = int(choice)

if choice == 1:

display()

if choice == 2:

bike\_name = input("Enter name of bike")

no\_bikes = input("Enter number of bikes")

no\_bikes = int(no\_bikes)

book(bike\_name, no\_bikes)

if choice == 3:

bike\_name = input("Enter name of bike")

if bike\_name in bikes.keys():

no\_bikes = input("Enter number of bikes")

no\_bikes = int(no\_bikes)

hours = input("Number of hours")

hours = int(hours)

ret\_bike(bike\_name, no\_bikes, hours)

else:

print("Check the bike name")

if \_\_name\_\_ == "\_\_main\_\_":

main()

## **PEP8 Screenshot**



## **Day2:**

**Code:**

from datetime import datetime

bike\_l = {"Pulsar": 20, "Unicorn": 30, "Royal Enfield": 10}

class Display:

def \_\_init\_\_(self):

self.bikes = {}

def display(self, bikes):

print("\*\*\*\*\*\*\*\*\*\*\*\* BIKES AVAILABLE \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

for (bike, no) in zip(bikes.keys(), bikes.values()):

print("Bike:", bike, "|", "Numbers available: ", no)

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

class Bike(Display):

def \_\_init\_\_(self):

self.bikes = {}

self.bike\_name = ""

self.number = 0

def rent(self, bikes, bike\_name, number):

self.bikes = bikes

self.bike\_name = bike\_name

self.number = number

if self.bike\_name in self.bikes.keys():

value = self.bikes.get(bike\_name)

newval = value-self.number

if(newval >= 0):

self.bikes[self.bike\_name] = newval

print("You have rented: ", self.bike\_name)

print("Numbers Rented:", self.number)

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

print()

Display.display(self, bike\_l)

else:

print("Required Number of bikes not available")

else:

print("Type a valid bike name")

def ret\_bike(self, bikes, bike\_name, number, hours\_used):

if bike\_name in bikes.keys():

cost = hours\_used\*number\*50

value = bikes.get(bike\_name)

newval = value+number

bikes[bike\_name] = newval

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

print("You have returned:", bike\_name)

print("Number Returned:", number)

print("Amount Payable:", cost)

class Customer(Bike, Display):

def \_\_init\_\_(self):

self.user\_n = ""

self.phone\_n = ""

def c\_rent(self, user\_n, phone\_n, bikes, bike\_name, number):

if bike\_name in bikes.keys():

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

print("UserName: ", user\_n)

print("Phone number: ", phone\_n)

Bike.rent(self, bikes, bike\_name, number)

else:

print("\nPlease Enter a valid bike name")

def c\_return(self, bike, bike\_name, number, hours\_used):

now = datetime.now()

current\_time = now.strftime("%H:%M:%S")

Bike.ret\_bike(self, bike, bike\_name, number, hours\_used)

print("You have returned the bike at ", current\_time)

print()

print("Database Updated")

print("New Count of Bikes")

print()

Display.display(self, bike)

def main():

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

print("Bikes at 50rs per hour")

print("\n1. Display Bike Details")

print("2. Rent Bike")

print("3. Return Bike")

print()

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

choice = input("Enter Choice: ")

if choice == "1":

dis = Display()

dis.display(bike\_l)

if choice == "2":

usrname = input("Enter User Name: ")

phoneno = input("Enter Phone Number: ")

bike\_n = input("Enter Name of Bike to rent: ")

number = int(input("Enter Number of bikes to rent: "))

ren = Customer()

ren.c\_rent(usrname, phoneno, bike\_l, bike\_n, number)

if choice == "3":

bike\_n = input("Enter Name of Bike to return: ")

number = int(input("Enter Number of bikes to return: "))

hours\_used = int(input("Enter Number of hours used: "))

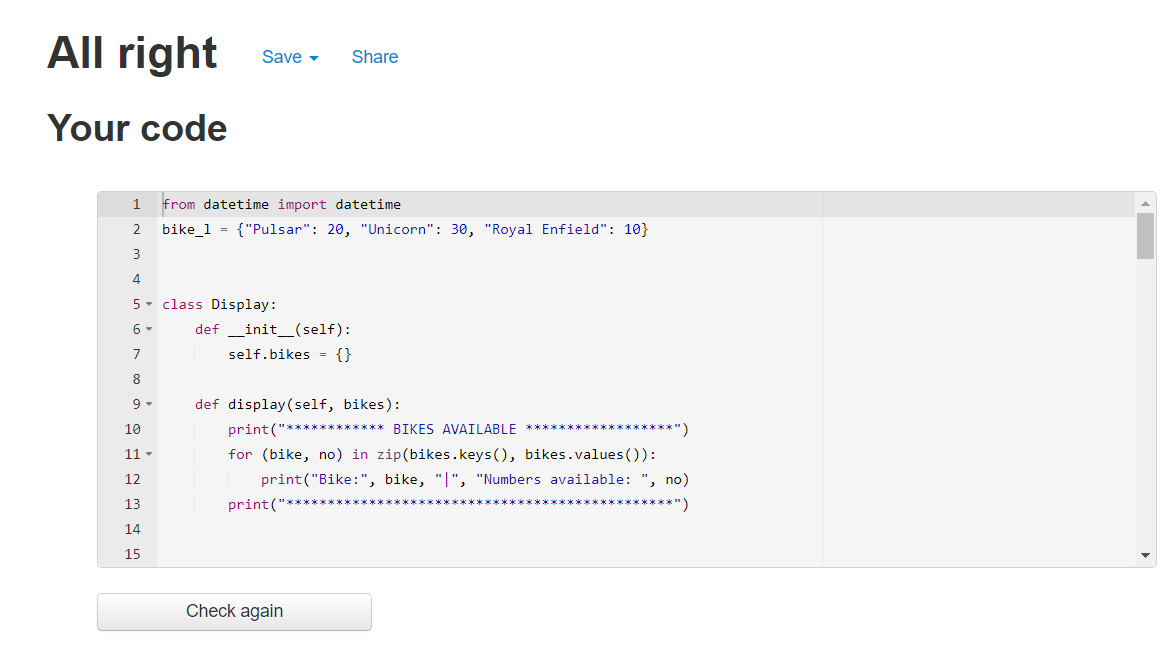
retu = Customer()

retu.c\_return(bike\_l, bike\_n, number, hours\_used)

if \_\_name\_\_ == "\_\_main\_\_":

main()

## **PEP8 Screenshot**



## **Day3**

**Github Link:** <https://github.com/99003146/PythonProject/blob/main/bike-rental.py>

**Code:**

import os

import re

from datetime import datetime

bike\_l = {"Pulsar": 20, "Unicorn": 30, "Royal Enfield": 10}

'''Class to Display Details'''

class Display:

def \_\_init\_\_(self):

self.bikes = {}

def display(self, bikes):

self.bikes = bikes

print("\*\*\*\*\*\*\*\*\*\*\*\* BIKES AVAILABLE \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

for (bike, no) in zip(self.bikes.keys(), self.bikes.values()):

print("Bike:", bike, "|", "Numbers available: ", no)

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

'''Class for renting and returning'''

class Bike(Display):

def \_\_init\_\_(self):

self.bikes = {}

self.bike\_name = ""

self.number = 0

self.hours\_used = 0

def rent(self, bikes, bike\_name, number):

self.bikes = bikes

self.bike\_name = bike\_name

self.number = number

if self.bike\_name in self.bikes.keys():

value = self.bikes.get(bike\_name)

newval = value-self.number

if(newval >= 0):

self.bikes[self.bike\_name] = newval

print("You have rented: ", self.bike\_name)

print("Numbers Rented:", self.number)

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

print()

os.system('pause')

Display.display(self, bike\_l)

else:

print("Required Number of bikes not available")

else:

print("Type a valid bike name")

def ret\_bike(self, bikes, bike\_name, number, hours\_used):

self.bikes = bikes

self.bike\_name = bike\_name

self.number = number

self.hours\_used = hours\_used

if self.bike\_name in self.bikes.keys():

cost = self.hours\_used\*self.number\*50

value = self.bikes.get(self.bike\_name)

newval = value+self.number

bikes[bike\_name] = newval

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

print("You have returned:", self.bike\_name)

print("Number Returned:", self.number)

print("Amount Payable:", cost)

'''Customer Class'''

class Customer(Bike, Display):

def \_\_init\_\_(self):

self.user\_n = ""

self.phone\_n = ""

def c\_rent(self, user\_n, phone\_n, bikes, bike\_name, number):

self.user\_n = user\_n

self.phone\_n = phone\_n

if bike\_name in bikes.keys():

print("\*\*\*\*\*\*\*\*\*\*\* BILL \*\*\*\*\*\*\*\*\*\*\*\*\*\*")

print("UserName: ", self.user\_n)

print("Phone number: ", self.phone\_n)

Bike.rent(self, bikes, bike\_name, number)

else:

print("\nPlease Enter a valid bike name")

def c\_return(self, bike, bike\_name, number, hours\_used):

now = datetime.now()

current\_time = now.strftime("%H:%M:%S")

Bike.ret\_bike(self, bike, bike\_name, number, hours\_used)

print("You have returned the bike at ", current\_time)

print()

os.system('pause')

print("Database Updated")

print("New Count of Bikes")

print()

Display.display(self, bike)

'''Main Function'''

def main():

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* MAIN MENU \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

print(" Bikes at 50rs per hour ")

print("\n1. Display Bike Details")

print("2. Rent Bike")

print("3. Return Bike")

print("4. Exit")

print()

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

choice = input("Enter Choice: ")

if choice == "1":

dis = Display()

dis.display(bike\_l)

os.system('pause')

main()

if choice == "2":

usrname = input("Enter User Name: ")

phoneno = input("Enter Phone Number: ")

''' Regex to Match Phone Number'''

number = re.match("[0-9]{10}$", phoneno)

if number is None:

print("Invalid Phone Number! Try Again")

os.system('pause')

main()

else:

bike\_n = input("Enter Name of Bike to rent: ")

number = int(input("Enter Number of bikes to rent: "))

ren = Customer()

ren.c\_rent(usrname, phoneno, bike\_l, bike\_n, number)

os.system('pause')

main()

if choice == "3":

bike\_n = input("Enter Name of Bike to return: ")

number = int(input("Enter Number of bikes to return: "))

hours\_used = int(input("Enter Number of hours used: "))

retu = Customer()

retu.c\_return(bike\_l, bike\_n, number, hours\_used)

os.system('pause')

main()

if choice == "4":

exit()

if \_\_name\_\_ == "\_\_main\_\_":

main()

## **PEP8 Screenshot**

