



# THE UNIVERSITY OF TEXAS AT DALLAS

MIS 6382.503 – Object-Oriented Programming in Python

## Scooter Rental System

*Submitted By:*

**Ishan Chandrahas Haldankar**

**Ned Id: ich220000**

### **Acknowledgment**

To begin with, I would like to express our gratitude to Prof. Vatsal Maru and Teaching Assistant Harshita Katakam for allowing me to complete a project and helping me understand the subject and concepts throughout the semester and coursework. Their enthusiasm, patience, insightful comments, helpful information, practical advice, and unceasing ideas have helped me tremendously in our analysis and writing of this project. Their immense knowledge, profound experience, and professional expertise in object-oriented programming have enabled me to complete this project successfully. Without their support and guidance, this project would not have been possible.

Thank You

### **Introduction**

This project aims to provide a versatile and practical Scooter Rental System to manage scooter rentals efficiently across several stores. The technology enables clients to lease scooters for different periods, including options for renting by the hour, day, or week. The solution incorporates dynamic pricing based on the chosen rental duration, promotional incentives for renting multiple scooters, and the ability to generate bills upon scooter return.

The ScooterRentalShop class contains the essential features, enabling the development of many rental shops with a set quantity of scooters. The program guarantees that consumers cannot rent a quantity that exceeds the number of scooters currently in stock at the shop. Moreover, it computes the entire sum owed by considering the chosen rental choices, including the length of time and number of scooters rented.

The system integrates a discount mechanism for clients renting more than two scooters to improve customer satisfaction. This motivates renting in significant quantities by providing a 15% reduction in the overall cost.

The program has an emergency stop function, enabling rental shop operators to cease the program immediately. This feature offers a resilient and regulated method to terminate the system's operation in unexpected situations.

The user interface, integrated within the core function, enables consumer engagement by presenting rental choices, gathering customer data, and overseeing the rental and return procedures. The algorithm guarantees a smooth and uninterrupted process by reminding users of essential inputs and delivering precise information regarding the rental transactions.

The following sections of this report will examine the intricate operation of the system, including insights into the code structure, the reasoning behind design decisions, and scenarios

for testing the software. The implementation adheres to the requirements specified in the project description, offering a comprehensive solution for a scooter rental system.

### **Method**

#### **Constants:**

MAX\_RENTAL\_PERIOD = 24

DISCOUNT\_THRESHOLD = 2

DISCOUNT\_PERCENT = 15

HOURLY\_RATE = 10

DAILY\_RATE = 20

WEEKLY\_RATE = 50

These variables establish the parameters for the scooter rental system.

MAX\_RENTAL\_PERIOD defines the upper limit of the rental period, measured in hours.

DISCOUNT\_THRESHOLD and DISCOUNT\_PERCENT are utilized to develop the conditions under which a discount should be granted for renting multiple scooters.

#### **ScooterRentalShop Class:**

This session encompasses the fundamental features of the scooter rental shop. The system comprises functionalities for presenting rental choices, leasing scooters, returning scooters, and activating an emergency halt.

#### **Main Function:**

The primary purpose of the main function is to act as the user interface. It creates a scooter rental shop, displays rental options, and interacts with the user to facilitate renting and returning scooters. The iteration persists until the user opts to cease renting.

#### **If \_\_name\_\_ = "main":**

This code block guarantees that the main function is invoked only when the script is executed directly rather than imported as a module.

Essentially, the code establishes a flexible and engaging scooter rental system that enables users to rent and return scooters, computes rental costs, and implements discounts when appropriate. The emergency stop option offers a regulated method to terminate the program. The constants and methods are meticulously arranged, ensuring the code is easily understandable and efficiently maintained.

### **Result**

#### **Scenario 1:**

- Customer Name: Ishan
- Rental Request: 5 scooters for 9 hours
- Rental Type: Hourly
- Discount Applied: Yes (15% for renting more than 2 scooters)
- Total Amount Due: \$382.5
- Return: Yes, returning all 5 scooters
- Program Continues: Yes

Ishan leases a fleet of 5 scooters for 9 hours, which leads to a cumulative payment of \$382.5. A 15% deduction is implemented for renting more than 2 scooters. Ishan relinquishes all 5 scooters, and the program proceeds with further rentals.

#### **Scenario 2:**

- Customer Name: Ishan
- Rental Request: 4 scooters for 4 days
- Rental Type: Daily
- Discount Applied: Yes (15% for renting more than 2 scooters)
- Total Amount Due: \$272.0
- Return: Yes, returning all 4 scooters
- Program Continues: Yes

Ishan hires 4 scooters for 4 days, leading to a total payment of \$272.0. A 15% deduction is implemented for renting more than 2 scooters. Ishan opts to return all four scooters, and the program proceeds with additional rentals.

**Scenario 3:**

- Customer Name: Ishan
- Rental Request: 3 scooters for 1 week
- Rental Type: Weekly
- Discount Applied: Yes (15% for renting more than 2 scooters)
- Total Amount Due: \$127.5
- Return: Yes, returning all 3 scooters
- Program Continues: No (Emergency stop)

Ishan leases three scooters for one week, which leads to a cumulative payment of \$127.5. A 15% reduction is granted due to renting more than 2 scooters. Ishan opted to relinquish all three scooters, ending the program abruptly with an emergency halt.



**Summary of the Project:**

The provided scenarios illustrate the operational capabilities of the Scooter Rental System. Customers can rent scooters for different lengths of time, get discounts for renting in large quantities, and return the scooters, which results in the computation of the total payment. If necessary, the emergency stop function enables the rental business to terminate the program abruptly. These scenarios demonstrate the system's adaptability and quick response, guaranteeing a seamless user experience in various rental situations. The specified requirements have implemented the discount mechanism and rental calculations, resulting in a robust solution for managing scooter rentals.