BIRZEIT UNIVERSITY

Computer Science Department

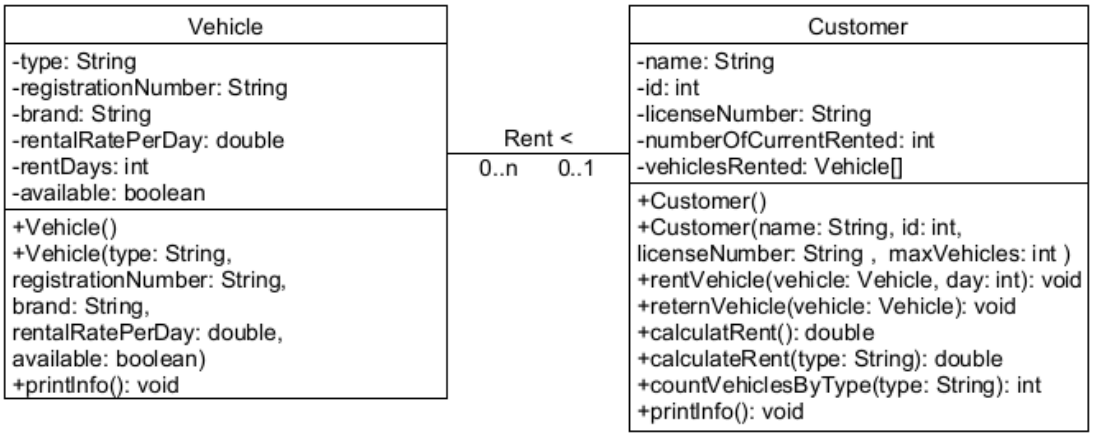
COMP2310 (Fall 2024/2025)

Assignment #2

Notes:

1. The assignment should be submitted by ***Friday 22/11/2024 by 11:59 pm on ITC (Late Assignments will not be accepted for any reason).***
2. The assignments are ***individual*** effort and copying the assignment will be treated as a cheating attempt, which may lead to *FAILING* the course.

Using ONLY eclipse IDE, write a complete Java program that first creates the following two classes:



- **Vehicle Methods**:
    - **Constructors**: A default constructor and a parameterized constructor to initialize the attributes.
    - **Setters and Getters** for each attribute.
    - `printInfo()`: Prints the vehicle's details in an organized format.
- **Customer Methods**:
    - **Constructors**: A default constructor and a parameterized constructor to initialize attributes.
    - **Setters and Getters** for each attribute.
    - `rentVehicle(Vehicle vehicle, int day)`: Adds a vehicle to the customer's array of rented vehicles.
    - `returnVehicle(Vehicle vehicle)`: remove a vehicle from the customer's array of rented vehicles.

- `calculateRent()`: Calculates the total rental cost based on the number of rental days.
- `calculateRent(String type)`: overload `calculateRent()` method to calculates the total rental cost based on *type*.
- `countVehiclesByType(String type)`: Returns the count of rented vehicles of a specific type.
- `printInfo()`: Prints customer details in an organized format, including rented vehicle details.

1. **Driver Class**

- **Predefined Array of Available Vehicles**:

  - Create an array of `Vehicle` objects with a predefined set of vehicles.
  - You can use the following:

```
Vehicle[] availableVehicles = {

          new Vehicle("Car", "ABC123", "Toyota", 50.0, true),

          new Vehicle("Bike", "DEF456", "Honda", 20.0, true),

          new Vehicle("Truck", "GHI789", "Ford", 80.0, true),

          new Vehicle("Car", "JKL012", "Hyundai", 55.0, true),

          new Vehicle("Bike", "MNO345", "Yamaha", 160.0, true)

       };
```

- **Main Program Flow:**

1. **Initialize the Program**:

   - Prompt the user to enter the **number of customers**.
   - Create an array of `Customer` objects with the specified size.

2. **Gather Customer Information**:

   - For each customer:

     - Prompt the user to enter **name, ID, license number,** and the **maximum number of vehicles** they wish to rent.
     - Initialize the `numberOfCurrentRented` field to `Zero` for each customer.

3. **Display Available Vehicles for Rent**:

   - Show a list of all available vehicles with details such as type, registration number, brand, and daily rate.

4. **Rent Vehicles to the Customer**:

   - For each vehicle the customer wishes to rent:

- Check if `numberOfCurrentRented` is less than the **maximum number of vehicles**. If it's equal, display a message (e.g., "You have reached your rental limit") and stop the process for this customer.
- Prompt the user to **enter a vehicle's registration number** or type **"exit"** to stop renting:
  - **If the customer enters "exit"**, display a message like "Rental process ended by customer," and break out of the rental loop, ending the process for this customer.
  - If the customer enters a registration number, proceed with the following steps:
    - **Find the Vehicle**: Search for the vehicle in the `Vehicles` array.
      - If not found, display "Vehicle not found, please try again."
    - **Check Availability**: If the vehicle is found but unavailable, display "Vehicle not available, please choose another."
    - **Rent Vehicle to Customer**: If the vehicle is available:
      - Add it to the customer's array of rented vehicles.
      - Update the vehicle's availability to false, to indicate rented.
      - Increment `numberOfCurrentRented` by `1`.
      - Confirm the rental (e.g., "Vehicle [registration number] rented successfully").

5. **End of Customer Input**:

- Once all customers have been processed, proceed to the **main menu** for further options.

- **Static Methods**:
  - `public static Customer findCustomerById(Customer[] customers, int customerId)`:
    This method searches through an array of `Customer` objects to find a customer by their ID. If found, it returns the `Customer` object; otherwise, it returns `null`.
  - `public static Vehicle findVehicleByRegistrationNumber(Vehicle[] vehicles, String regNumber)`:
    This method searches through an array of `Vehicle` objects to find a vehicle by its registration number. If found, it returns the `Vehicle` object; otherwise, it returns `null`.
  - `public static void displayVehiclesByPrice(Vehicle[] vehicles)`:
    This method displays an array of `Vehicle` objects in ascending order of `rentalRatePerDay`. You have to write the function by implementing the sorting algorithm. **Do not use built-in method**.
  - `public static void displayVehiclesByType(Vehicle[] vehicles)`:
    This method displays an array of `Vehicle` objects in alphabetical order of `type`. You have to write the function by implementing the sorting algorithm. **Do not use built-in method**.

- **Menu Options**:
  - **1**: **Print Customer Information**
    Prompt for a customer ID and display all of the customer's details.
  - **2**: **Display Total Rental Cost for a Customer**
    Prompt for a customer ID and the number of rental days, then calculate and display the total rental cost.

- **3**: **Count Rented Vehicles by Type**
  Prompt for a customer ID and vehicle type, then display the count of that vehicle type for the customer.
- **4**: **Rent a New Vehicle**
  - Prompt for customer ID then ask the user to enter the type of vehicle they wish to rent.
  - Display a list of available vehicles of that type along with their price per day.
  - Before proceeding, check if the customer has reached the maximum rental limit. If they have, inform the user that they cannot rent more vehicles.
  - If under the limit, ask the user to enter the registration number of the selected vehicle and the number of rental days.
  - Calculate and display the total rental cost.
  - Update the customer's rental details with the selected vehicle
  - Update the vehicle's availability to false, to indicate rented.
  - Increment customer's numberOfCurrentRented by 1.
- **5: Return a Vehicle**
  - Prompt for customer ID.
  - Display a list of all vehicles currently rented by the customer.
  - Ask the user to enter the registration number of the vehicle they wish to return.
  - Confirm the return, then update the system by removing the returned vehicle from the customer's rentals array.
  - Update the vehicle's availability to true to indicate return.
  - Decrement customer's numberOfCurrentRented by 1.
  - Do not use any built-in method.
- **6**: **Display All Available Vehicles in Ascending Order of Price**
  Show a list of all vehicles currently available for rent.
- **7: Display All Available Vehicles in alphabetical Order of Type**
  Show a list of all vehicles currently available for rent.
- **8: Exit**
  Exit the program.

Note: You need to validate if the customer or vehicle is found and available. However, there is No need to implement input validation for any other user inputs.

Sample Run:

```
Enter number of customers: 2


Enter details for customer 1:

Enter name: John Doe

Enter ID: 1001

Enter license number: 12345678

Enter Maximum number of vehicles can rent by this customer: 10

Enter the number of vehicles to rent now: 2


Available vehicles for rental:
```

```
1. Car (Registration: ABC123), Toyota - $50.0/day
2. Bike (Registration: DEF456), Honda - $20.0/day
3. Truck (Registration: GHI789), Ford - $80.0/day
4. Car (Registration: JKL012), Hyundai - $55.0/day
5. Bike (Registration: MNO345), Yamaha - $160.0/day

Enter registration number of vehicle 1 to rent: XYZ999
Vehicle not found. Please enter a valid registration number.
Enter registration number of vehicle 1 to rent: ABC123
Enter rental days: 5

Vehicle rented successfully.

Enter registration number of vehicle 2 to rent: DEF456
Enter rental days: 3
Vehicle rented successfully.

Enter details for customer 2:
Enter name: Alice Smith
Enter ID: 1002
Enter license number: 87654321
Enter Maximum number of vehicles can rent by this customer: 7
Enter the number of vehicles to rent: 1

Available vehicles for rental:
1. Truck (Registration: GHI789), Ford - $80.0/day
2. Car (Registration: JKL012), Hyundai - $55.0/day
3. Bike (Registration: MNO345), Yamaha - $160.0/day

Enter registration number of vehicle 1 to rent: DEF456
Vehicle not available. Please choose another.
Enter registration number of vehicle 1 to rent: GHI789
Enter rental days: 2
Vehicle rented successfully.

Main Menu:
1: Print Customer Information
2: Display Total Rental Cost for a Customer
```

```
3: Count Rented Vehicles by Type
4: Rent a New Vehicle
5: Return a Vehicle
6: Display All Available Vehicles in Ascending Order of Price
7: Display All Available Vehicles in Alphabetical Order of Type
8: Exit
Enter your choice: 1
Enter customer ID: 9999
Customer not found.

Main Menu:
1: Print Customer Information
2: Display Total Rental Cost for a Customer
3: Count Rented Vehicles by Type
4: Rent a New Vehicle
5: Return a Vehicle
6: Display All Available Vehicles in Ascending Order of Price
7: Display All Available Vehicles in Alphabetical Order of Type
8: Exit

Enter your choice: 1
Enter customer ID: 1001
Customer Information:
Name: John Doe
ID: 1001
License Number: 12345678
Vehicles Rented:
   - Vehicle Code: CARABC123, Type: Car, Brand: Toyota, Daily Rate: $50.0
   - Vehicle Code: BIKEDEF456, Type: Bike, Brand: Honda, Daily Rate: $20.0

Main Menu:
1: Print Customer Information
2: Display Total Rental Cost for a Customer
3: Count Rented Vehicles by Type
4: Rent a New Vehicle
5: Return a Vehicle
6: Display All Available Vehicles in Ascending Order of Price
7: Display All Available Vehicles in Alphabetical Order of Type
```

```
8: Exit

Enter your choice: 2
Enter customer ID: 1003
Customer not found.

Main Menu:
1: Print Customer Information
2: Display Total Rental Cost for a Customer
3: Count Rented Vehicles by Type
4: Rent a New Vehicle
5: Return a Vehicle
6: Display All Available Vehicles in Ascending Order of Price
7: Display All Available Vehicles in Alphabetical Order of Type
8: Exit

Enter your choice: 2
Enter customer ID: 1001
Total Rental Cost for John Doe: $310.0

Main Menu:
1: Print Customer Information
2: Display Total Rental Cost for a Customer
3: Count Rented Vehicles by Type
4: Rent a New Vehicle
5: Return a Vehicle
6: Display All Available Vehicles in Ascending Order of Price
7: Display All Available Vehicles in Alphabetical Order of Type
8: Exit
Enter your choice: 3
Enter customer ID: 1001
Enter vehicle type to count (e.g., Car, Bike): Car
Number of Car(s) rented by John Doe: 1

Main Menu:
1: Print Customer Information
2: Display Total Rental Cost for a Customer
3: Count Rented Vehicles by Type
```

```
4: Rent a New Vehicle
5: Return a Vehicle
6: Display All Available Vehicles in Ascending Order of Price
7: Display All Available Vehicles in Alphabetical Order of Type
8: Exit
Enter your choice: 4
Enter customer ID: 1002
Enter the type of vehicle to rent (e.g., Car, Bike): Car

Available vehicles of type 'Car':
1. Car (Registration: JKL012), Hyundai - $55.0/day

Enter registration number of the vehicle to rent: JKL012
Enter the number of rental days: 4
Vehicle JKL012 rented successfully.

Main Menu:
1: Print Customer Information
2: Display Total Rental Cost for a Customer
3: Count Rented Vehicles by Type
4: Rent a New Vehicle
5: Return a Vehicle
6: Display All Available Vehicles in Ascending Order of Price
7: Display All Available Vehicles in Alphabetical Order of Type
8: Exit

Enter your choice: 5
Enter customer ID: 1001
List of vehicles currently rented:
   - Car (Registration: ABC123), Toyota - $50.0/day
   - Bike (Registration: DEF456), Honda - $20.0/day
Enter registration number of the vehicle to return: ABC123
Vehicle ABC123 returned successfully.

Main Menu:
1: Print Customer Information
2: Display Total Rental Cost for a Customer
3: Count Rented Vehicles by Type
```

```
4: Rent a New Vehicle
5: Return a Vehicle
6: Display All Available Vehicles in Ascending Order of Price
7: Display All Available Vehicles in Alphabetical Order of Type
8: Exit

Enter your choice: 5
Enter customer ID: 1002
List of vehicles currently rented:
   - Truck (Registration: GHI789), Ford - $80.0/day
   - Car (Registration: JKL012), Hyundai - $55.0/day
Enter registration number of the vehicle to return: ABC123
Vehicle GHI789 returned successfully.

Main Menu:
1: Print Customer Information
2: Display Total Rental Cost for a Customer
3: Count Rented Vehicles by Type
4: Rent a New Vehicle
5: Return a Vehicle
6: Display All Available Vehicles in Ascending Order of Price
7: Display All Available Vehicles in Alphabetical Order of Type
8: Exit

Enter your choice: 6
Available Vehicles Sorted by Rental Rate:
1. Car (Registration: ABC123), Toyota - $50.0/day
2. Truck (Registration: GHI789), Ford - $80.0/day
3. Bike (Registration: MNO345), Yamaha - $160.0/day

Main Menu:
1: Print Customer Information
2: Display Total Rental Cost for a Customer
3: Count Rented Vehicles by Type
4: Rent a New Vehicle
5: Return a Vehicle
6: Display All Available Vehicles in Ascending Order of Price
7: Display All Available Vehicles in Alphabetical Order of Type
```

```
8: Exit


Enter your choice: 7

Available Vehicles in Alphabetical Order of Type:

1. Bike (Registration: MNO345), Yamaha - $160.0/day

2. Car (Registration: ABC123), Toyota - $50.0/day

3. Truck (Registration: GHI789), Ford - $80.0/day


Main Menu:

1: Print Customer Information

2: Display Total Rental Cost for a Customer

3: Count Rented Vehicles by Type

4: Rent a New Vehicle

5: Return a Vehicle

6: Display All Available Vehicles in Ascending Order of Price

7: Display All Available Vehicles in Alphabetical Order of Type

8: Exit


Enter your choice: 8

Goodbye!
```

VERY IMPORTANT:

1. Your project folder (containing all your ***.java*** project files) should be compressed (.rar) and saved as ***ass2_youridnumber_yourLabsectionnumber.rar.*** your compressed folder should be submitted on **Moodle ITC *COMP2310* - *Meta*** "Assignment Two"

2. Ensure that each class file begins with a comment including your full name, student ID number, and both your lecture and lab section numbers.

3. Any late or incorrect submissions (**even by one minute**) or submissions not sent as a reply to the coordinator's message EXACTLY as instructed will not be graded and will result in a zero. **NO EXCEPTIONS for ANY REASON**.