

**SCHOOL OF INFOCOMM TECHNOLOGY**

Diploma in Information Technology

Diploma in Financial Informatics

**OBJECT-ORIENTED PROGRAMMING**

**Year 2014/15 - Semester 2**

**Assignment**

**Duration :**  3 weeks (19 Jan 2015 to 6 Feb 2015)

**Weightage :** 40% of total coursework

**Individual/Team :** Individual

**Format :** Programming (40%)

Walkthrough (60%)

**Cut-Off Time: Friday, 6 February 2015, 8:45 AM**

There is a total of 12 pages (including this page) in this hand-out.

|  |
| --- |
| ***WARNING***  ***If a student is found to have submitted work not done by him/her, he/she will not be awarded any marks for this assignment. Disciplinary action will also be taken.***  ***Similar action will be taken for the student who allows other student(s) to copy his/her work.*** |

**Vehicle Rental System**

|  |
| --- |
| In this assignment, you are to apply Object Oriented Programming to develop a simple ***Vehicle Rental System***. The assignment requirements described below are broken down into 2 stages of development, described in this document as **'*Basic Features*'** and **'*Advanced* *Features*'**. You are advised to do your programming progressively in these stages. Refer to the **'*Grading Criteria*'** to have an idea of how the different components are graded. |

# Background

Reliance Rental, a vehicle rental company on our island country state, has engaged you to develop a computerised vehicle rental system to streamline its rental process. Its management has requested for a simple prototype for the vehicle rental system to be developed for a start.

The company has a fleet of cars and trucks to be rented. Each vehicle is identified by a registration number, make, model, daily rate and availability status. The rented car may come with child seat, and each child seat is chargeable at $10 per day. As for the rented truck, customer may request for a driver, and the driver fee is $100 per day. Information of some vehicles is given in Table 1 below.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Registration No** | **Make** | **Model** | **Daily Rate ($)** | **Availability** |
| Car | SJV1883R | Honda | Civic | 60.00 | Yes |
| SJZ2987A | Toyota | Altis | 60.00 | Yes |
| SKA4370H | Honda | Accord | 80.00 | Yes |
| SKD8024M | Toyota | Camry | 80.00 | Yes |
| SKH5922D | BMW | 320i | 90.00 | Yes |
| SKM5139C | BMW | 520i | 100.00 | Yes |
| SKP8899H | Mercedes | S500 | 300.00 | Yes |
| Truck | GB3221K | Tata | Magic | 200.00 | Yes |
| YB8283M | Isuzu | NPR | 250.00 | Yes |
| YK5133H | Isuzu | NQR | 300.00 | Yes |

Table 1 – Information of the vehicles

The information of their existing customers are as shown in Table 2 below.

|  |  |  |  |
| --- | --- | --- | --- |
| **IC Number** | **Name** | **Date of Birth** | **Tel** |
| S9099999A | Adrian | 22/03/1990 | 91112222 |
| S8558911B | Benjamin | 11/08/1989 | 96668888 |
| S7670326C | Christina | 13/12/1979 | 93335555 |

Table 2 – Existing customer data

The class diagram for the vehicle rental system is shown in Appendix 1.

# Basic Features

* **Register customer**
* *prompt user to enter the following information for the customer: IC number, name, date of birth and telephone number*
* *create a customer object with the information given*
* *add the customer object to the customer list*
* *display the status of the registration (i.e. successful or not successful)*
* **List all customers**
* *display the information of all the customers*
* **List all vehicles**
* *display the information of all the vehicles*
* **List all available vehicles**
* *display the information of all the available vehicles*
* **Rent a vehicle**
  + *list the customers*
  + *prompt user to select a customer*
  + *retrieve the selected customer*
  + *list all vehicles*
  + *prompt user to select a vehicle*
  + *retrieve the selected vehicle*
  + *prompt user to enter the pick-up date*
  + *prompt user to enter the return date*
  + *display the rental charges*
  + *create a rental object with the information given*
  + *add the rental object to the rental list of the customer*
  + *add the rental object to the rental list of the vehicle rental system*
  + *display the status of the rental (i.e. successful or not successful)*
* **List rental details for all cars or trucks**
* *prompt user to select car or truck*
* *retrieve and display the rental details for all cars or trucks according to the user input*
* **Validations** (and feedback)
* *The program should handle all invalid entries by the user*

*e.g. invalid option, invalid year, invalid month, invalid day, etc.*

* *If user made a mistake in the entry, the program should inform the user via appropriate feedback*

# Advanced Features

You are required to do all the advanced features below.

* **Return vehicle**
* *prompt user to enter the IC number of the customer*
* *prompt user to enter the registration number of the vehicle*
* *search and retrieve the rental details of the vehicle*
* *prompt user to enter the return date*
* *update the return date to the rental object*
* *calculate and display the total rental charges*
* **List car or truck with the highest number of rental transactions**
* *prompt user to select car or truck*
* *display the information of the car or truck that has the most number of rental records*
* **List rental details of a customer over a period**
* *prompt user to input IC number of the customer*
* *prompt user for a start date and an end date*
* *retrieve and display the rental details for the customer over the period*

***Please note that you should implement the advanced features only AFTER all the basic features have been fully implemented and working.***

*NO MARKS will be awarded for the advanced features if the basic features have NOT been fully implemented and working.*

## Screen Shots

Some sample screen shots are provided to guide you in your assignment. You are free to design your own interface.

**Main screen:**

|  |
| --- |
| MENU  =====  1. Register customer  2. List all customers  3. List all vehicles  4. List all available vehicles  5. Rent a vehicle  6. List all rental details for all cars or trucks  0. Exit  Enter your option : \_\_ |

**Option 1 : Register customer**

|  |
| --- |
| Enter your option : **1**  Option 1. Register Customer  Enter NRIC Number: **S8999223A**  Enter Name: **James**  Enter Date of Birth (DD MM YYYY): **28 05 1985**  Enter Tel No: **91125566**  James is registered successfully. |

**Option 2 : List all customers**

|  |
| --- |
| Enter your option : **2**  Option 2. List All Customers  S/No IC Number Name Date of Birth Tel  1 S9099999A Adrian 22/03/1990 91112222  2 S8988888B Benjamin 11/08/1989 96668888  3 S7901234C Christina 13/12/1979 93335555 |

**Option 3 : List all vehicles**

|  |
| --- |
| Enter your option : **3**  Option 3. List All Vehicles  S/No Reg No Make Model Daily Rate ($) Availability  1 SJV1883R Honda Civic 60.00 Yes  2 SJZ2987A Toyota Altis 60.00 Yes  3 SKA4370H Honda Accord 80.00 Yes  4 SKD8024M Toyota Camry 80.00 Yes  5 SKH5922D BMW 320i 90.00 Yes  6 SKM5139C BMW 520i 100.00 Yes  7 SKP8899H Mercedes S500 300.00 Yes  8 GB3221K Tata Magic 200.00 Yes  9 YB8283M Isuzu NPR 250.00 Yes  10 YK5133H Isuzu NQR 300.00 Yes |

**Option 4 : List all available vehicles**

|  |
| --- |
| Enter your option : **4**  Option 4. List All Available Vehicles  S/No Reg No Make Model Daily Rate ($)  1 SJV1883R Honda Civic 60.00  2 SJZ2987A Toyota Altis 60.00  3 SKA4370H Honda Accord 80.00  4 SKD8024M Toyota Camry 80.00  5 SKH5922D BMW 320i 90.00  6 SKM5139C BMW 520i 100.00  7 SKP8899H Mercedes S500 300.00  8 GB3221K Tata Magic 200.00  9 YB8283M Isuzu NPR 250.00  10 YK5133H Isuzu NQR 300.00 |

**Option 5 : Rent a vehicle**

|  |
| --- |
| Enter your option : **5**  Option 5. Rent A Vehicle  S/No IC Number Name Date of Birth Tel  1 S9099999A Adrian 22/03/1990 91112222  2 S8988888B Benjamin 11/08/1989 96668888  3 S7901234C Christina 13/12/1979 93335555  Enter S/No of customer : **2**  S/No Reg No Make Model Daily Rate ($)  1 SJV1883R Honda Civic 60.00  2 SJZ2987A Toyota Altis 60.00  3 SKA4370H Honda Accord 80.00  4 SKD8024M Toyota Camry 80.00  5 SKH5922D BMW 320i 90.00  6 SKM5139C BMW 520i 100.00  7 SKP8899H Mercedes S500 300.00  8 GB3221K Tata Magic 200.00  9 YB8283M Isuzu NPR 250.00  10 YK5133H Isuzu NQR 300.00  Enter S/No of Vehicle : **4**  Enter Child Seat Required : **1**  Enter Pick Up Date (DD MM YYYY) : **22 1 2015**  Enter Return Date (DD MM YYYY) : **25 1 2015**  Total rental charges is $270.00. |

|  |
| --- |
| Enter your option : **5**  Option 5. Rent A Vehicle  S/No IC Number Name Date of Birth Tel  1 S9099999A Adrian 22/03/1990 91112222  2 S8988888B Benjamin 11/08/1989 96668888  3 S7901234C Christina 13/12/1979 93335555  Enter S/No of customer : **2**  S/No Reg No Make Model Daily Rate ($)  1 SJV1883R Honda Civic 60.00  2 SJZ2987A Toyota Altis 60.00  3 SKA4370H Honda Accord 80.00  4 SKD8024M Toyota Camry 80.00  5 SKH5922D BMW 320i 90.00  6 SKM5139C BMW 520i 100.00  7 SKP8899H Mercedes S500 300.00  8 GB3221K Tata Magic 200.00  9 YB8283M Isuzu NPR 250.00  10 YK5133H Isuzu NQR 300.00  Enter S/No of vehicle : **9**  Driver Required (Y/N) : **Y**  Enter Pick Up Date (DD MM YYYY) : **22 1 2015**  Enter Return Date (DD MM YYYY) : **23 1 2015**  Total rental charges is $350.00. |

**Option 6 : List rental details for all cars or trucks**

|  |
| --- |
| Enter your option : **6**  Option 6. List rental details for all cars or trucks  Enter 1 for Cars, 2 for Trucks : **1**  Rentals for Cars:  Reg No Make Model Rental No Customer Pick Up Return  SJZ2987A Toyota Altis 3 Adrian 24/12/2014 26/01/2015  SJZ2987A Toyota Altis 5 Benjamin 30/12/2014 02/01/2015  SKM5139C BMW 520i 6 Christina 17/01/2015 18/01/2015 |

|  |
| --- |
| Enter your option : **6**  Option 6. List rental details for all cars or trucks  Enter 1 for Cars, 2 for Trucks : **2**  Rentals for Trucks:  Reg No Make Model Rental No Customer Pick Up Return  GB3221K Tata Magic 1 Benjamin 03/11/2014 04/11/2014  YB8283M Isuzu NPR 2 Adrian 05/12/2014 06/12/2014  YB8283M Isuzu NPR 4 Benjamin 09/01/2015 10/01/2015 |

## ACTIVITY PLAN

###### Suggestions for Getting Started

1. **Analysis**

1. Understand the program specification and the requirements before attempting the assignment.

*e.g. the relationships between the classes*

*the use of the attributes in each class*

1. **Program Design**

2. Work out the User Interface required for user input and suitable output.

1. Work out the main logic of the program using Object-Oriented programming techniques;

*i.e. use inheritance and association of the classes properly.*

1. You are required to use suitable classes appropriately for this assignment.

*Marks will be deducted for inefficient use of the classes or improper use of classes*

**c) Implementation & Testing**

1. Determine the order in which the classes are to be implemented (certain classes need to be implemented before other classes can be implemented).
2. Implement the classes **ONE** at a time.
3. You should implement the initialization methods in the application class that generate the customer and vehicles objects for the customer and vehicle arraylists respectively in the application class.

*Note: you may choose to download the application class template – VehicleRental.java that is already implemented with the methods that generate the customer and vehicle objects for the customer and vehicle arraylists respectively from MeL under Assignment.*

1. Test your program logic to make sure that it works as expected.

*You must prepare test data to see that your program works correctly. All data entry should be validated and illegal data entry should be highlighted to the user so that the user can enter correct data.*

## DELIVERABLES

You are required to submit your work in **2** stages :

**Stage 1 – Friday, Week 15 (30 Jan 2015) @ 8:45 AM**

* ALL the 6 classes (source files) shown in the class diagram and at least 3 basic features to your OOP Network folder (in the **assignment->stage1** folder)

In each of your .java file, you MUST include a *blocked comment* at the top stating your **student number, name**, and **group** as shown below:

//============================================================

// Student Number : **S10151234**

// Student Name : **John Tan**

// Module Group : **IT01**

//============================================================

*Note : 10 marks of the assignment will be deducted for any late submission.*

**Stage 2 – Friday, Week 16 (6 Feb 2015) @ 8:45 AM**

* ALL the classes (source files) that you have written for the whole assignment to your OOP Network folder (in the **assignment->stage2** folder)
* Complete test plan to your OOP Network folder (in the **assignment->stage2** folder)

*Note : NO submission will be accepted after the Walkthrough.*

## ASSIGNMENT WALKTHROUGH OF YOUR PROGRAM

The ***ASSIGNMENT WALKTHROUGH*** will be conducted on 6 February 2015 from 4.30 - 6.00 pm. In the walkthrough**,** you will be required to answer questions relating to the assignment. These questions will assess your basic understanding of the code that you have submitted. If you fail to display adequate understanding of your own program, your final grade for the assignment may be **down-graded by up to two letter grades** (e.g. from A to C).

You are required to bring your own program listings (i.e. hardcopy with line numbers, and **NO** **EXTRA** codes or comments) for the walkthrough and submit them at the end of the walkthrough.

Any EXTRA codes or comments not required in the assignment will be deemed as unauthorised materials.

**8. GRADING CRITERIA**

This assignment constitutes 40% of this module.

Performance Criteria for grading the assignment is as described below. Marks awarded will be based on **program code** as well as student’s degree of understanding of work done as assessed during the **walkthrough**.

**Grading criteria for the program is given below.**

***A Grade***

|  |
| --- |
| * Program implements the *Basic Features* successfully * Program implements all the basic *input validations* successfully * Program implements the three *Advanced Features* successfully * Program demonstrates good design with the correct use of methods * Program provides strong evidence of good programming practice * Program has been tested adequately, with submission of the test plan |

***B Grade***

|  |
| --- |
| * Program implements the *Basic Features* successfully * Program implements some basic *input validations* successfully * Program implements one *Advanced Features* successfully * Program attempts to use methods * Program provides sufficient evidence of good programming practice * Program has been tested adequately, with submission of the test plan |

***C Grade***

|  |
| --- |
| * Program implements the *Basic Features* successfully * Program provides some evidence of good programming practice * Program has been tested adequately, with submission of the test plan |

***D Grade***

|  |
| --- |
| * Program implements the *Basic Features* successfully * Program has been tested adequately * Score at least a ‘D’ in the walkthrough |

*NOTE*

* *Evidence of good programming practice include the use of meaningful variable names, proper indentation of code, appropriate and useful comments, adoption of standard naming conventions etc.*
* *Basic Input validation refers to the checking of the inputs entered by the user.*

*e.g. invalid option, invalid year, invalid month, invalid day*

**Appendix 1 – Class Diagram for Reliance Vehicle Rental System**

