

CSE105-CW3: Java programming project.

Due date: 11th Dec 2016 at 6pm. Worth 30% of final mark.

Design and code a Used Car System in Java.

This will be a menu based Used Car System, using standard input/output.

Joe's Garage buys and sells vehicles. They may buy a used car from a customer that has some damage, and improve its condition before they sell it. Joe's garage is a small family business and can only store 20 cars at any time.

The details of operation are flexible and left to your imagination. The main functions are:

- Add a new car to the system when it is purchased, noting the purchase price and date of purchase. A new car can only be purchased if there are currently less than 20 cars for sale.
- Sell a car to a customer, noting the sale price and date of sale. The car is marked as sold. Its details are kept in the system for use in sales reports.
- Display a list of all cars for sale sorted by date of purchase.
- Display a sales report showing all car sales. Show the purchase price, sale price. Show the total profit at the end of the report.

In addition, the program;

- Reads a list of cars from a text file at start up. The cars are stored in a suitable data structure in memory.
- Saves all cars to a text file just before the program exits.

Note, the storage in the text file must be true object persistence. Objects will be read from the file at the program commencement and stored prior to program termination. All operations on objects (cars) must be done in memory.

It is expected that you will have:

- At least one Java class to control the operation of the system.
- At least one Java class to store data as objects.

Use a package name of **CSE105** for java classes.

The expected level of comments in the Java code:

Typically, **one line of comment to describe each method**. Comments are not needed for getters and setters of data classes. Include your **name and student number as a comment at the top of each class**.

You will submit:

1. A report in the form of **one** Word document containing:
 - a. **System Design**. The detailed design of each class showing attributes and methods.
 - b. **Testing**. Show the test cases needed to test the system. Show the actual test results obtained by running the system.
2. **Implementation** (coding):
 - a. 1 file for each class (ie. .java files)
 - b. 1 text file containing the data.

You must pack all of your files with WinRAR or WinZIP, using your student ID as the file name and then upload one **rar/zip** file to ICE. Each student should also submit a hardcopy of the assignment. The content must be identical to the electronic version.

Note, the project must contain English characters only – no Chinese.

This assignment is individual work. Plagiarism (e.g. copying materials from other sources without proper acknowledgement) is a serious academic offence. Plagiarism will not be tolerated and will be dealt with in accordance with the University Code of Practice on Assessment.

Marking scheme for CW3 project.

Assessment criteria	Percentage
A design document is provided. Class names, main attributes, and behaviour are shown. The overall design of the classes and methods within are properly presented. OO principles are observed including encapsulation and appropriate use of static and object methods. Sufficient detail is shown to enable implementation. Do not use Java code in the design. Pseudocode or equivalent should be used.	20
Test cases. Test cases are provided to test all functions of the system.	10
Test results are properly presented showing expected and actual results for all test cases.	10
Main functions work. 4 functions x 10 marks.	40
Data storage in text file. Data retrieval, data storage.	10
Menu navigation. It is possible to navigate all functions and also exit.	10
Total	100%

Deduct 5 marks for each of the following:

- Compiler errors
- The use of break or continue (except break within switch).
- Identifier naming not conforming to commonly accepted Java programming conventions.
- Insufficient comments in the Java code.