

## Term Project

### Inventory Control System

Grade weight: 10%

Due: 23/4/2022

#### Problem Description

An engineer keeps an inventory of car parts, but on paper. The usual operations on the 'inventory database' include adding new parts, updating levels of current stock and producing different reports on the current stock levels. As the inventory is getting larger, this manual system is becoming difficult to manage. The engineer decided to develop a computerized system to help manage the inventory. Being not familiar with database management systems, the engineer decided to develop an application, based on the Python knowledge received during the early university education. As a first step, the engineer use MS-Excel to create the inventory table, similar to the one below. This was chosen because MS-Excel can save the inventory sheet as a CSV file. The engineer is already familiar with CSV files from the Computer Programming course.

PartNo	PartName	Make	Model	Quantity	Price
1	BLADE W/S WIPER	Pajero	NA1H25	26	3.65
2	OIL SEAL-T/M CASE	Pajero	NA1S25	12	65.7
3	OIL SEAL-DIST	Pajero	NA3H25	20	14.6
4	DISC-CLUTCH	Pajero	NA3H25	26	10.95
5	FUSIBLE LINK	Pajero	NC3V25	13	14.6
6	WEATHERSHIELD PKGE-L	Pajero	ND0000	12	3.65

#### 1. Requirements

You are required to help this engineer by developing the Inventory Control System. Your system should perform the following operations:

- Add a new item. Write a function called `addItem()` that asks the user to input data of a new part and then add the data to the inventory file.
- Update stock level. Write a function called `update_item()` that takes as parameter the `partNo` if the part to be updated. The function prompts the user to enter the quantity to be updated. Check that enough items are there if the operation is a remove operation (i.e. if the quantity is negative). The price could also be updated by the function.
- Remove an item. Write a function called `delete_item()` that takes as parameter the `partNo` to be removed.
- List the current inventory in a nice tabular format. Monetary values should be shown to the nearest 2 decimal positions. In the last column of this table show the total price of each item (i.e.  $\text{price} \times \text{quantity}$ ).
- Produce lists of the following:
  - Re-order list, which is a list of items whose quantity field  $< 4$ .
  - All items whose price is greater than a value provided by the user.
  - All items whose price is greater than the average item price.
  - Stock statistics such as: the current value of the stock (sum of  $\text{price} \times \text{quantity}$ ), the average price, the most expensive item, the least expensive item, the least stocked item.

**Note:**

1. All prices are in QR.
2. The `inventory.csv` file will be given. Get a copy from the Lab Materials folder on BB.
3. Properly format your output.
4. Validate user's input.
5. Whenever possible, reuse functions.
6. The project's main function should provide a menu that will allow the user to choose one of the tasks as shown below. The program exits when the user selects option 9.

The following figure shows a typical session with the Inventory system.

```
Inventory Control System
[1] Add part
[2] Update part
[3] Remove part
[4] Display inventory
[5] Display re-order list
[6] Parts above a given price
[7] Parts above the average price
[8] Stock statistics

[9] Exit

Enter your choice: 1

Stock Control - Add Item:
=====
Enter the car make :Toyota
Enter the car model: Corolla
Enter partNo: 846
Enter part name: BRAKE SHOES
Enter price: 250
Enter quantity: 20

Save (Y/N): Y
.....Record saved.

Inventory Control System
[1] Add part
[2] Update part
[3] Remove part
[4] Display inventory
[5] Display re-order list
[6] Parts above a given price
[7] Parts above the average price
[8] Stock statistics

[9] Exit

Enter your choice: 9
```

## 2. Submissions:

- The due date for submission is end of the day on 23/4/2022 (on Blackboard. NO email submission, please).
- Submissions late by up to three days will receive a 10% penalty per day. No submissions overdue by more than three days will not be accepted.
- Your group should have **three** members. Please inform your lab instructor immediately when you form your groups. If the number of students in a section is not divisible by three, then your instructor may exceptionally decide to make a group with two or four students.
- Each group should prepare a report, which is to be submitted by each team member, as explained below. The report should include an introduction, description of the problem and clear documentation of the used formulas, if any. Also, document each function showing its inputs, preconditions, return values ... etc. Your report must include a list of references used. Include a copy of all the developed code as an appendix. Together with the report, you are required to prepare a file archive containing your Python program file(s) and the updated inventory.csv file.
- Each member of the group **MUST** submit a copy of the report on Blackboard. Each member submission must include a cover page, containing peer evaluation. The cover page of your submission MUST contain the following items:

**Computer Programming – GENG 106**  
**Term Project – Spring 2022**  
**Inventory Control System**

Submitted by: \_\_\_\_\_ QUID: \_\_\_\_\_

[Each member of the group will be required to submit a peer group evaluation with an estimate of the contribution from each member to the project. Write the percentage contribution made by each team member so that it adds up to 100%. This evaluation *may* be used to adjust the marks awarded to each team member]

- Effort distribution of the student:

QUID:_____	STUDENT NAME: _____	Effort given_____%
QUID:_____	STUDENT NAME: _____	Effort given_____%
QUID:_____	STUDENT NAME: _____	Effort given_____%
QUID:_____	STUDENT NAME: _____	Effort given_____%

- Course number \_\_\_\_\_
- Submission date \_\_\_\_\_
- Section CRN: \_\_\_\_\_

- **DECLARATION:** *We hereby certify that no part of this project or product has been copied from any other student's work or from any other sources except where due acknowledgement is made in the project. No part of this project/product has been written/produced for us by any other persons.*

*Note: If the above submission requirements are not met, up to 50% marks will be deducted from the project.*

- Be aware of: *Submitted work must be students' own work*

### 3. Evaluation criteria:

Grading %	Criteria
70	Complete and correct implementation of the project (50% for functions, 10% for file output, and 10% for the main function)
5	Validation of inputs and exception handling
5	Good programming style and comments + Choice of meaningful names for variables and functions.
5	Project materials (softcopy): including print out of the project source programs (*.cpp), and snap shots of all project screens (user interfaces), in one Word file.
15	Each group will present and demo their implementation in 10 minutes during the lab session
-100	Plagiarism
-100	Not presenting/demonstrating your work
100	<b>Total</b>