# Introduction

This report details the solutions achieved using a procedural programming approach and an object-oriented approach to programme the shop assignment as part of the Multi-Paradigm Programming module.

## Folder structure & files explained

This section of the report details the folder structure and provides a brief explanation of the files included in the folder. The main project folder is titled:

* “G00387815\_Keith\_Quinn\_Shop\_Assignment”.

Within the main folder there are two sub folders:

1. Shop\_in\_C (ignore this folder for this report).
2. Shop\_in\_Python (only this folder is considered in this report).

Within the Shop\_in\_Python folder there are 2 files:

1. shop\_OOP.py
2. shop\_procedural.py

These are the two files that are of interest for this report. In the next section, both the object-oriented programming (OOP) and the procedural implementations of the shop assignment are discussed.

Also included in the main project folder are the following 4 excel files:

1. stock.csv (file used to stock the shop in all implementations).
2. A.csv (customer A file to demonstrate when the shop hasn’t enough stock).
3. B.csv (customer B file to demonstrate when the customer hasn’t enough money).
4. C.csv (customer C file to demonstrate when the order can be fully processed).

# Object Oriented and Procedural Implementations

Both the OOP and the procedural implementations use the same menu to provide identical “user experience” as per the assignment specification. Figure 1 shows the menu that is displayed when either implementation is started.

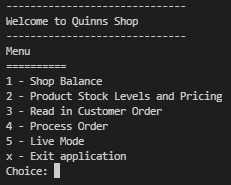


Figure 1 OOP and procedural "Menu"

When making comparisons between OOP and procedural paradigms one of the fundamental differences is that in OOP the class of the object has both state and functionality, unlike procedural which only has state. Consider Figure 2 and Figure 3 below

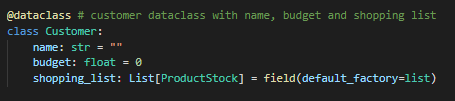


Figure 2 Customer class in procedural

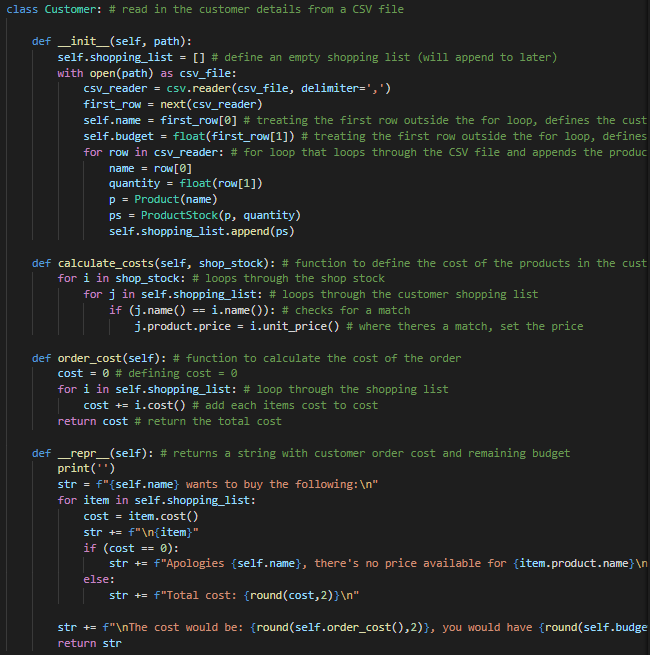


Figure 3 Customer class (part of) in object oriented

With reference to Figure 2 and Figure 3, in the procedural implementation the Customer class contains the state only (name, budget and shopping list), whereas in the object-oriented implementation the Customer class contains state and functionality. The functionality reads in the customer details from a csv file, calculates the costs and outputs information.

Considering that Customer is a class, or template there can be many instances of it. In the shop assignment there are 3 instances of customer. Each of the 3 customers have different names, budgets, and shopping lists but all 3 use the Customer class.

The approach from here is to look at each of the 5 options in the menu and discuss the OOP and procedural implementations.

## 1 – Shop Balance

The assignment specification states, *“The shop CSV should hold the initial cash value for the shop”.* For both implementations “Option 1” in the menu displays the cash value. For this the same file “stock.csv” is used.

In the procedural implementation there are 2 functions “stock\_shop()” and “print\_shop\_balance()”. In the OOP implementation there is one method call to “shop.shop\_balance()”. In the procedural implementation “Shop” is a separate class that’s used within the “stock\_shop” function whereas within the OOP implementation the “Shop” class includes state and function.

## Product Stock Levels and Pricing

Selecting Option 2 from the menu returns the product stock levels and pricing for the shop. This was not required per the assignment but felt it was worthwhile as it shows what’s available and how much is costs. In the procedural implementation there’s a “print\_shop()” function. This function combined with the stock\_shop() is used. In the OOP implementation this is much easier as the existing “Shop” class is used where there’s a “stock\_levels\_pricing(self)” method.

## Read in Customer Order

The assignment specification states, *“Read in customer orders from a CSV file.*

* *That file should include all the products they wish to buy and in what quantity.*
* *It should also include their name and their budget.”*

## Process Order

The assignment specification states,

## Live Mode

The assignment specification states,