Python Assessment – Dillon Constantine

GitHub - <https://github.com/dconstantine22/Python_Assesment_Dillon_Constantine>

# Abstract

This report explains the process of demonstrating the core learning outcomes of python programming. Throughout this report and the accompanying code, the aim was to use various examples to show the understanding of the main areas within the python programming language.

# Introduction

Python is ranked as the top computer languages worldwide with almost 30% of the programming population being users according to PYPL (Statistics Times, 2022). With the popularity of the python programming language growing, the aim of this project is to demonstrate the key knowledge areas by implementing examples of each.

The core learnings that are demonstrated in this project are;

* Functions
* Boolean-Logic (Loops and Conditions)
* Data Structures
* Object Orientated Programming (OOP)
* Error & Exception Handling

# Implementation Process

This section of the report will step through the process of demonstrating the core learning outcomes of Python programming. All the examples talked about below can be seen in the accompanying python script.

## Functions

Functions are a block of reusable code that is used to perform specific tasks. Python has many built-in functions that have been predefined and can be called upon easily. Some examples of these built-in functions are sum(), max(), abs() and sort(). These built-in functions are used throughout the code of this project.

Python also allows for users to create their own functions based on tasks they need to perform. In this project to demonstrate the use of functions the following were implemented as examples. Functions like the below can be built for easy recall within scripts and for code efficiency.

* **Welcome function:** This function can be called to say ‘Hello’ and welcome to any user. The user is asked to enter their name and it will print out a welcome message to the specific individual.
* **Calculation function:** This function once called allows for the user to perform a quick calculation by entering any three numbers (a, b & c).
* **Circumference of a circle function:** This function asks the user to enter the radius of there circle and the function calculated the circumference.
* **Bill Plus Service Charge function:** This function automatically generates the total value of a bill including a 10% service charge. The user needs to enter the bill amount and the function will add on the service charge and return the total.
* **Even Number Check function**: This function when called checks if a number is even or odd.

The reason for implementing the functions within this project was to demonstrate a variety of different uses of functions from a simple welcome message to getting the total of a bill plus the tip given the amount.

## Boolean-Logic (Loops & Conditions)

Boolean Logic is a way of finding out the truth of an expression by simply using true or false concepts. To demonstrate this in a straightforward way the ‘AND’ and ‘OR’ operators were implemented. For example, ‘True’ and ‘True’ equals ‘True’ whereas ‘True’ and ‘False’ returns ‘False’ as both values needed to be true. This demonstrated simply how Boolean logic works.

Next was the implementation of conditioning (if-else) where two examples are used to demonstrate an understanding. The first example was the drinking eligibility of the user, and the second example is based on the tyre type on F1 cars. The reason behind the use of these two examples is to demonstrate conditioning on both string and integer data.

Finally, to demonstrate Boolean logic, loop examples where implemented; one to print out a range of values and the second to loop through a list of brands. As the loop iterates through the values when it is returned ‘True’ the value is then printed. The examples used for both conditioning and loops demonstrate an understanding of this core area of python programming.

## Data Structures

To demonstrate an understanding of the core data structures, lists and dictionary examples were implemented. These examples were picked to show the different ways in which data can be organised and the variety of structures to store data. By using the separate data structures, it shows the difference between lists and dictionaries. Lists are an orders sequence of objects that are called via indexed whereas dictionaries are unorders sets that use keys to access items.

In this project lists were implemented first, starting with a list of sports and then a list of numbers to show that lists can store different data types. Once the lists were created several functions are used to demonstrate an understanding. Functions such as type(), len(), append() and min, max() on the integer list.

Dictionaries were implemented next to demonstrate another example of a data structure. Again, two examples were created; a GAA teams’ example and a dog’s details, and different functions used to manipulate dictionaries. Through this implementation the different ways to slice the data can be seen based on the data structure being used so understanding the structures is important when using python.

## Object Orientated Programming (OOP)

To demonstrate an understanding of OOP in this project the class of an electric car was created. To begin, the class car was created, the properties established, and the methods defined. For demonstration, there was 6 methods created for class car;

* + Drive: Each time the car drives it decreases the battery by 10%.
  + Air Conditioning: Decreases the battery by 1%.
  + Charge: Increases the battery by 20%.
  + Crash: Reduces price of car by €5000.
  + Repair: Increases price by €1500
  + Respray: Allows the car to be painted a different colour.

The cars themselves where the objects and three cars were implemented to demonstrate OOP. The implementation of these objects show that the methods can be called to update the various properties.

## Handling Various error messages & Exceptions

Throughout this project several error/exception messages were returned so handing these is important in python programming. In this project the try, except method of exception handling is used for examples; ValueError, ZeroDivisionError and KeyError. Implementing the try, except code to catch and return easily understandable messages to the users, simply demonstrates an understanding of exception handling in python programming.

# Bibliography

Statistics Times, 2022. *Top Computer Languages.* [Online]   
Available at: https://statisticstimes.com/tech/top-computer-languages.php  
[Accessed 28 July 2022].