*We will be using PyCharm as an environment to code python in; as it allows you to create a virtual environment of a new python project (hence we downloaded python initially)*

**Contents Page**

**[1] Testing PyCharm environment**

**[2] The 2 ways to comment on Python**

**[3] Data Types**

**[4] Variables**

**[5] More On Strings**

**[6] Control Flow**

**[7] Collections**

**[8] Loops**

**[9] Functions**

**[10] The code for a basic calculator**

**[11] The code for a scrabble calculator**

**[1] Testing PyCharm Environment**

How to test if your PyCharm virtual environment is working correctly;

You can run a print code on it, e.g., print(“hello world!”), If it displays Hello World! Upon running, the environment is functioning correctly.

**[2]** **The 2 ways to comment on python**

The 2 ways to comment:

a) via a hash symbol #

b) via a “”” comment “””

**[3] Data Types**

**- There are mainly 3 types of data types:**

1. **Numbers;**
2. **Integers; which are whole numbers**
3. **Floating point numbers; which are numbers with decimals**
4. **Strings**

**Anything which contains characters/text**

1. **Boolean**

**A diagram of a basket and numbers

AI-generated content may be incorrect.Something which is either true or false**

**- How to ask python the nature of a number data type (integer or float) or if it’s a Boolean:**

To do this, we run commands

Print(type(5)) – Should output int

Print(type(5.5)) – Should output float

Print(type(true)) – Should output boolean

- **Types of Operators:**

+ Plus

- Minus

\* Times

/ Divide

% Modulus – returns the remainder of a division

**- ‘ or “ when presenting a string?**

It simply doesn’t matter, I can have either:

Print (‘hello world’)

Print (“hello world”)

- **Python doesn’t read text, it reads a group of Unicode’s; Each letter is assigned as a Unicode:**

E.g., another way to print a, other than the normal method: print(‘a’)

Print(u’\u0061) will print a *(https://www.compart.com/en/unicode/U+0061)*

**- How to get the characters length in a string:**

Use the ‘len’ function

e.g., print(len(‘hi’)) will print 2, as hi contains only 2 characters

**- Boolean Works usually works with equality operators:**

(i)Identical Symbol (==)

E.g., True – 4 == 4 E.g., False – 3 == 4

(ii) Not Equal Symbol (!=)

E.g., True – 3 != 4 E.g., False – 4 == 4

(iii) Greater than (>)

1. Greater than or equal to (>=)
2. Smaller than (<)
3. Smaller than or equal to (<=)

\* IMPORTANT NOTE; you can’t compare a string with a number, will come with an error (e.g., 2 <= “3” – will give back an error)

\* IMPORTANT NOTE; python will only recognise an empty strong, or 0 as false, everything else is true. [(e.g., print(bool(1))] – true [(e.g., print(bool(hi))] – true

[(e.g., print(bool(0))] – false [(e.g., print(bool(‘’))] – false

\*NEED TO FIND OUT ABOUT Q2 ON THIS, SHOULD ONLY BE ONE WRONG ANSWER AS RANGES ARE 0 TO 28 AND -1 TO -29, SO WHY WOULD -29 GIVE OF AN ERROR)