Python Review

Review of everything we have covered so far.

Variable

- How do we define a variable
- Give a name and use ‘=‘ to give it a value.

How Do we print something

- Two ways
  - Just type the variable name in the notebook
  - Use Print function
    - print(variable_name) or print(“String to be printed”)

Data Types

- String
- Integer
- Float
- List
  - List of other data types like string or numbers.
- Boolean

- There are few more as well which we will learn later on

How to check variable type

- Use the function
  - type(variable_name)

Strings

- What is a string , anything enclosed in single quotes or double quotes
- example

string_var = “This is a string”
string_var1 = ’This is also a string’

We prefer using “ instead of ‘

- To access characters in String we use square brackets with index []

Strings Access

- str_hello = “HEY”
- H is at index 0
- E is at index 1
- Y is at index 2

- To access Y we can do str_hello[2]
- Reverse Index, we can also use negative index which is taken from the end.
- For example to access Y we can also do str_hello[-1] which means 1 character from the end.
- We can also access part of strings using str_hello[start_index:end_index] ex, str_hello[1:3]

Numbers

- Integer
  - Whole numbers
- Float
  - Decimal numbers

- Can do mathematical operations on it.

- Add (+), Subtract(-), Divide(/), Reminder (%), Power(**)

Lists

- Collection of other data types like numbers and strings and even lists.
- Defined as
  - var_list = [] empty list
  - var_list_nums = [1,2,3]
  - var_list_strs = [“Hi”, “Hey”, “Hello”]
- Starts with [and ends with] and elements in a list are comma(,) separated.

List Access

- You can access elements of list like you access characters in string.
  - Using index and []
- Define a list , list_var = [“One”, “Two”, “Three”, “Four”]
- To get string ”One” we can do list_var[0]

What brackets to use

- Generally [] are used to define lists or access elements / characters with index.
- Use () when we have functions.
- Functions are code that's already given to you like tools.
  - Functions can be global , that is can be called directly like len() or type() or print()
    - These functions generally take a value
  - Or functions can be called on a data type like List or String.

List Functions

Define a list , list_var = [1,2,3,4,5,6]

- Append - add something to the end of the list , list_var.append(0)
  - list_var now becomes [1,2,3,4,5,6,0]
- Remove – removes a random occurrence of value passed if more than one exist.
  - list_var.remove(1), list_var now becomes [2,3,4,5,6,0]
- Insert- adds a value at a particular index
  - list_var.insert(1, 0), list_var now becomes back to [1,2,3,4,5,6,0]

Input function

- Input function takes a value from the user, generally its used to store a input from the user to a variable.

- Used like

  - var_input = input(“Enter your name ”)
  - It will bring something like : Enter a number of your choice: Aman
  - Now var_input value will be “Aman”

Boolean

- Data type that has only two values True or False
- Used to store values from comparisons

Boolean Comparisons

- One value == Second value, compares the two value and gives True or False
  - 2 == 2 , is True whereas 2 == 3 is False
- != is read as not equal to
  - 2 != 3 is True whereas 2 != 2 is False
- Similarly we can > (greater than), < (less than), >=(greater than or equal to), <=(less than equal to) to do comparison
- we can combine several comparisons as well with two keywords
  - and – means all comparisons need to be true, ex
    - 2 < 3 and 3>7 will give True
    - But 2 > 3 and 3<7 will give False
  - or – means even if one of the comparisons is true the whole thing is true ex
    - 2 != 2 or 3 > 7 or 1< 9 is True since 1<9 is True, so entire thing becomes True

Control Statements

- Control statements allow us to write function code like
- If you are hungry -> Eat food, else, go play
- Another example
- If light is off -> turn on the switch, else -> do nothing

Control Statement – If / else

- Indentation is very important for if and else in python

- If uses a Boolean variable or expression to go in

- Example

```
var_word = “WORD”

if var_word == “WORD” :
    print(“We found our word”)
else:
    print(“Try again please”)
```

- In this example, we compare if the variable var_word is same as the string “WORD” , if it is we print “We found our word” , see we use colon(:) after if and even else