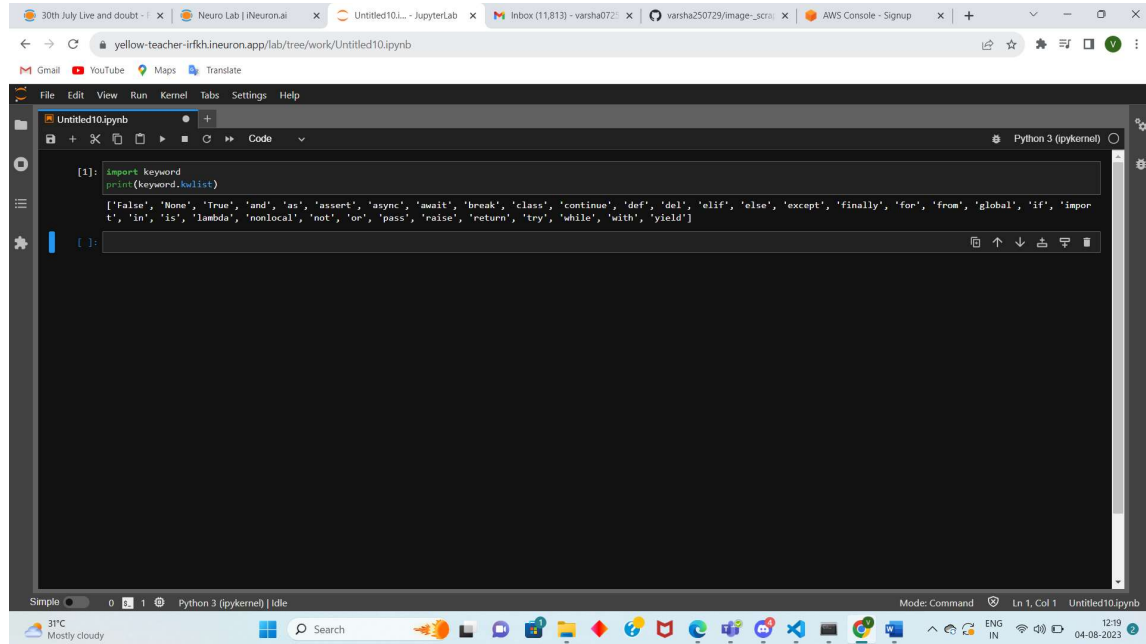


# Assignment -6

Q.1. What are keywords in python? Using the keyword library, print all the python keywords.

**Ans1:-** Keywords in Python are reserved words that have predefined meaning ,that cannot be used as ordinary identifiers. They are used to define syntax & structure of the python language. Keywords are immutable.

A screenshot of a JupyterLab interface. The top bar shows several open tabs: '30th July Live and doubt', 'Neuro Lab | iNeuron.ai', 'Untitled101... - JupyterLab', 'Inbox (11,813) - varsha072...', 'varsha250729/image\_scri...', and 'AWS Console - Signup'. The main area displays a Jupyter notebook with the following code in the first cell: 

```
[1]: import keyword
print(keyword.kwlist)
```

The output of the code is a list of 33 Python keywords: 

```
['False', 'None', 'True', 'and', 'as', 'assert', 'async', 'await', 'break', 'class', 'continue', 'def', 'del', 'elif', 'else', 'except', 'finally', 'for', 'from', 'global', 'if', 'import', 't', 'in', 'is', 'lambda', 'nonlocal', 'not', 'or', 'pass', 'raise', 'return', 'try', 'while', 'with', 'yield']
```

Q.2. What are the rules to create variables in python?

**Ans 2:-** There are following rules:-

1. Variables are declared by writing the variable name & assigning it a value using the equal sign.
2. Variable name consist of letters, digit & underscore(\_).They must start with a letter or an underscore. Variable names are case sensitive.
3. Variable names cannot start with a digit.
4. Variable names cannot contain spaces or special character like ! ,@,#,\$ etc.
5. We cannot use reserved keywords as variable names.
6. Variable names cannot contain spaces.

Q.3. What are the standards and conventions followed for the nomenclature of variables in python to improve code readability and maintainability?

**Ans3:-** Following are some standards and conventions:-

1. Use lowercase letter for variable names & separate words with underscore.
2. Choose variable name that are meaningful & describe the purpose of variable.
3. If variable's value is a constant & should not be changed , use uppercase letter & underscore.
4. Avoid using a variable name that shadow a built in function.

Q.4. What will happen if a keyword is used as a variable name?

**Ans4:-** If we use keyword as a variable name ,we will encounter a syntax error.

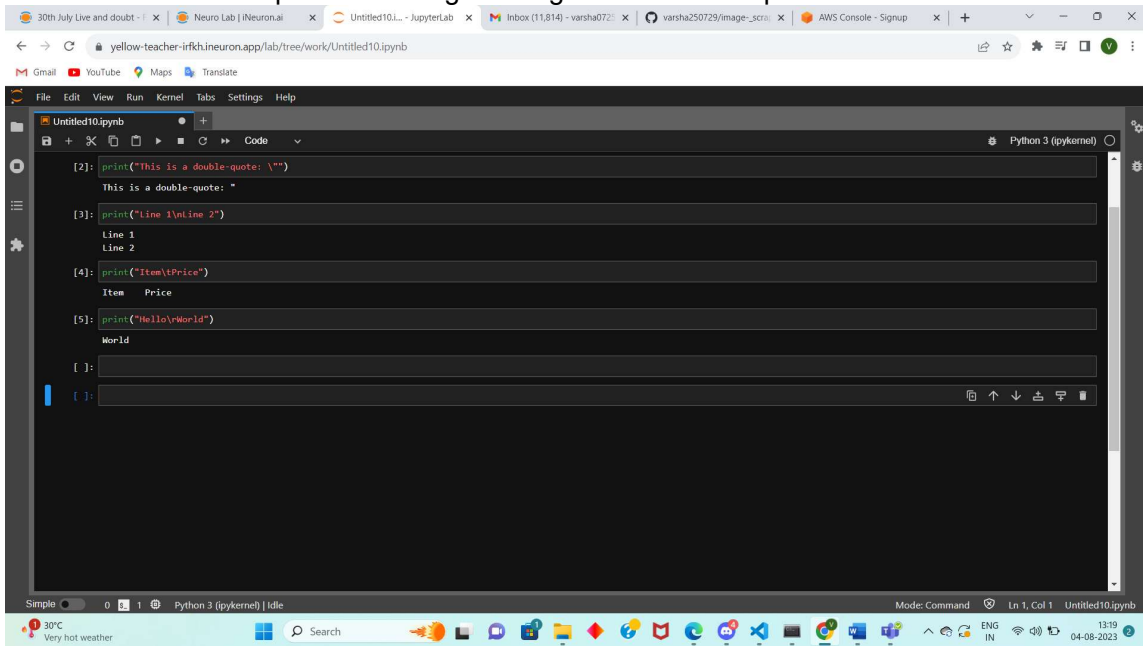
Q.5. For what purpose def keyword is used?

**Ans5:-** def Keyword is used to define a function. A function is a block of code that perform a specific task & can be reused throughout the program.

Using def keyword allow us to create our own custom function & encapsulate code into organized & reusable component.

Q.6. What is the operation of this special character '\'?

**Ans6:-** The special character '\' is called backslash or escape character. It is used to escape character that have special meaning in string & to introduce special character.

The screenshot shows a web browser window with a JupyterLab interface. The browser's address bar shows a URL from 'yellow-teacher-irfkh.neuron.app'. The JupyterLab interface has a menu bar (File, Edit, View, Run, Kernel, Tabs, Settings, Help) and a toolbar. The main area displays a Python notebook with the following code cells:

```
[2]: print("This is a double-quote: \"")  
This is a double-quote: "  
  
[3]: print("Line 1\nLine 2")  
Line 1  
Line 2  
  
[4]: print("Item\tPrice")  
Item    Price  
  
[5]: print("Hello\nWorld")  
Hello  
World  
  
[ ]:  
[ ]:
```

The bottom status bar indicates 'Python 3 (ipykernel)' and 'Ln 1, Col 1'. The Windows taskbar is visible at the bottom with various application icons and a system clock showing 13:19 on 04-08-2023.

Q.7. Give an example of the following conditions:

- (i) Homogeneous list
- (ii) Heterogeneous set
- (iii) Homogeneous tuple

**Ans7:- Homogeneous list:-** It is a list where all elements are of same data type. It is an ordered collection of items.

**Ex:- Homogeneous list of integer:-** `int_list = [2,4,6,8,10]`

**Heterogeneous set:-** It is a set where the element can be of different data type. It is an unordered collection of items where every element is unique.

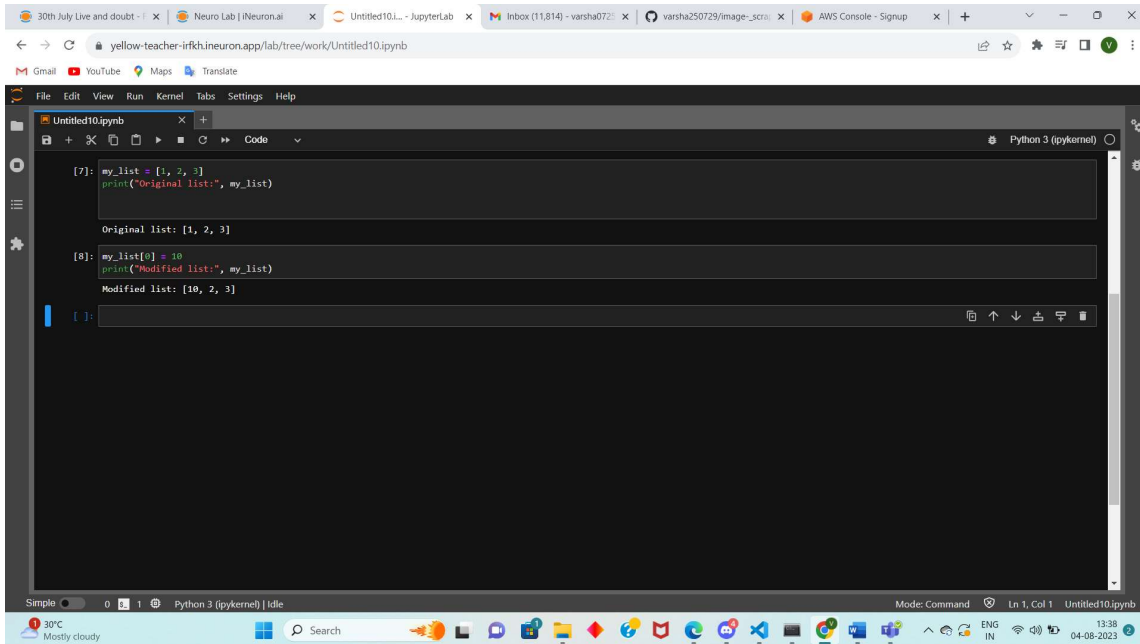
**Ex:- hetero\_set = {1,"apple",2.18,True}**

**Homogeneous tuple:-** It is a tuple where all elements are of the same data type.

**Ex:-** `string_tuple=("apple","banana","cherry","grapes")`

Q.8. Explain the mutable and immutable data types with proper explanation & examples.

**Ans 8:- Mutable Data Type:-** Mutable Data Type are those whose value can be modified after they created . It means we can change the internal state of an object without crating a new object.



The screenshot shows a JupyterLab notebook with two code cells. The first cell creates a list `my_list = [1, 2, 3]` and prints it. The second cell modifies the first element `my_list[0] = 10` and prints the list again. The output shows the list changing from `[1, 2, 3]` to `[10, 2, 3]`, demonstrating that lists are mutable.

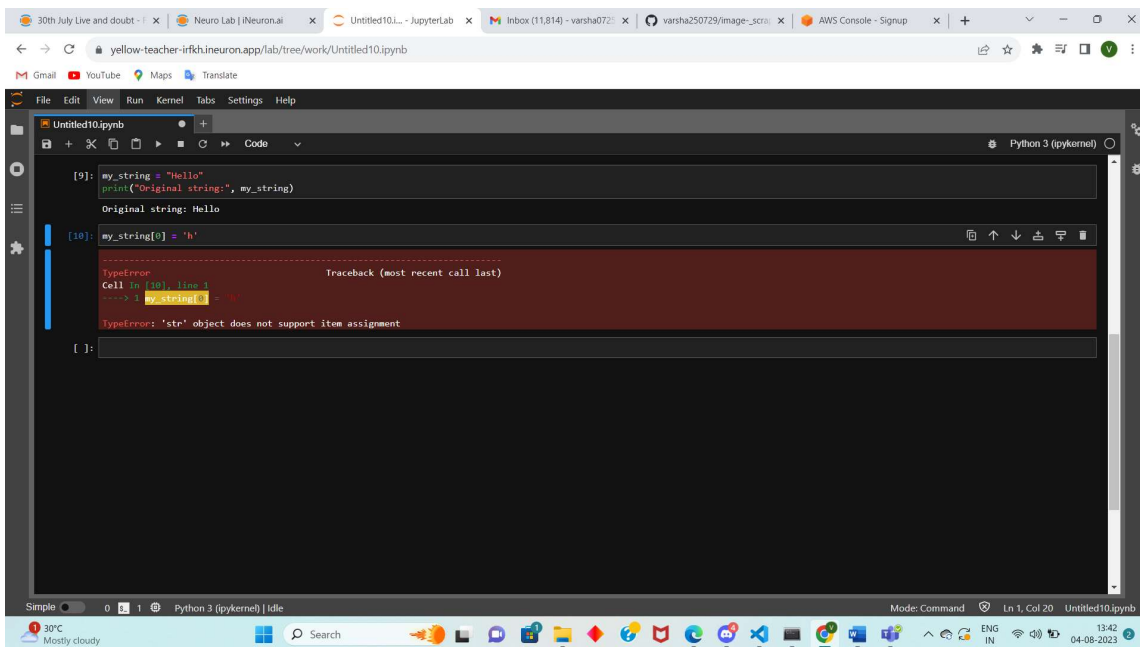
```
[7]: my_list = [1, 2, 3]
    print("Original list:", my_list)

Original list: [1, 2, 3]

[8]: my_list[0] = 10
    print("Modified list:", my_list)

Modified list: [10, 2, 3]
```

**Immutable Data Types :-** Immutable Data types are those whose values cannot be modified after they created. If we need to change the value, we have to create a new object.



The screenshot shows a JupyterLab notebook with two code cells. The first cell creates a string `my_string = "Hello"` and prints it. The second cell attempts to modify the first character `my_string[0] = 'h'`, which results in a `TypeError: 'str' object does not support item assignment`. This demonstrates that strings are immutable.

```
[9]: my_string = "Hello"
    print("Original string:", my_string)

Original string: Hello

[10]: my_string[0] = 'h'

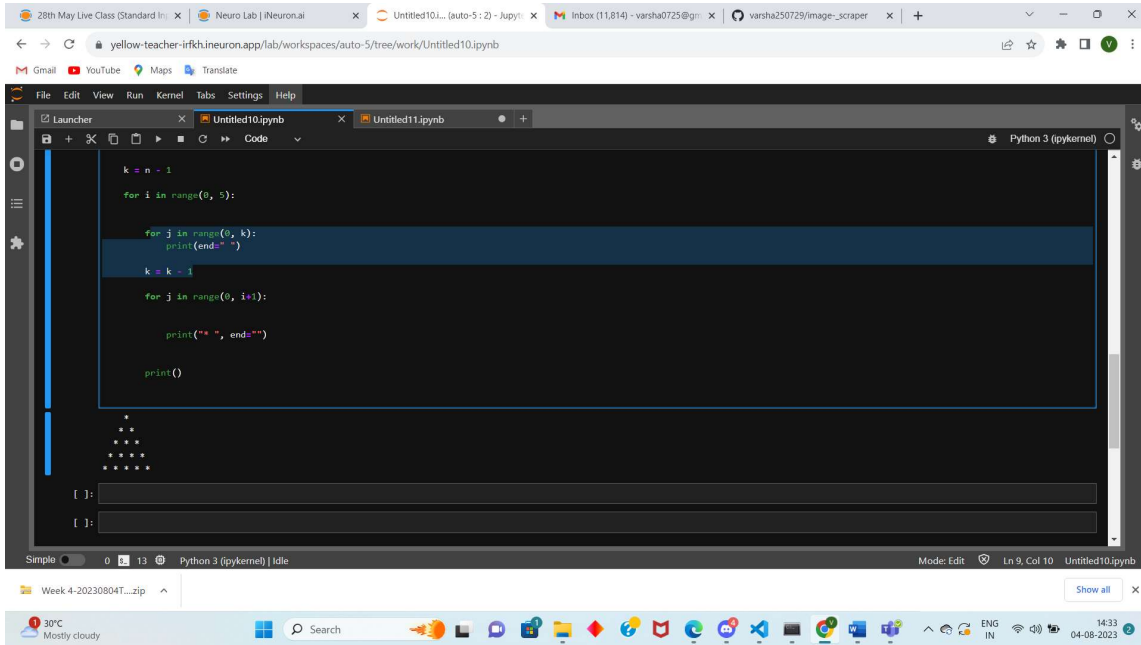
TypeError                                 Traceback (most recent call last)
Cell In [10], line 1
----> 1 my_string[0] = 'h'

TypeError: 'str' object does not support item assignment
```

Q.9. Write a code to create the given structure using only for loop.

```
*
***
*****
*****
*****
*****
```

Ans:-



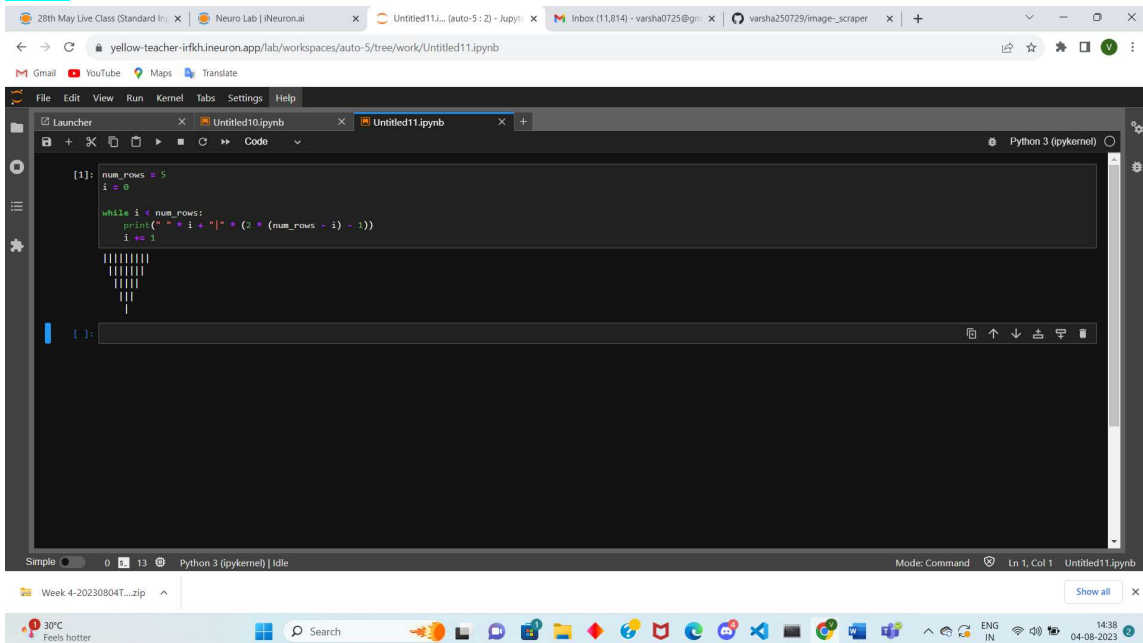
```
k = n - 1
for i in range(0, 5):
    for j in range(0, k):
        print(end=' ')
    k = k - 1
    for j in range(0, i+1):
        print('*', end=' ')
    print()

*
***
*****
*****
*****
```

Q.10. Write a code to create the given structure using while loop.

```
|||||
|||||
|||||
|||
|
```

Ans:-



```
num_rows = 5
i = 0
while i < num_rows:
    print('*' * i + '|' * (2 * (num_rows - i) - 1))
    i += 1

|||||
|||||
|||||
|||
|
```