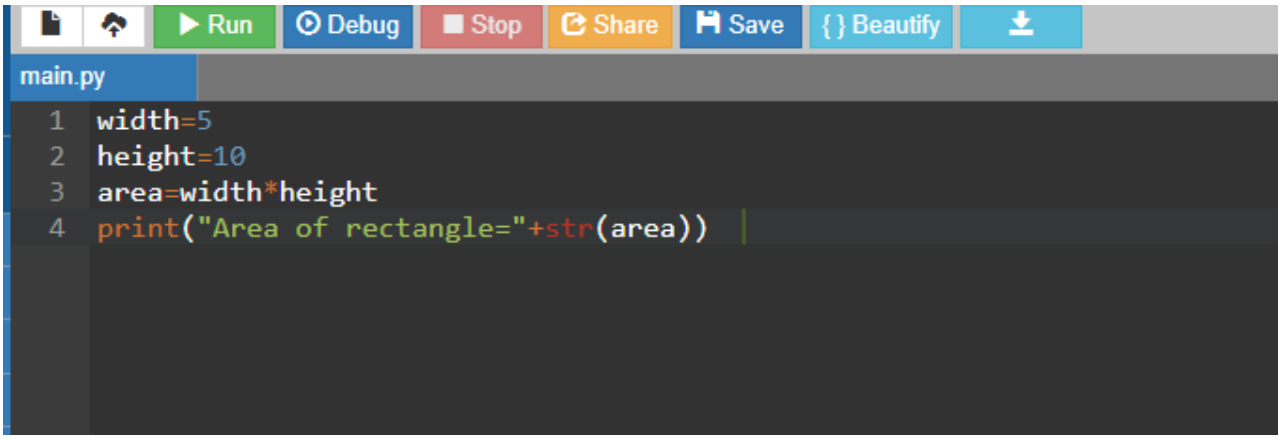
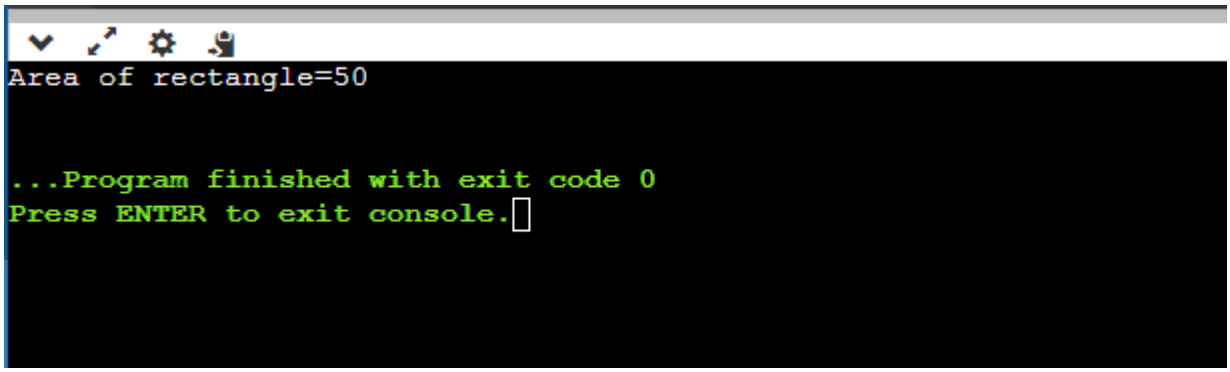
Assessment-1

1. Write a Python program to calculate the area of a rectangle given its length and width.

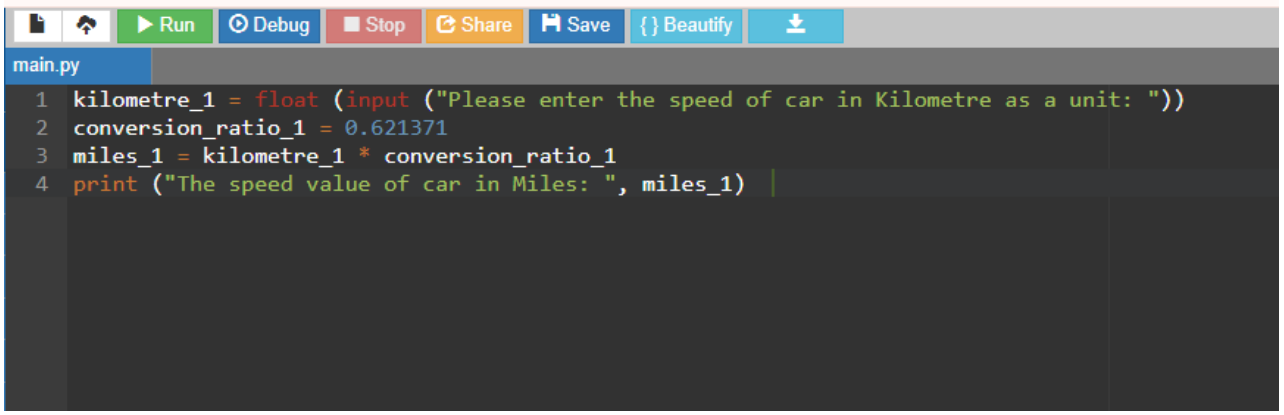
Ans.





2. Write a program to convert miles to kilometers

Ans



```
input
Please enter the speed of car in Kilometre as a unit: 20
The speed value of car in Miles:  12.42742


...Program finished with exit code 0
Press ENTER to exit console.
```

3. Write a function to check if a given string is a palindrome.

Ans.

Run | Debug | Stop | Share | Save | { } Beautify

main.py

```
1 string=input(("Enter a letter:"))
2 if(string==string[::-1]):
3       print("The letter is a palindrome")
4 else:
5       print("The letter is not a palindrome")
```

```
Enter a letter:madam
The letter is a palindrome


...Program finished with exit code 0
Press ENTER to exit console.
```

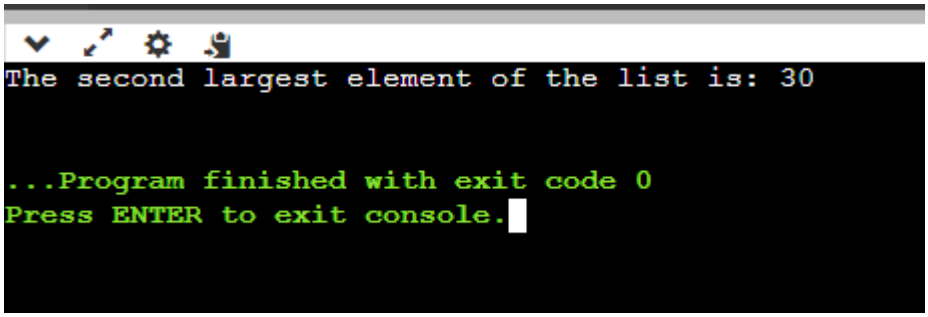
4.Write a python program to find the second largest element in a list.

Ans.

Run | Debug | Stop | Share | Save | { } Beautify

main.py

```
1 list_val = [20, 30, 40, 25, 10]
2 list_val.sort()
3 print("The second largest element of the list is:", list_val[-2])
```



5.Explain what indentation means in python.

Ans. Indentation is used to define the structure and hierarchy of code. It serves as a way to group blocks of code together. Unlike many other programming languages that use curly braces {} or keywords like begin and end to denote code blocks, Python uses indentation.

Here's what indentation signifies in Python:

Block Structure: Indentation is used to define blocks of code such as loops, conditional statements,function definitions, and class definitions. Each level of indentation represents a nested block.

Readability: Python places a strong emphasis on code readability. By enforcing indentation, Python code tends to be more readable and visually organized.

No Delimiters: In Python, there are no explicit block delimiters like curly braces {}. Instead, indentation is used to determine the start and end of blocks.

Consistency: Since Python requires consistent indentation, it enforces a consistent coding style across projects and among developers. This reduces the likelihood of syntax errors and improves code maintainability.

Whitespace: Python is sensitive to whitespace, so the spaces or tabs used for indentation must be consistent throughout the codebase. Mixing spaces and tabs can lead to indentation errors.

Example to illustrate indentation in Python:

```
if True:
    print("This line is indented")
   if False:
       print("This line is indented further")
   print("Back to the first indentation level")
print("This line is not indented")
```

6.Write a program to perform set difference operation

Ans

Run Debug Stop Share Save { } Beautify

main.py

```
A = {10, 20, 30, 40, 80}
B = {100, 30, 80, 40, 60}
print (A.difference(B))
print (B.difference(A))

```

```
{10, 20}
{100, 60}


...Program finished with exit code 0
Press ENTER to exit console.
```

7.Write a Python program to print numbers from 1 to 10 using a while loop.

Ans.

Run Debug Stop Share Save { } Beautify

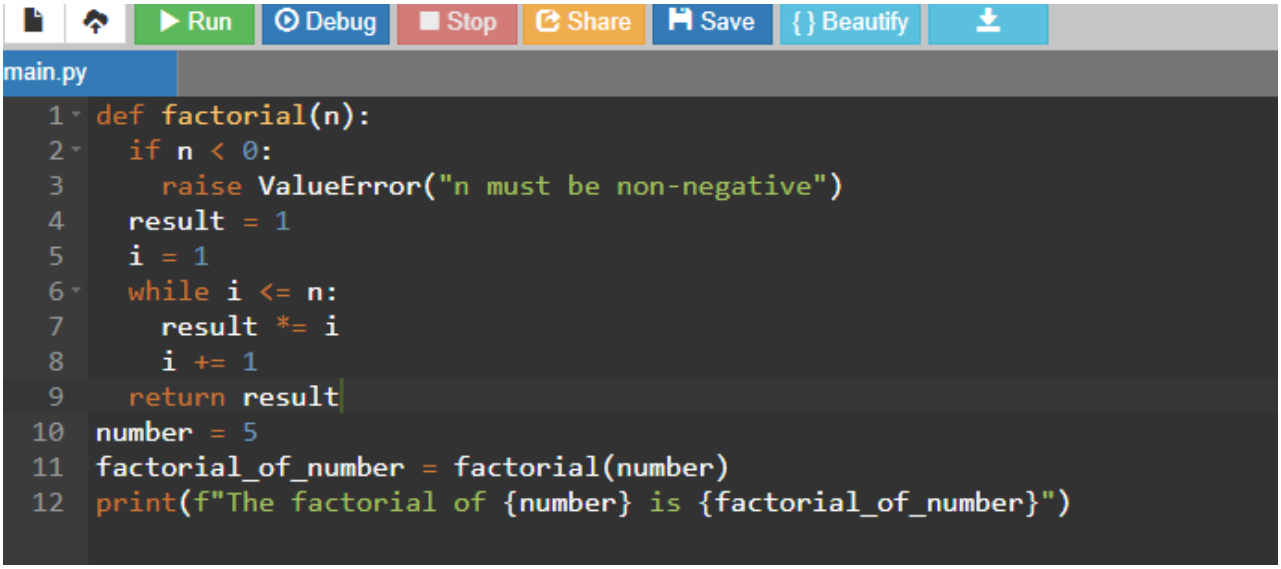
main.py

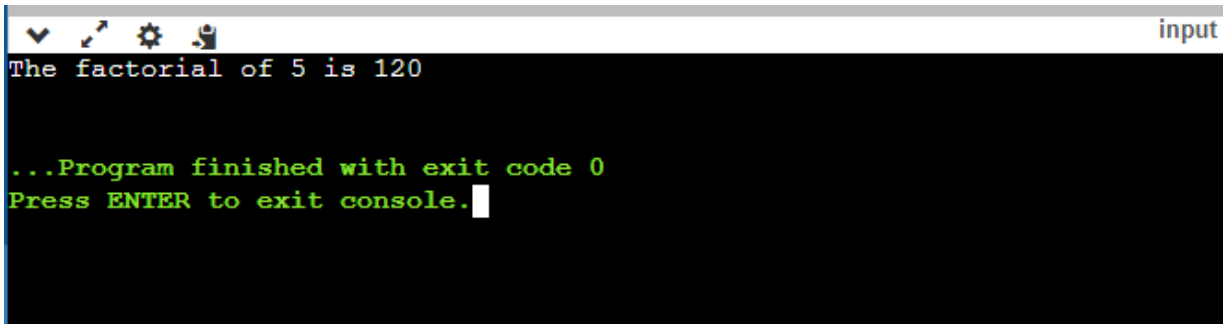
```
print('Numbers from 1 to 10:')
n = 1
while n <= 10:
    print(n, end=' ')
    n = n+1
```

```
Numbers from 1 to 10:
1 2 3 4 5 6 7 8 9 10

...Program finished with exit code 0
Press ENTER to exit console.
```

8.Write a program to calculate the factorial of a number using a while loop.

Ans.





9.Write a Python program to check if a number is positive, negative, or zero using if-elif-else statements.

Ans.

```
Enter a number: 34
Positive number


...Program finished with exit code 0
Press ENTER to exit console.
```

10.Write a program to determine the largest among three numbers using conditional statements.

Ans.

Run | Debug | Stop | Share | Save | { } Beautify

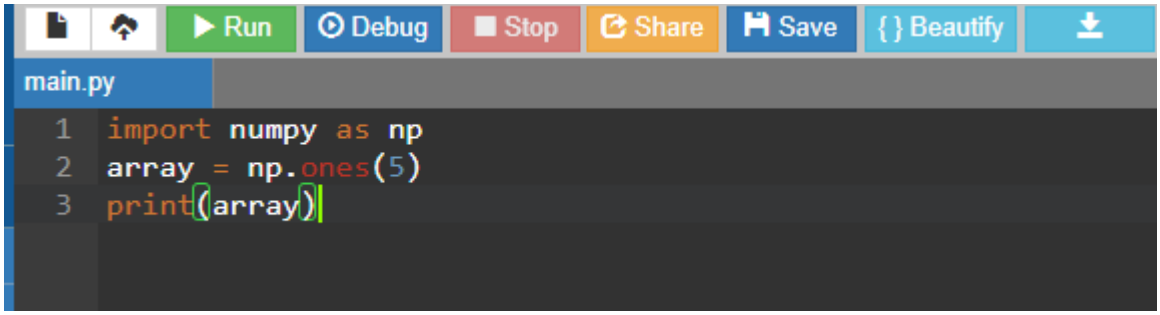
main.py

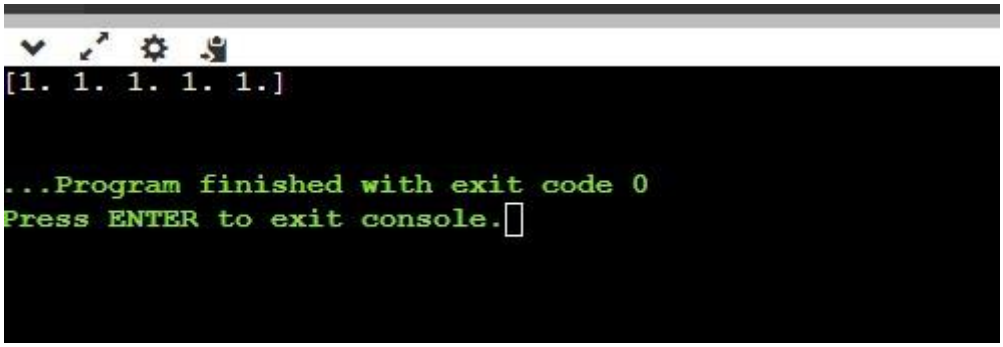
```
num1 = 10
num2 = 14
num3 = 12
if (num1 >= num2) and (num1 >= num3):
   largest = num1
elif (num2 >= num1) and (num2 >= num3):
   largest = num2
else:
   largest = num3
print("The largest number is", largest)
```

```
The largest number is 14


...Program finished with exit code 0
Press ENTER to exit console.
```

11.Write a Python program to create a numpy array filled with ones of given shape.

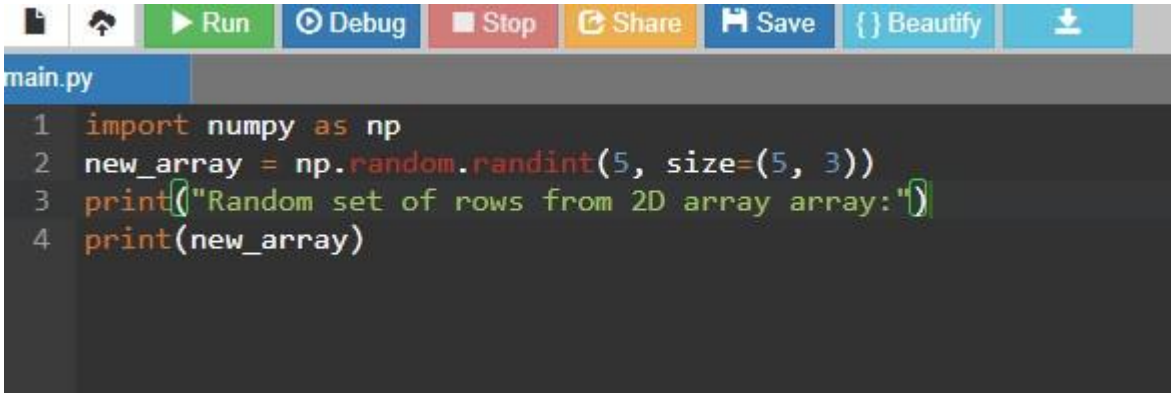
Ans.

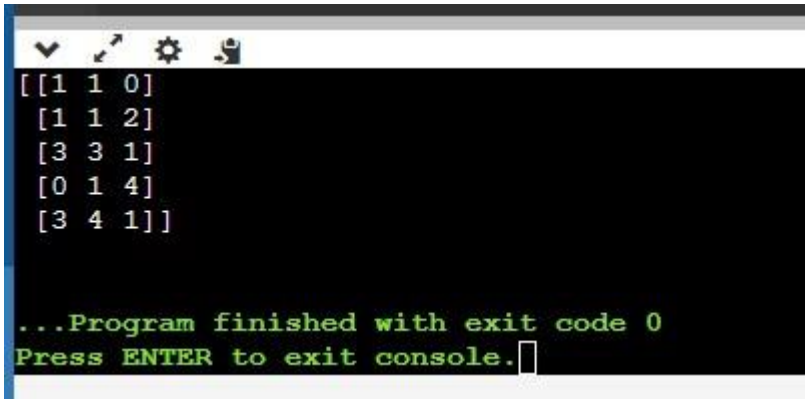




12.Write a program to create a 2D numpy array initialized with random integers.

Ans.





13.write a python program to generate an array of evenly spaced numbers over a specified range using linspace.

Ans.

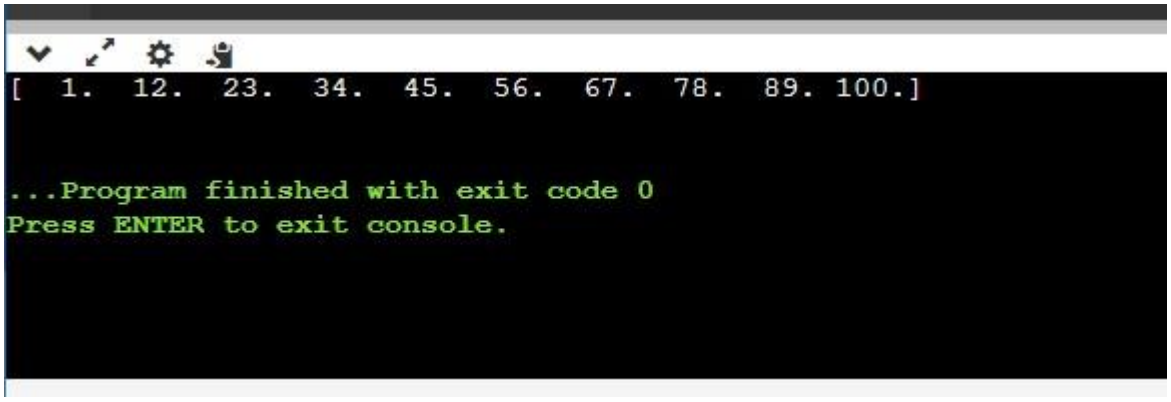
```
main.py
1  import numpy as geek
2  print("B\n", geek.linspace(2.0, 3.0, num=5, retstep=True), "\n")
3  x = geek.linspace(0, 2, 10)
4  print("A\n", geek.sin(x))
5
```

```
B
 (array([2.  , 2.25, 2.5 , 2.75, 3.  ]), 0.25)

A
 [0.         0.22039774 0.42995636 0.6183698  0.77637192 0.8961922
 0.9719379  0.99988386 0.9786557  0.90929743]


...Program finished with exit code 0
```

14.write a program to generate an array of 10 equally spaced values between 1 and 100 using linspace

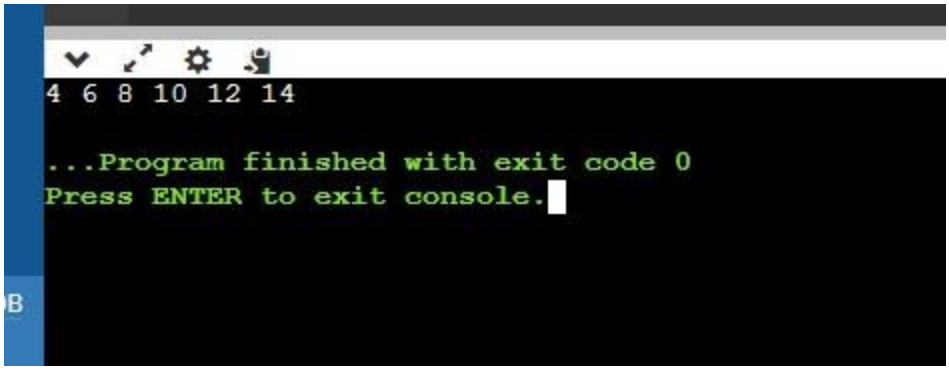
Ans.

```
main.py
1  import numpy as np
2  start = 1
3  end = 100
4  num_values = 10
5  values = np.linspace(start, end, num_values)
6  print(values)
```



15.Write a Python program to create an array containing even numbers from 2 to 20 usinga range.

Ans.



16. Write a program to create an array containing numbers from 1 to 10 with a step size of 0.5 using a range.

Ans.

Run Debug Stop Share Save {} Beautify

main.py

```
1 import numpy as np
2 start = 1
3 end = 10
4 step = 0.5
5 array = np.arange(start, end, step)
6 print(array)
```

input

```
[1.  1.5 2.  2.5 3.  3.5 4.  4.5 5.  5.5 6.  6.5 7.  7.5 8.  8.5 9.  9.5]


...Program finished with exit code 0
Press ENTER to exit console.
```