

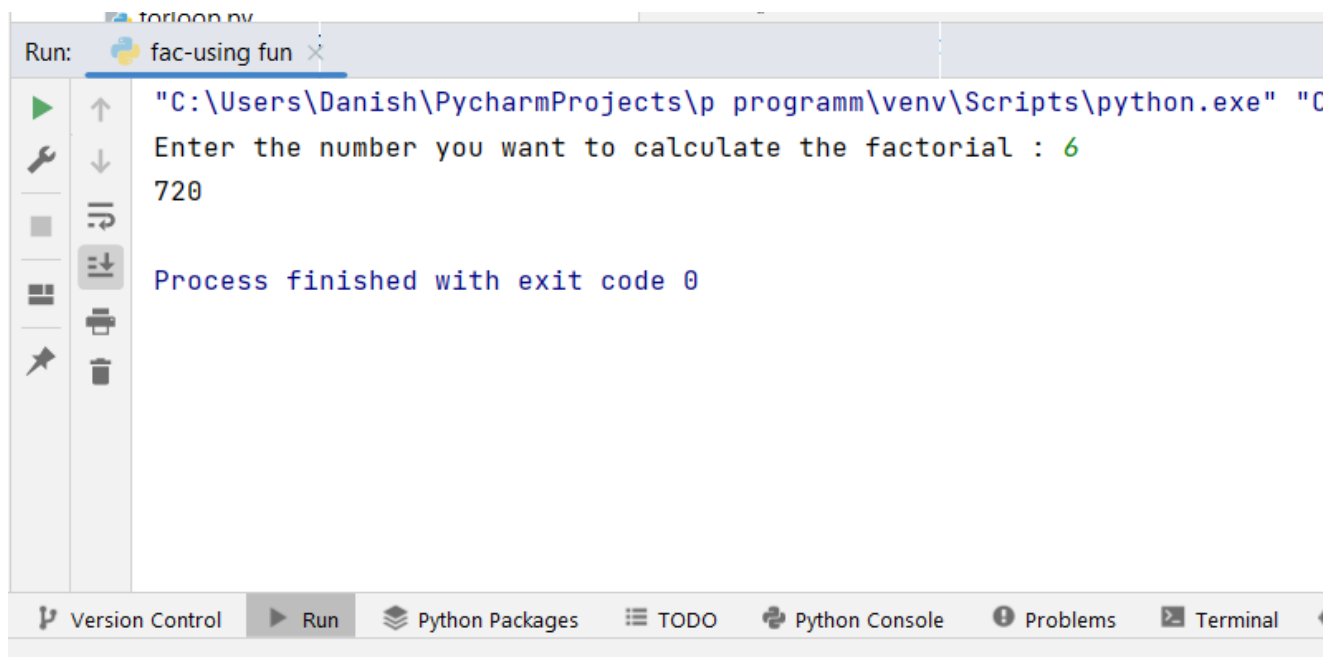
Practical No-13

Title- Program to calculate the factorial of a number entered by the user by using functions

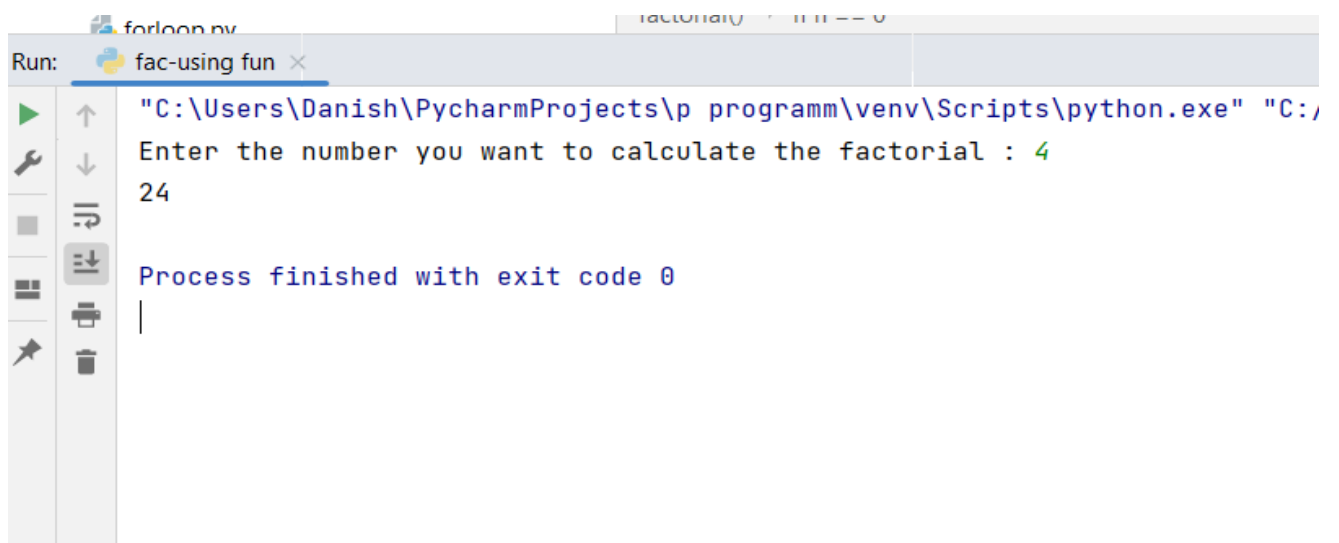
CODE:-

```
def factorial(n):  
    if n == 0:  
        return 1  
    else:  
        return n * factorial(n-1)  
n=int(input("Enter the number you want to calculate the factorial : "))  
print(factorial(n))
```

OUTPUT



The screenshot shows the PyCharm Run console for a file named 'fac-using fun'. The command executed is '"C:\Users\Danish\PycharmProjects\p programm\venv\Scripts\python.exe" "C:\Users\Danish\PycharmProjects\p programm\fac-using fun.py"'. The input provided is '6', and the output is '720'. The console also displays 'Process finished with exit code 0'. The bottom toolbar includes buttons for Version Control, Run, Python Packages, TODO, Python Console, Problems, and Terminal.



This screenshot shows the same PyCharm Run console with the input changed to '4', resulting in an output of '24'. The command and 'Process finished with exit code 0' message are also visible.

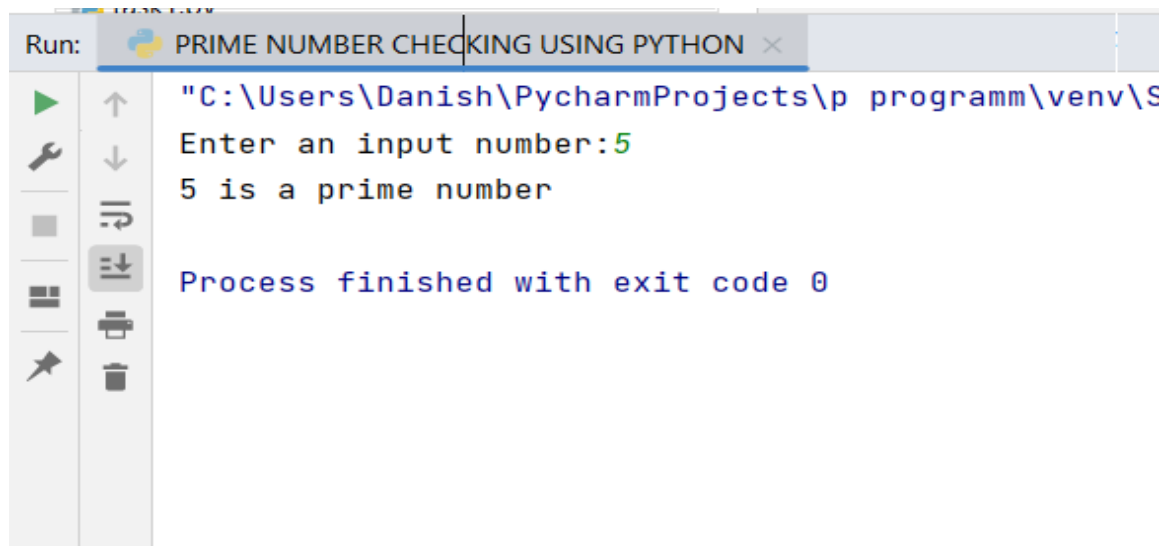
Practical No-14

Title:- Program to check whether entered number by the user is a prime number or not by using functions.

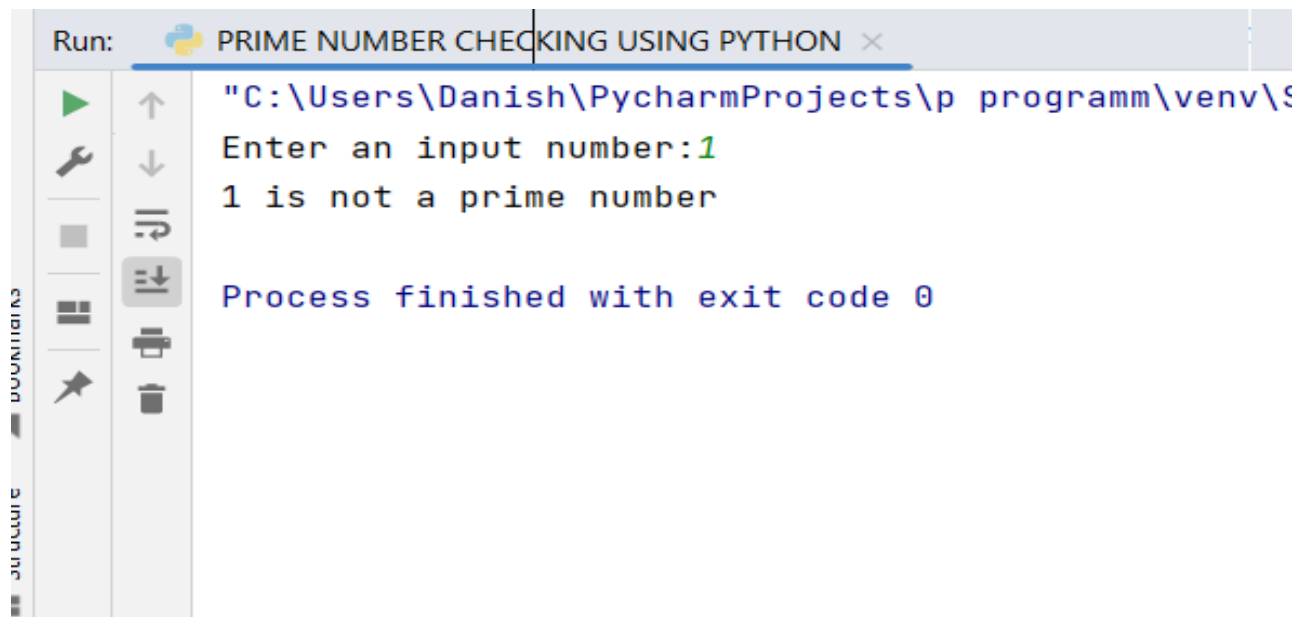
CODE:-

```
def PrimeChecker(a):  
  
    if a > 1:  
  
        for j in range(2, int(a/2) + 1):  
  
            if (a % j) == 0:  
                print(a, "is not a prime number")  
                break  
  
            else:  
                print(a, "is a prime number")  
  
        else:  
            print(a, "is not a prime number")  
  
a = int(input("Enter an input number:"))  
  
PrimeChecker(a)
```

OUTPUT:-



```
Run: PRIME NUMBER CHECKING USING PYTHON x  
"C:\Users\Danish\PycharmProjects\p programm\venv\Scripts  
Enter an input number:5  
5 is a prime number  
  
Process finished with exit code 0
```



The screenshot shows a PyCharm Run window titled "Run: PRIME NUMBER CHECKING USING PYTHON". The window contains a vertical toolbar on the left with icons for running, debugging, and other actions. The main area displays the output of the program in a monospaced font. The output shows the program's path, a prompt for an input number, the user's input '1', and the resulting message '1 is not a prime number'. Finally, it states 'Process finished with exit code 0'.

```
Run: PRIME NUMBER CHECKING USING PYTHON x
"C:\Users\Danish\PycharmProjects\p programm\venv\
Enter an input number:1
1 is not a prime number

Process finished with exit code 0
```

Practical No-15

Title:- Program to print Fabonacci series using functions

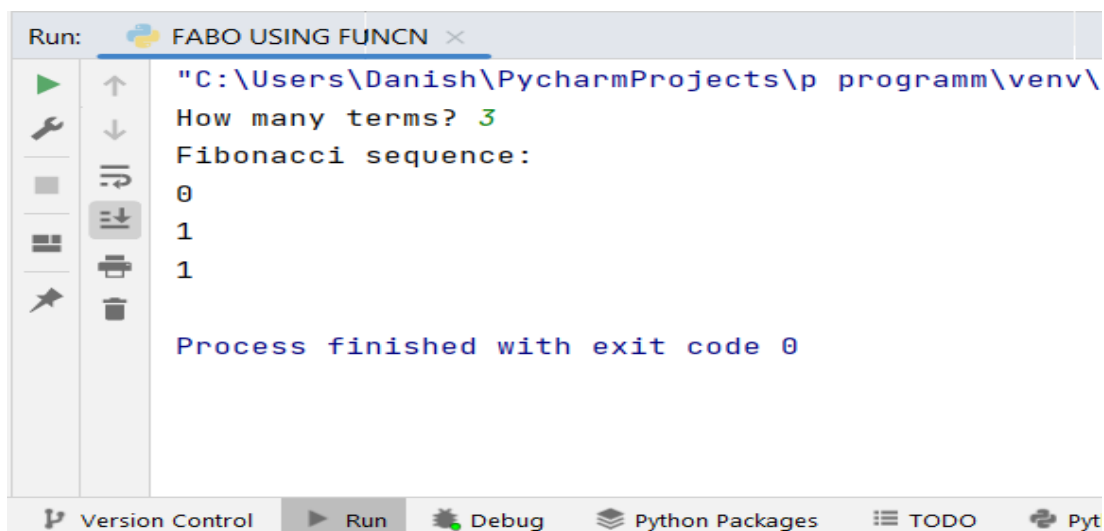
CODE:-

```
def recur_fibo(n):  
    if n <= 1:  
        return n  
    else:  
        return(recur_fibo(n-1) + recur_fibo(n-2))
```

```
nterms = int(input("How many terms? "))
```

```
if nterms <= 0:  
    print("Plese enter a positive integer")  
else:  
    print("Fibonacci sequence:")  
    for i in range(nterms):  
        print(recur_fibo(i))
```

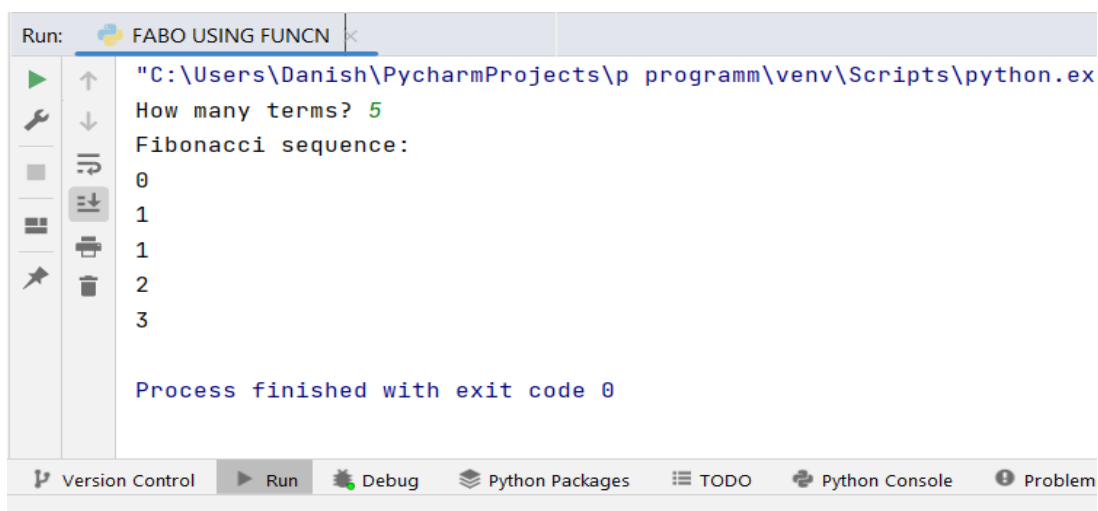
OUTPUT:-



The screenshot shows the PyCharm Run console for a file named 'FABO USING FUNCN'. The output is as follows:

```
"C:\Users\Danish\PycharmProjects\p_programm\venv\  
How many terms? 3  
Fibonacci sequence:  
0  
1  
1  
  
Process finished with exit code 0
```

The bottom toolbar includes buttons for Version Control, Run, Debug, Python Packages, TODO, and Pyt.



The screenshot shows the PyCharm Run console for the same file 'FABO USING FUNCN'. The output is as follows:

```
"C:\Users\Danish\PycharmProjects\p_programm\venv\Scripts\python.ex  
How many terms? 5  
Fibonacci sequence:  
0  
1  
1  
2  
3  
  
Process finished with exit code 0
```

The bottom toolbar includes buttons for Version Control, Run, Debug, Python Packages, TODO, Python Console, and Problem.

Practical No- 16**Title- Strings in Python.**

Aim- To understand all the functions & concepts of strings in Python.

What is String?

String is a data type in Python, composed of a collection of character.

EXAMPLES:

Variable= "String" Variable= "Danish"

Creating strings in Python

String in Python can be created by using single quote, double quote, and even in triple quote.

CODE

```
String1 = 'Danish'  
print('String with the use of Single Quotes: ')  
print(String1)
```

OUTPUT

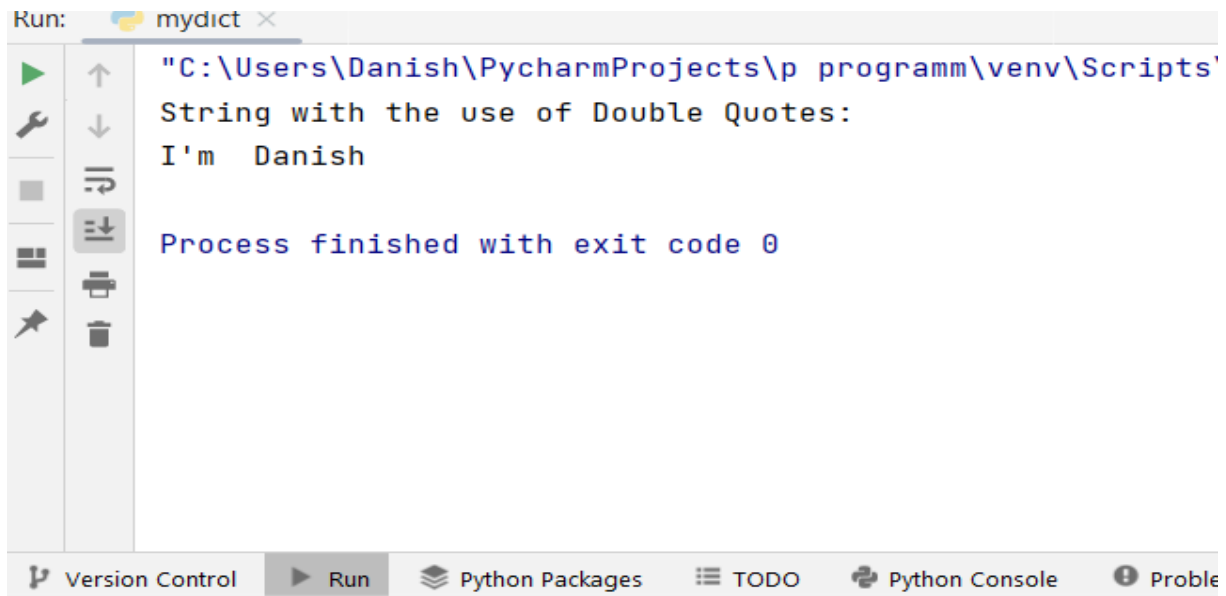
The screenshot shows a terminal window from a Python IDE. The command prompt is "C:\Users\Danish\PycharmProjects\p programm\venv\Scripts\python.exe" "C". The program output is "String with the use of Single Quotes:" followed by "Danish" on the next line. The terminal also shows "Process finished with exit code 0". The IDE interface includes a toolbar on the left with icons for running, debugging, and other actions, and a bottom status bar with tabs for Version Control, Run, Python Packages, TODO, Python Console, Problems, and Terminal.

```
"C:\Users\Danish\PycharmProjects\p programm\venv\Scripts\python.exe" "C  
String with the use of Single Quotes:  
Danish  
  
Process finished with exit code 0
```

CODE

```
String2 = "I'm Danish"  
print("String with the use of Double Quotes: ")  
print(String2)
```

OUTPUT



The screenshot shows the PyCharm Run console for a file named 'mydict'. The output is as follows:

```
"C:\Users\Danish\PycharmProjects\p programm\venv\Scripts'  
String with the use of Double Quotes:  
I'm  Danish  
  
Process finished with exit code 0
```

The bottom toolbar includes buttons for Version Control, Run, Python Packages, TODO, Python Console, and Problems.

CODE

```
String3 = "I'm Danish and im a Hustler"  
print("String with the use of triple Quotes: ")  
print(String3)
```

OUTPUT



The screenshot shows the PyCharm Run console for a file named 'mydict'. The output is as follows:

```
"C:\Users\Danish\PycharmProjects\p programm\venv\Scripts\python.exe  
String with the use of triple Quotes:  
I'm Danish and im a Hustler  
  
Process finished with exit code 0
```

The bottom toolbar includes buttons for Version Control, Run, Python Packages, TODO, Python Console, Problems, and Terminal.

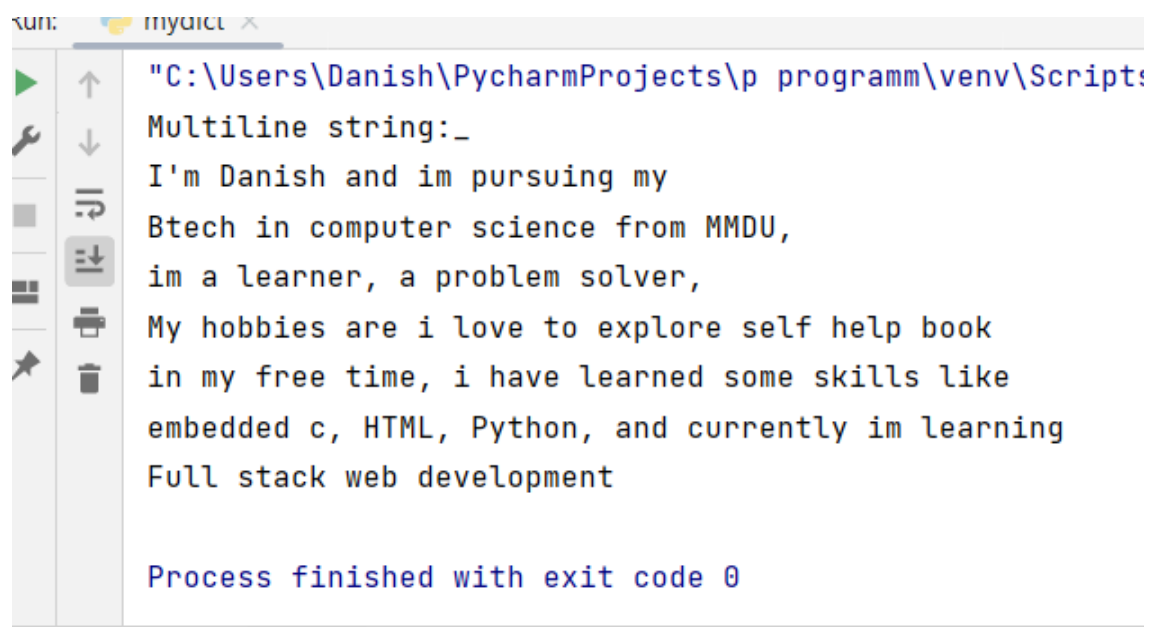
Multiline strings in Python

We can write multiline string in Python by string in triple quotes.

CODE

```
String4 = "I'm Danish and im pursuing my  
Btech in computer science from MMDU,  
im a learner, a problem solver,  
My hobbies are i love to explore self help book  
in my free time, i have learned some skills like  
embedded c, HTML, Python, and currently im learning  
Full stack web development"  
print("Multiline string:_")  
print(String4)
```

OUTPUT

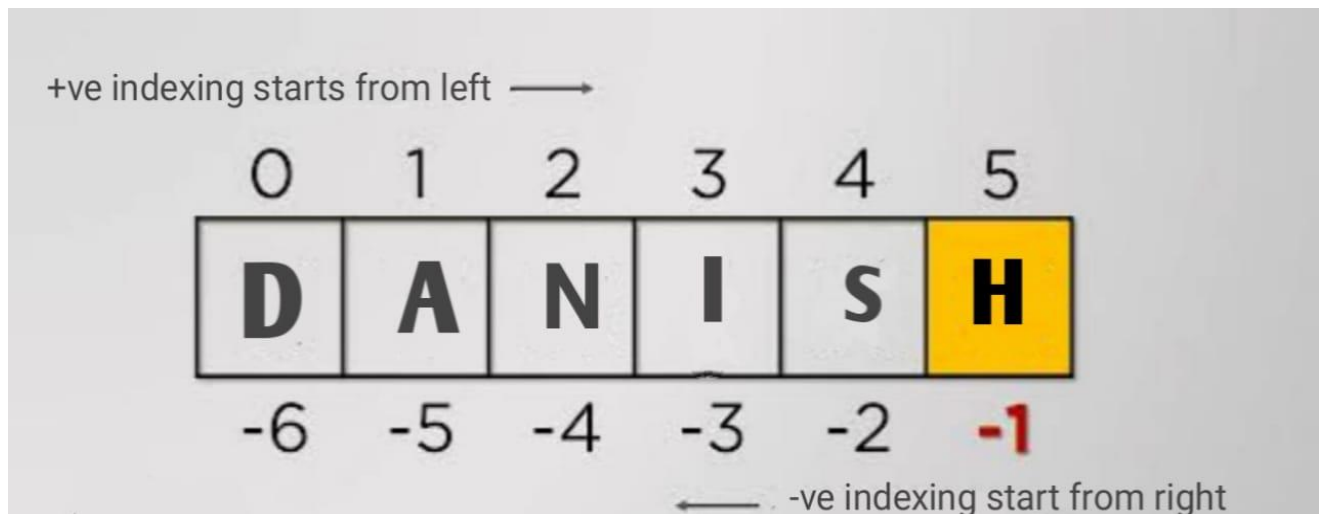


The screenshot shows a terminal window with a light gray background. On the left, there is a vertical toolbar with icons for running, stepping through, and other debugging actions. The main area of the terminal displays the output of a Python script. The first line is the file path: "C:\Users\Danish\PycharmProjects\p programm\venv\Scripts\python.exe". The second line is the prompt "Multiline string:_". The following lines are the content of the multiline string, displayed with line wrapping: "I'm Danish and im pursuing my", "Btech in computer science from MMDU,", "im a learner, a problem solver,", "My hobbies are i love to explore self help book", "in my free time, i have learned some skills like", "embedded c, HTML, Python, and currently im learning", and "Full stack web development". The final line of the output is "Process finished with exit code 0".

```
"C:\Users\Danish\PycharmProjects\p programm\venv\Scripts\python.exe"  
Multiline string:_  
I'm Danish and im pursuing my  
Btech in computer science from MMDU,  
im a learner, a problem solver,  
My hobbies are i love to explore self help book  
in my free time, i have learned some skills like  
embedded c, HTML, Python, and currently im learning  
Full stack web development  
  
Process finished with exit code 0
```

Accessing characters in python String

In Python, individual characters of a string can be accessed by using the method of Indexing.



CODE

```
str="Danish"
```

```
print(str[0])
```

```
print(str[2])
```

```
print(str[-1])
```

OUTPUT

```
"C:\Users\Danish\PycharmProjects\p programm\venv\Script:
D
n
h
Process finished with exit code 0
```


Reversing String in Python

With accessing character from a string, we can also reverse them. We can reverse a string by writing `[::-1]` and the string will be reversed.

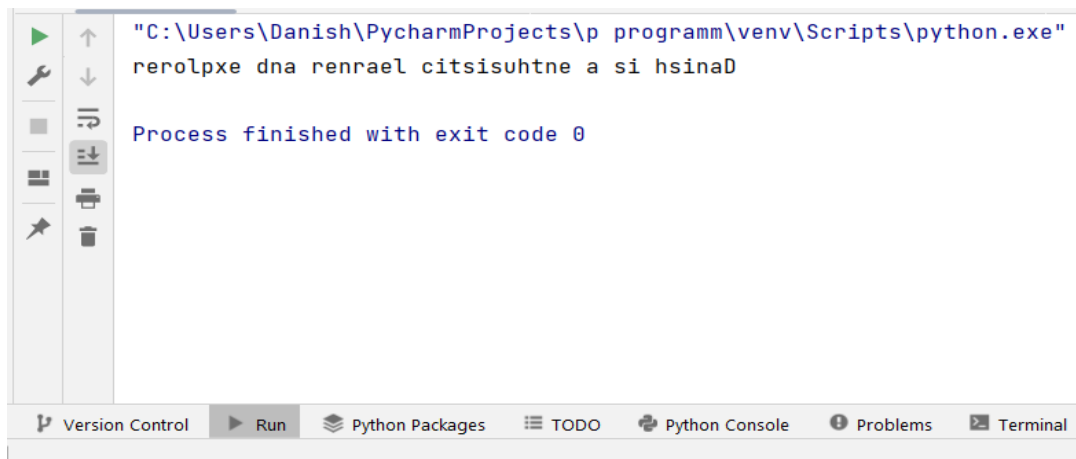
Note- We can also reverse a string by using built in `join` and `reversed` function.

CODE

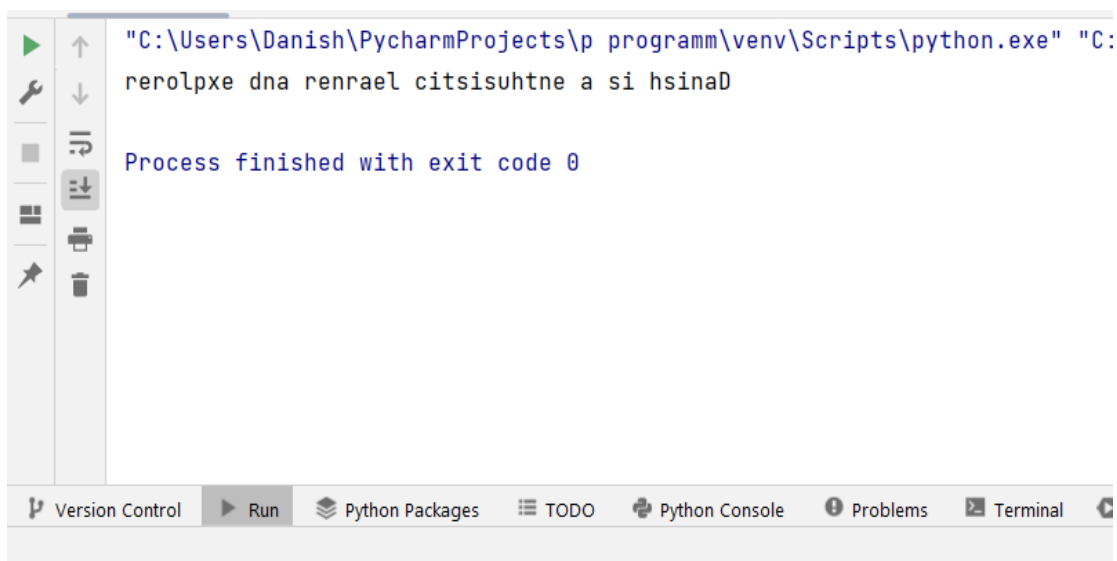
```
str="Danish is a enthusiastic learner and explorer"  
print(str[::-1])
```

```
str = "".join(reversed(str))  
  
print(str)
```

OUTPUT



The image shows a PyCharm Run console window. The top line displays the command prompt path: `"C:\Users\Danish\PycharmProjects\p programm\venv\Scripts\python.exe"`. The next line shows the output of the first print statement: `rerolpxe dna renrael citsisuhtne a si hsinaD`. The final line indicates the process finished with exit code 0. The bottom toolbar includes buttons for Version Control, Run, Python Packages, TODO, Python Console, Problems, and Terminal.



The image shows a PyCharm Run console window. The top line displays the command prompt path: `"C:\Users\Danish\PycharmProjects\p programm\venv\Scripts\python.exe"`. The next line shows the output of the first print statement: `rerolpxe dna renrael citsisuhtne a si hsinaD`. The final line indicates the process finished with exit code 0. The bottom toolbar includes buttons for Version Control, Run, Python Packages, TODO, Python Console, Problems, and Terminal.

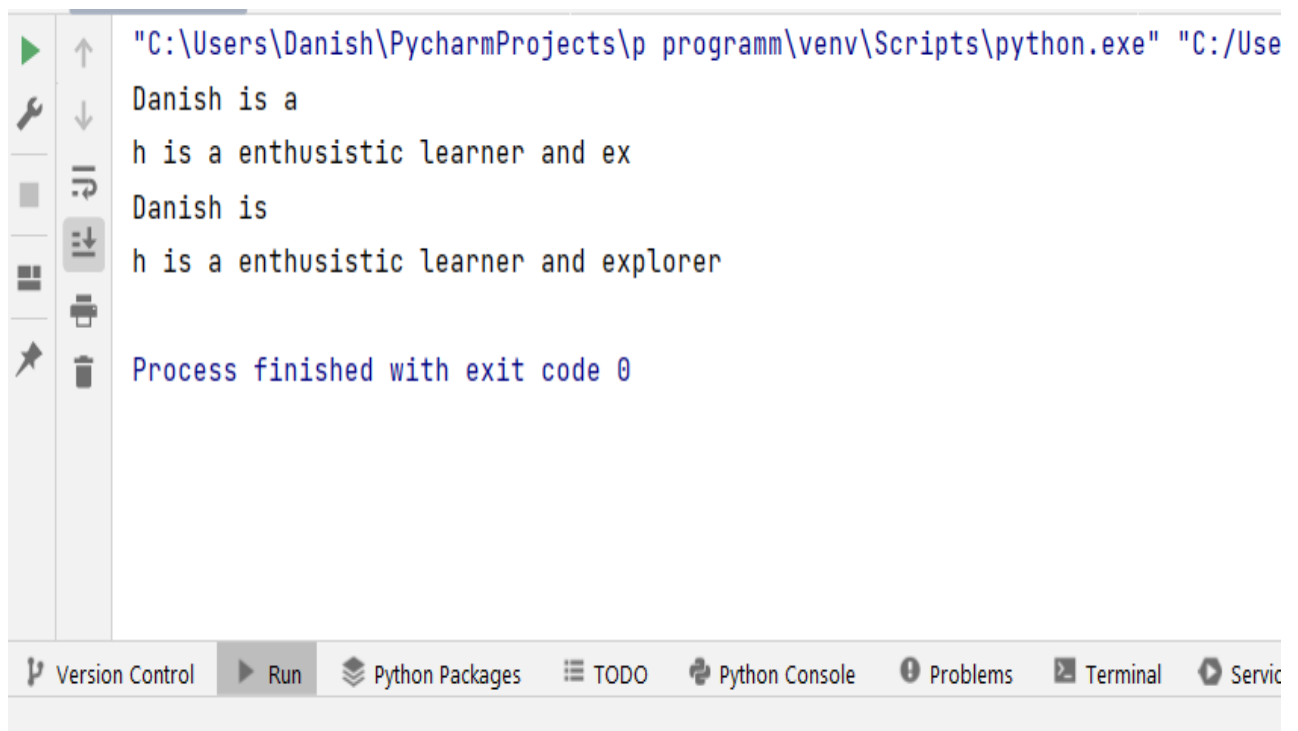
Slicing of string in Python

To access a range of characters in the string, the method of slicing is used.

CODE

```
str = "Danish is a enthusiastic learner and explorer"  
print(str[0:12])  
  
print(str[5:-6])  
  
print(str[:9])  
  
print(str[5:])
```

OUTPUT



```
"C:\Users\Danish\PycharmProjects\p programm\venv\Scripts\python.exe" "C:/Use  
Danish is a  
h is a enthusiastic learner and ex  
Danish is  
h is a enthusiastic learner and explorer  
Process finished with exit code 0
```

Functions of string in Python

1) The below functions are used to change the case of the strings.

Lower(): Convert all uppercase character in a string into lowercase

Upper(): Convert all lowercase character in a string into uppercase

Title(): Convert string to title case

CODE

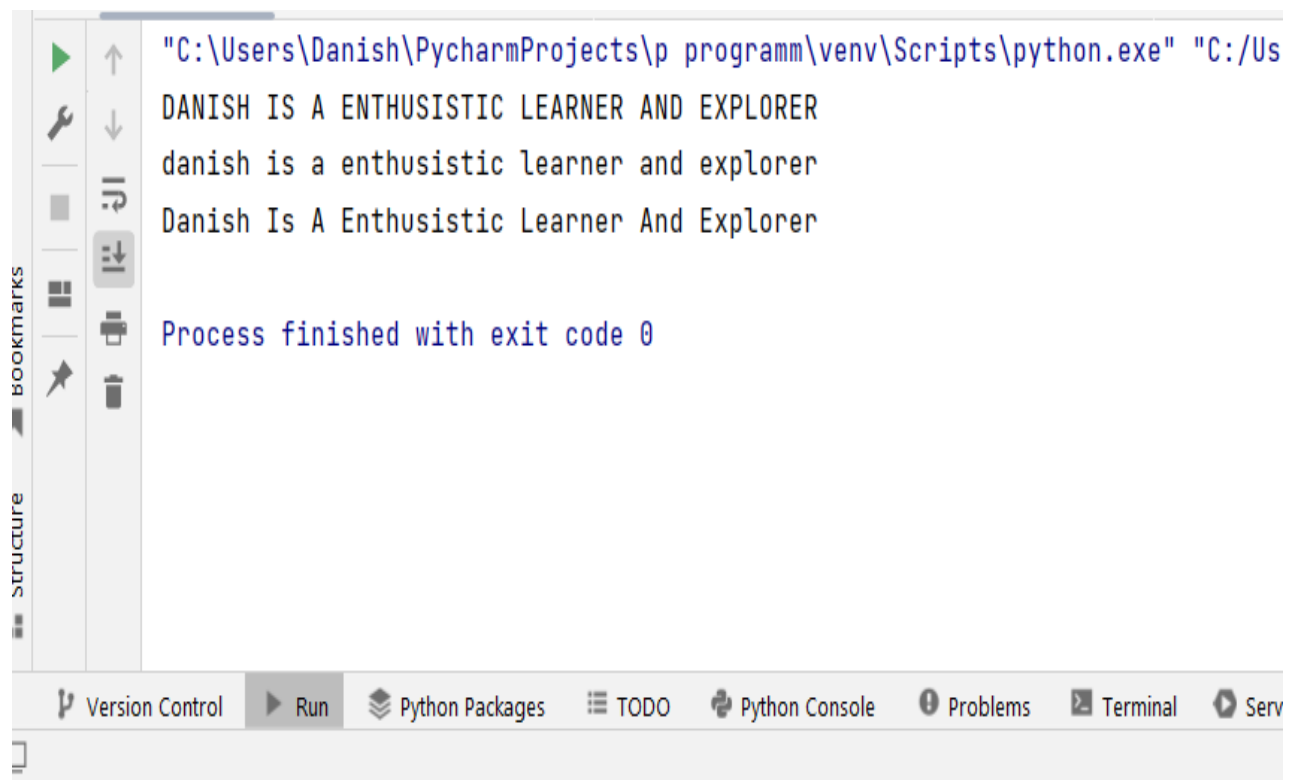
```
str = "Danish is a enthusiastic learner and explorer"
```

```
print(str.upper())
```

```
print(str.lower())
```

```
print(str.title())
```

OUTPUT



The screenshot shows a PyCharm terminal window with the following output:

```
"C:\Users\Danish\PycharmProjects\p programm\venv\Scripts\python.exe" "C:/Us  
DANISH IS A ENTHUSISTIC LEARNER AND EXPLORER  
danish is a enthusiastic learner and explorer  
Danish Is A Enthusistic Learner And Explorer  
  
Process finished with exit code 0
```

The terminal window has a sidebar on the left with icons for 'Structure', 'Bookmarks', and 'Run'. The bottom status bar shows 'Version Control', 'Run', 'Python Packages', 'TODO', 'Python Console', 'Problems', 'Terminal', and 'Serv'.

2) Len(): Use to know the length of string

CODE

```
str1 = "Danish is a enthusiastic learner and explorer"

str2="Python is a high level interpreted language,
      which is best suited for writing python scripts
      for automation and code re-usability.
      It was created in 1991 by Guido Van Rossum.
      The origin of its name is inspired by the comedy series called 'Monty   python'"

print(len(str1))

print(len(str2))
```

OUTPUT



3) find(): it is use to find the first occurance of the specified value.

It will return -1 if the value is not found.

CODE

```
str1 = "Danish is a enthusistic learner and explorer"  
x = str1.find("D")  
y = str1.find("z")  
  
print(x)  
print(y)
```

OUTPUT

A screenshot of a Python IDE's console window. The console shows the execution of a Python script. The first line of output is the file path: "C:\Users\Danish\PycharmProjects\p programm\venv\Scripts\python". The next two lines are the results of the find() function: 0 and -1. The final line of output is "Process finished with exit code 0". The IDE interface includes a toolbar on the left with icons for running, debugging, and other actions. At the bottom, there is a status bar with tabs for Version Control, Run, Python Packages, TODO, Python Console, and Problems. A message "PEP 8: W391 blank line at end of file" is visible in the status bar.

4) isalnum(): it is used to check wether all the character in a given string are either alphabet or numeric (alphanumeric) characters

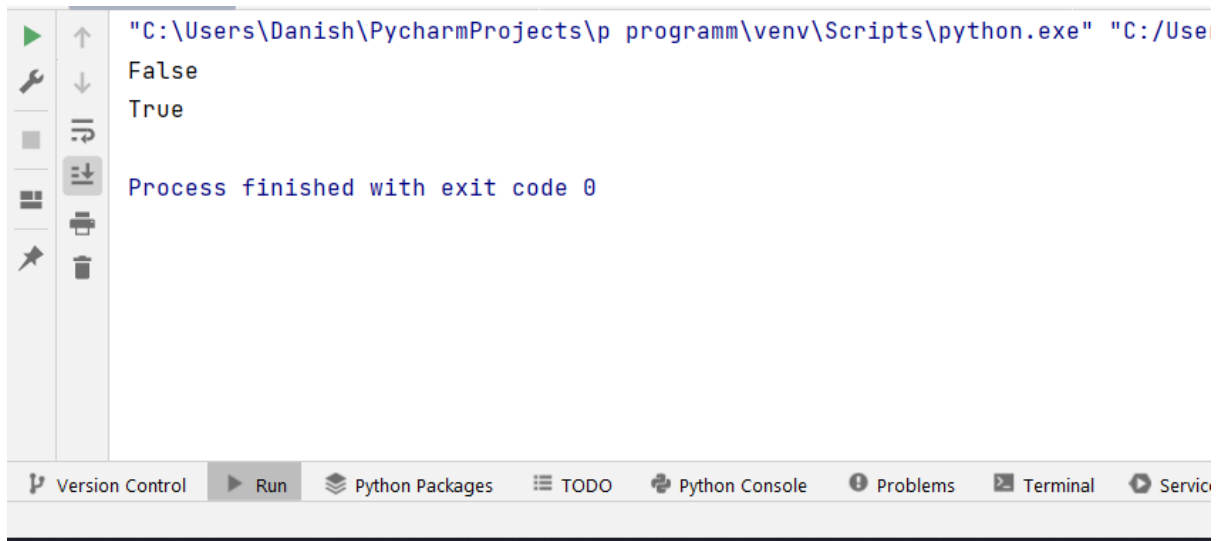
It will return **TRUE** if all characters are alphanumeric

It will return **FALSE** if all characters are not alphanumeric

CODE

```
str1 = "Danish is a enthusistic learner and explorer"  
str2 = "abc145"  
print(str1.isalnum())  
print(str2.isalnum())
```

OUTPUT

A screenshot of the PyCharm Run console. The console shows the execution of a Python script. The output is as follows:

```
"C:\Users\Danish\PycharmProjects\p programm\venv\Scripts\python.exe" "C:/Use  
False  
True  
  
Process finished with exit code 0
```

The interface includes a toolbar on the left with icons for running, debugging, and other actions. At the bottom, there is a tab bar with options: Version Control, Run, Python Packages, TODO, Python Console, Problems, Terminal, and Services.

5) split(): It is used to split string into the list

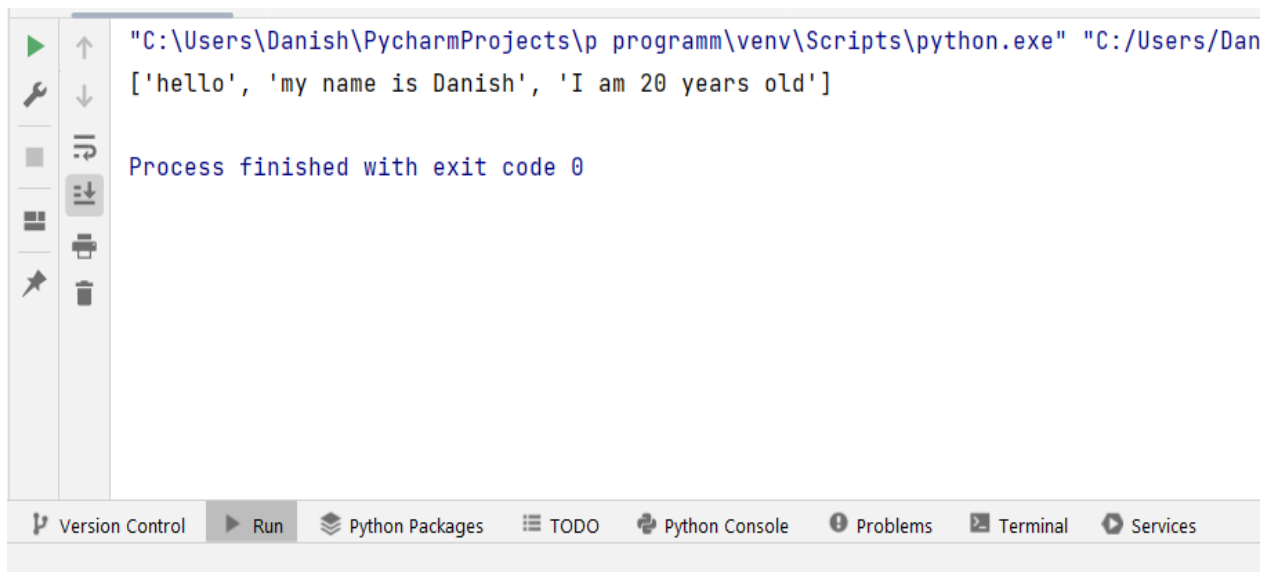
CODE

```
txt = "hello, my name is Peter, I am 26 years old"
```

```
x = txt.split(", ")
```

```
print(x)
```

OUTPUT

A screenshot of the PyCharm Run console showing the output of the split() function. The output is as follows:

```
"C:\Users\Danish\PycharmProjects\p programm\venv\Scripts\python.exe" "C:/Users/Dan  
['hello', 'my name is Danish', 'I am 20 years old']  
  
Process finished with exit code 0
```

The interface is identical to the previous screenshot, showing the same toolbar and bottom tab bar.

6) replace(): The replace method replace a specified phrase with another phrase

CODE

```
string = "Hoy it's me Danish"  
new_string = string.replace("o", "e" )
```

```
print(string)  
print(new_string)
```

OUTPUT



7) isdigit(): The isdigit method returns **True** if all the character are digit, otherwise **false**.

CODE

```
a = "u0030"  
b = "u00B2"
```

```
print(a.isdigit())  
print(b.isdigit())
```

OUTPUT

