DS_2303	Python	Worksheet 1

1.	C) %
2.	B) 0
3.	C) 24
4.	A) 2
5.	D) 6
6.	C) the finally block will be executed no matter if the try block raises an error or not
7.	A) It is used to raise an exception.
8.	A) in defining an iterator
9.	A) _abc and C) abc2
10.	A) yield and B) raise
11.	In [3]: ``` # Factorial number import math def factorial(n): return(math.factorial(n)) # Code num=9 print("factorial of", num, "is",factorial(num)) ``` ``` factorial of 9 is 362880 ```
12.	In [25]: ``` #Prime or Composite n=1 if n>1: for i in range(2,int(n/2)+1): if(n%i)==0: print(num,"is not a prime number") break else: print(n,"is a prime number") # if the number is less than 1, it is nota prime number. else: print(n,"is not a prime number") n=5 if n>1: for i in range(2,int(n/2)+1): if(n%i)==0: print(num,"is not a prime number") break else: print(n,"is a prime number") # if the number is less than 1, it is nota prime number. else: print(n,"is not a prime number") ```

```
n=9

if n>1:
    for i in range(2,int(n/2)+1):
        if(n%i)==0:
            print(num,"is not a prime number")
            break
    else:
        print(n,"is a prime number")
# if the number is less than 1, it is nota prime number.
else:
    print(n,"is not a prime number")
```

```
1 is not a prime number
5 is a prime number
9 is not a prime number
```

13.

```
In [38]: # Given String is palindrome or not
string=("SARAS")
if(string==string[::-1]):
    print("The String is Palindrome")
else:
     print("The String is not Palindrome")

string=("VISA")
if(string==string[::-1]):
    print("The String is Palindrome")
else:
     print("The String is not Palindrome")
```

```
The String is Palindrome
The String is not Palindrome
```

14.

In [98]:
```
def test(opposite_side,adjacent_side,hypotenuse):
    if opposite_side==str("x"):
        return(opposite==str(((hypotenuse**2)-(adjacent_side**2))**0.5))
    elif adjacent_side==str("x"):
        return(adjacent==str(((hypotenuse**2)-(opposite_side**2))**0.5))
    elif hypotenuse==str("x"):
        return(hypotenuse==str(((opposite_side**2)+(adjacent_side**2))**0.5))
    else:
        return "Already know the answer"

print(test(6,8,10))
```

```
Already know the answer
```

15.

In [79]:
```
#Frequency of string
test_string = "Vishakha"
all_freq = {}
for i in test_string:
    if i in all_freq:
        all_freq[i] += 1
    else:
         all_freq[i] = 1
print("count of all charcter in Vishakha is :\n "+str(all_freq))
```

```
count of all charcter in Vishakha is :
 {'V': 1, 'i': 1, 's': 1, 'h': 2, 'a': 2, 'k': 1}
```