In [1]:
```
6<<2
```

Out[1]:
```
24
```

In [2]:
```
2//3
```

Out[2]:
```
0
```

In [3]:
```
6&2
```

Out[3]:
```
2
```

In [13]:
```
num = 6
factorial = 1
if num < 0:
    print ("Factorial does not exist for negative number")
elif num ==0:
    print ("The factorial of is 1")
else:
    for i in range (1,num+1):
        factorial = factorial*i
    print ("The factorial of", num, "is",factorial)
```

```
The factorial of 6 is 720
```

In [22]:
```
num = int(input("enter Number:"))
if num > 1:
    for n in range(2,num):
        print (n)
        if num % n == 0:
            print ('not prime')
            break
    else:
        print('prime')

elif num == 0 or 1:
    print (num,"number is not prime nor composite number")
else:
    print(num,"it is not prime its composite number")
```

```
enter Number:0
0 number is not prime nor composite number
```

In [33]:
```
def is_palindrome (p):
    return p == p[::-1]

p = "121"
string = is_palindrome (p)

if string:
    print("yes,palindrome")
else:
    print("no,not palindrome")
```

```
yes,palindrome
```

In [40]:
```
def right_angle (opposite_side,adjacent_side,hypotenuse):
    if opposite_side == str("r"):
        return ("opposite = " + str (((hypotenuse**2) - (adjacent_side**2))*0.5))
    elif adjacent_side == str("r"):
        return ("adjacent =" + str(((hypotenuse**2) - (opposite_side**2))*0.5))
    elif hypotenuse == str("r"):
         return (" hypotenuse =" + str(((opposite_side**2) - (adjacent_side**2))*0.5))
    else:
        return "its done"
```

In [48]:

```
print (right_angle(4,5,"r"))
print (right_angle(4,"r",6))
print (right_angle("r",5,6))
```

```
 hypotenuse =-4.5
adjacent =10.0
opposite = 5.5
```

In [57]:

```
input_string = "Worksheet"
freq_of_string = {}
for x in input_string:
    if x in freq_of_string:
        freq_of_string[x] += 1
    else:
        freq_of_string[x] = 1
print ("characters present in given string:",input_string,str(freq_of_string))
```

```
characters present in given string: Worksheet {'W': 1, 'o': 1, 'r': 1, 'k': 1, 's': 1, 'h': 1, 'e': 2, 't': 1}
```

In [ ]:

In [ ]: