In [1]:
```
#Function to claculate alternate numbers

def alter(start, end):
    for value in range(start, end+1,2):
        print (value, end=" ")
    return
alter(12,22)
```

```
12 14 16 18 20 22
```

In [2]:
```
# Function to print reverse numbers

def reverserange(start, end):
    for value in range(end, start-1,-1):
        print (value, end=" ")
    return
reverserange(12, 20)
```

```
20 19 18 17 16 15 14 13 12
```

In [3]:
```
#Claculate reverve ODD numbers

def oddReverse(start, end):
    for value in range(end, start-1, -1):
        if value % 2 != 0:
            print(value, end="  ")
    return
oddReverse(123, 130)
```

```
129  127  125  123
```

In [4]:
```
# Calculate the numbers from given range with given gap

for i in range(0,50,5):
    print(i, end="  ")
```

```
0  5  10  15  20  25  30  35  40  45
```

In [5]:
```
#Calculate the no of hours per year

def noDays(startyear, endyear):
    sum = 0
    for year in range(startyear, endyear):
        if(year%4==0 and year%100!=0 or year%400==0):
            sum=sum+366
        else:
            sum=sum+365
    return sum*24
noDays(2016, 2017)
```

Out[5]:
```
8784
```

In [6]:
```
#Squaring the numbers from given range

for i in range (10):
          print(4**i, end=" ")
```

```
1 4 16 64 256 1024 4096 16384 65536 262144
```

In [7]:
```
#Calculate the Leap year from given Year

def isLeapyear(year):
    if(year%4==0 and year%100!=0 or year%400==0):
        print(year, "is a Leap Year")
    else:
        print(year, "is Not a Leap Year")
isLeapyear(1905)
```

```
1905 is Not a Leap Year
```

In [8]:
```
#Calculate the Leap year from given Year

def isLeapyear(year):
    if(year%4==0 and year%100!=0 or year%400==0):
        print(year, "is a Leap Year")
    else:
        print(year, "is Not a Leap Year")
isLeapyear(1905)
```

```
1905 is Not a Leap Year
```

In [18]:
```
#Calculate the No of Days from Given range years

from datetime import date
f_date=date(1996,1,16)
l_date=date(2019,6,13)
delta_days=-(f_date-l_date)
print(delta_days)
```

```
8549 days, 0:00:00
```

In [ ]:
```

```