1. Write a function that inputs a number and prints the multiplication table of that number

In [1]:

```
def DisplayTable(num):
    print("\nMultiplication table of", num)
    for i in range(1, 11):
        x = i * int(num)
        print(num, "*", i, "= ", x)

a = input("Enter the number ")
DisplayTable(a)
```

```
Enter the number 5

Multiplication table of 5
5 * 1 =  5
5 * 2 =  10
5 * 3 =  15
5 * 4 =  20
5 * 5 =  25
5 * 6 =  30
5 * 7 =  35
5 * 8 =  40
5 * 9 =  45
5 * 10 =  50
```

2. Write a program to print twin primes less than 1000. If two consecutive odd numbers are both prime then they are known as twin primes

In [20]:

```
def printTwinPrime(n):

    # Create a boolean array "prime[0..n]"
    # and initialize all entries it as true. A value in prime[i] will
    # finally be false if i is Not a prime, else true.
    prime = [True for i in range(n + 2)]
    p = 2

    while (p * p <= n + 1):

        # If prime[p] is not changed,
        # then it is a prime
        if (prime[p] == True):

            # Update all multiples of p
            for i in range(p * 2, n + 2, p):
                prime[i] = False
        p += 1

    # check twin prime numbers
    # display the twin prime numbers
    for p in range(2, n-1):
        if prime[p] and prime[p + 2]:
            print("(",p,",", (p + 2), ")")


# Calling the function
printTwinPrime(1000)
```

```
( 3 , 5 )
( 5 , 7 )
( 11 , 13 )
( 17 , 19 )
( 29 , 31 )
( 41 , 43 )
( 59 , 61 )
( 71 , 73 )
( 101 , 103 )
( 107 , 109 )
( 137 , 139 )
( 149 , 151 )
( 179 , 181 )
( 191 , 193 )
( 197 , 199 )
( 227 , 229 )
( 239 , 241 )
( 269 , 271 )
( 281 , 283 )
( 311 , 313 )
( 347 , 349 )
( 419 , 421 )
( 431 , 433 )
( 461 , 463 )
( 521 , 523 )
( 569 , 571 )
( 599 , 601 )
( 617 , 619 )
( 641 , 643 )
( 659 , 661 )
( 809 , 811 )
( 821 , 823 )
( 827 , 829 )
( 857 , 859 )
( 881 , 883 )
```

3. Write a program to find out the prime factors of a number.

Example: prime factors of 56 - 2, 2, 2, 7

In [28]:

```
n=int(input("Enter an integer:"))
print("Factors are:")
i=1
while(i<=n):
    k=0
    if(n%i==0):
        j=1
        while(j<=i):
            if(i%j==0):
                k=k+1
            j=j+1
        if(k==2):
            print(i)
    i=i+1
```

```
Enter an integer:56
Factors are:
2
7
```

5. Write a function that converts a decimal number to binary number

In [6]:

```
# Function to print binary number for the input decimal using recursion
def decimalToBinary(n):
    if(n > 1):
        # (discard remainder)
        print("N:", n)
        decimalToBinary(n//2)

    print(n%2, end=' ')

n = int(input("Enter the decimal number"))
decimalToBinary(n)
```

```
Enter the decimal number9
N: 9
N: 4
N: 2
1 0 0 1
```

7. Write a function prodDigits() that inputs a number and returns the product of digits of that number.

In [7]:

```
def prodDigits(num):
    product = 1
    try:
        for i in num:
            print("i:",i)
            product = product * int(i)
        print(product)
    except:
        print("Exception")

num = input("Enter the number ")
prodDigits(num)
```

```
Enter the number 45
i: 4
i: 5
20
```

9. Write a function sumPdivisors() that finds the sum of proper divisors of a number. Proper divisors of a number are those numbers by which the number is divisible, except the number itself. For example proper divisors of 36 are 1, 2, 3, 4, 6, 9, 18

In [26]:

```
def sumPdivisors(num):
        sum=0
        for i in range(1,num):
            if(num%i == 0):
                print(i, end=' ')
                sum = sum+i
        print("\nSum:",sum)
        if(sum == num):
            print("Perfect Number")

num = int(input("Enter the number "))
sumPdivisors(num)
```

```
Enter the number 28
1 2 4 7 14
Sum: 28
Perfect Number
```

10. A number is called perfect if the sum of proper divisors of that number is equal to the number. For example 28 is perfect number, since 1+2+4+7+14=28. Write a program to print all the perfect numbers in a given range

In [51]:

```
def perfectNumber(num):
    sum=0
    for i in range(1,num):
        if(num%i == 0):
            sum += i
    return (True if sum == num else False)

num = int(input("Enter the number "))
for n in range (1,num):
    if perfectNumber(n):
        print(n , " is a perfect number")
```

```
Enter the number 1000
6  is a perfect number
28  is a perfect number
496  is a perfect number
```

12. Write a program which can filter odd numbers in a list by using filter function

In [9]:

```
# a list contains both even and odd numbers.
seq = [0, 1, 2, 3, 5, 8, 13,15]

# result contains odd numbers of the list
result = list(filter(lambda x: x % 2 != 0, seq))
print(result)

# lstofNum = range(-10,15)

# def OddNumber(x):
#     if x%2 != 0:
#         return True

# adults = list(filter(OddNumber, lstofNum))
# print(adults)
```

```
[1, 3, 5, 13, 15]
```

13. Write a program which can map() to make a list whose elements are cube of elements in a given list

In [87]:

```
def Cube(num):
    return num ** 3

data = [1,2,4,5,6,7]
#result = list(map(Cube,data))
result = list(map(lambda x: x **3,data))
print(result)
```

```
[1, 8, 64, 125, 216, 343]
```

14. Write a program which can map() and filter() to make a list whose elements are cube of even number in a given list

In [10]:

```
data = [1,2,4,5,6]
result = list(filter(lambda x: x is not None,map(lambda x: x**3 if x%2==0 else Non
print(result)
```

```
[8, 64, 216]
```