**Python Program to Add two Matrices**

we will see **how to add two matrices in Python**. Before we see how to implement matrix addition in Python, lets see what it looks like:

M1 = [[1,1,1],

[1,1,1],

[1,1,1]]

M2 = [[1,2,3],

[4,5,6],

[7,8,9]]

Sum of these matrices:

= [[2,3,4],

[5,6,7],

[8,9,10]]

**Program for adding two matrices**

To represent a matrix, we are using the concept of [nested lists](https://beginnersbook.com/2018/02/python-list/). All the elements of both the input matrices are represented as nested lists. All the elements of output list are initialized as zero.

We are iterating the matrix and adding the corresponding elements of both the given matrices and assigning the value in the output matrix.

# This program is to add two given matrices

# We are using the concept of nested lists to represent matrix

# first matrix

M1 = [[1, 1, 1],

[1, 1, 1],

[1, 1, 1]]

# second matrix

M2 = [[1, 2, 3],

[4, 5, 6],

[7, 8, 9]]

# In this matrix we will store the sum of above matrices

# we have initialized all the elements of this matrix as zero

sum = [[0, 0, 0],

[0, 0, 0],

[0, 0, 0]]

# iterating the matrix

# rows: number of nested lists in the main list

# columns: number of elements in the nested lists

for i in range(len(M1)):

for j in range(len(M1[0])):

sum[i][j] = M1[i][j] + M2[i][j]

# displaying the output matrix

for num in sum:

print(num)

Output:

[2, 3, 4]

[5, 6, 7]

[8, 9, 10]

**1. Decimal to Binary conversion using recursive function**

In this program we have defined a function decimalToBinary() for the conversion. This function takes the decimal number as an input parameter and converts it into an equivalent binary number.

def decimalToBinary(num):

"""This function converts decimal number

to binary and prints it"""

if num > 1:

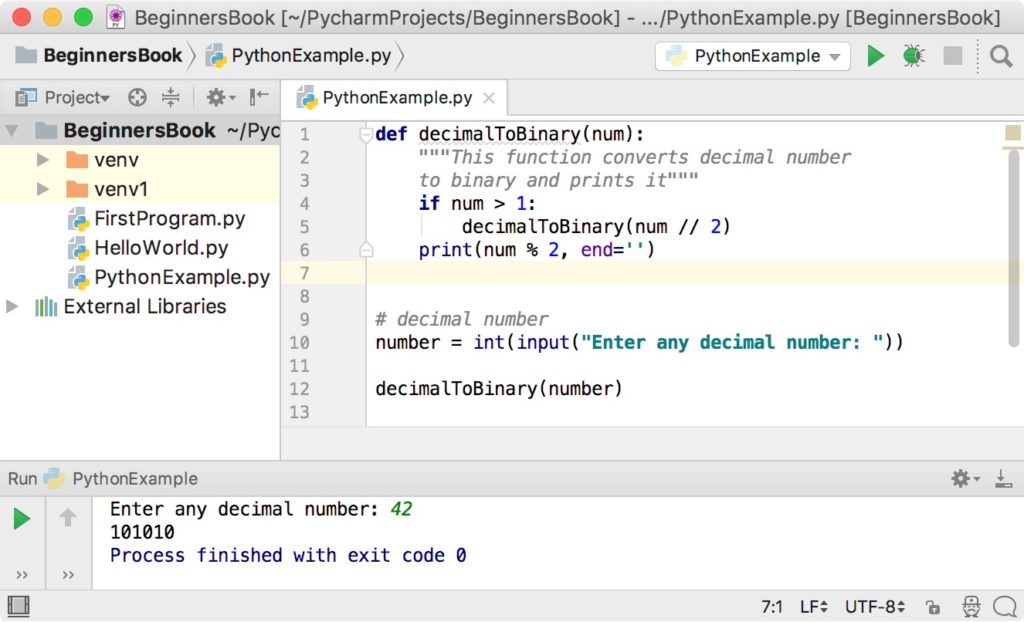
decimalToBinary(num // 2)

print(num % 2, end='')

# decimal number

number = int(input("Enter any decimal number: "))

decimalToBinary(number)

Output:  


**2. Python program to convert decimal number to binary using bin() function**

In this program, we are using a in-built function bin() to convert the decimal number to binary.

# decimal number

number = int(input("Enter any decimal number: "))

# print equivalent binary number

print("Equivalent Binary Number: ", bin(number))

Output:

Enter any decimal number: 42

Equivalent Binary Number: 0b101010

**Python Program to Check If a number is Prime or not**

we will write a program in Python to **check whether the input number is prime or not**. A number is said to be prime if it is only divisible by 1 and itself. For example 13 is a prime number because it is only divisible by 1 and 13, on the other hand 12 is not a prime number because it is divisible by 2, 4, 6 and number itself.

**Checking if number is prime or not**

A prime number is always positive so we are checking that in the beginning of the program.

We are dividing the input number by all the numbers in the range of 2 to (number – 1) to see whether there are any positive divisors other than 1 and number itself.

If any divisor is found then we display that the **“number is not a prime number”** else we display that the **“number is a prime number”**.

We are using the break statement in the loop to come out of the loop as soon as any positive divisor is found as there is no further check is required.

# taking input from user

number = int(input("Enter any number: "))

# prime number is always greater than 1

if number > 1:

for i in range(2, number):

if (number % i) == 0:

print(number, "is not a prime number")

break

else:

print(number, "is a prime number")

# if the entered number is less than or equal to 1

# then it is not prime number

else:

print(number, "is not a prime number")

**Output:**  
