

Assignment 1 Completed By Adarsh Tiwari

Problem 1: Find GCD of Two Numbers

Statement: Write a Python program to find the GCD of two numbers (take input from the user).

Approach:

1. Take two integers as input.
2. If either number is zero, display a message.
3. Otherwise, find the smaller number, loop backward from it to 1, and check divisibility.
4. The first number that divides both is the GCD. Print it.

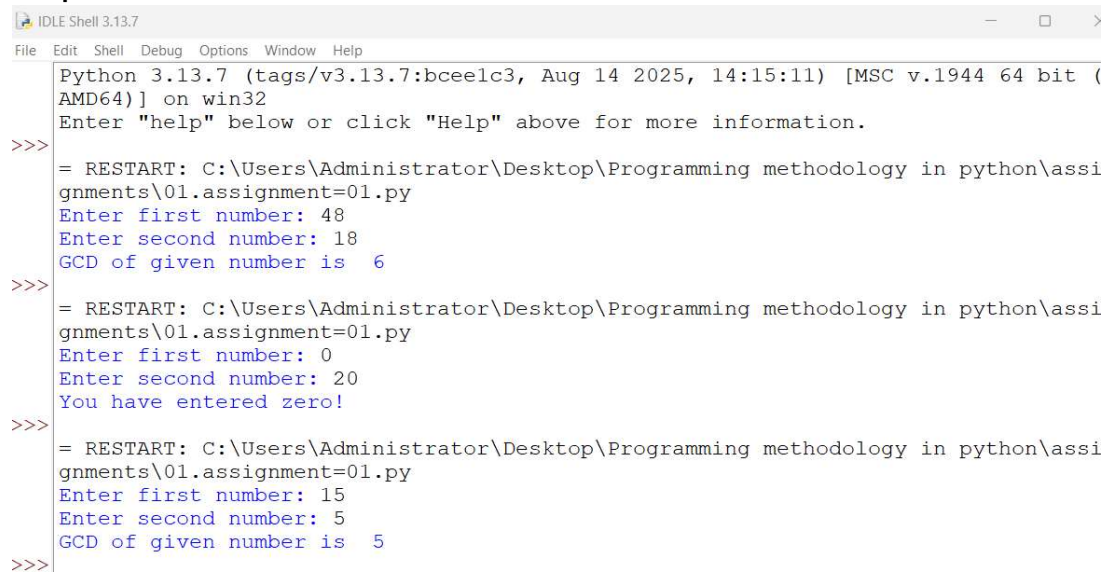
```
num1 = int(input("Enter first number: "))
num2 = int(input("Enter second number: "))

if num1 == 0 or num2 == 0:
    print('You have entered zero!')
else:

    n = min(num1, num2)

    for i in range(n, 0, -1):
        if (num1 % i == 0) and (num2 % i == 0):
            print("GCD of given number is ", i)
            break
```

Sample Runs:



```
IDLE Shell 3.13.7
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.

>>> = RESTART: C:\Users\Administrator\Desktop\Programming methodology in python\assignments\01.assignment=01.py
Enter first number: 48
Enter second number: 18
GCD of given number is 6

>>> = RESTART: C:\Users\Administrator\Desktop\Programming methodology in python\assignments\01.assignment=01.py
Enter first number: 0
Enter second number: 20
You have entered zero!

>>> = RESTART: C:\Users\Administrator\Desktop\Programming methodology in python\assignments\01.assignment=01.py
Enter first number: 15
Enter second number: 5
GCD of given number is 5

>>>
```

Problem 2: Factorial of a Positive Integer

Statement: Write a Python program to find the factorial of any given positive integer (take input from the user).

Approach:

1. Take integer input n.
2. If $n < 0$, print error.
3. If n is 0 or 1, factorial is 1.
4. Else, loop from 2 to n, multiplying to get factorial.
5. Print factorial.

Code:

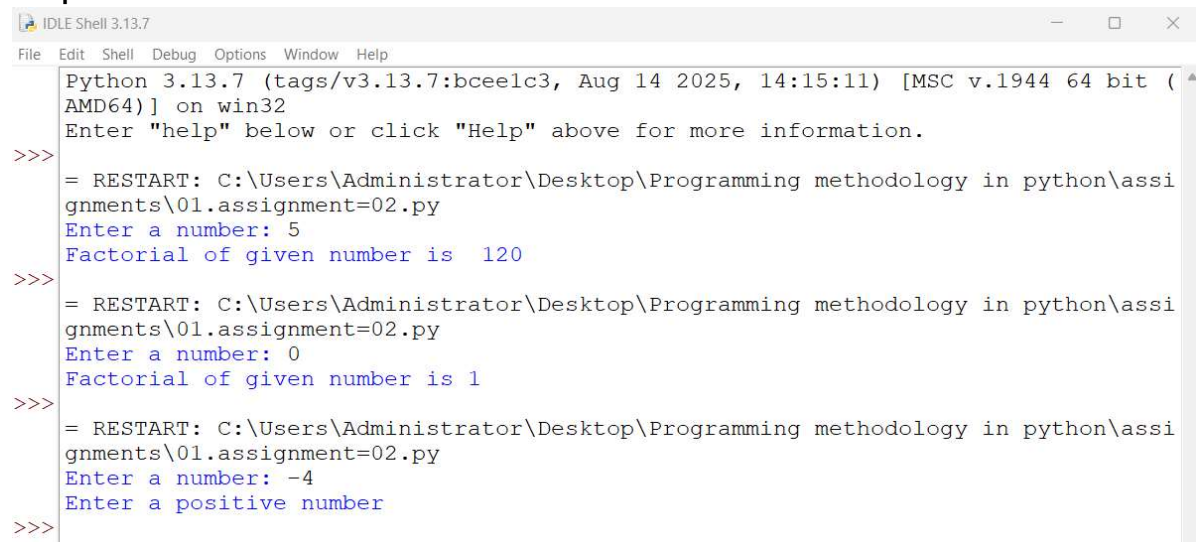
```
n = int( input("Enter a number: "))

if n<0:
    print("Enter a positive number")

elif n == 0 or n == 1:
    print("Factorial of given number is 1")

else:
    f = 1
    for i in range(2, n+1):
        f = f*i
    print("Factorial of given number is ", f)
```

Sample Runs:



```
IDLE Shell 3.13.7
File Edit Shell Debug Options Window Help
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
= RESTART: C:\Users\Administrator\Desktop\Programming methodology in python\assignments\01.assignment=02.py
Enter a number: 5
Factorial of given number is 120
>>>
= RESTART: C:\Users\Administrator\Desktop\Programming methodology in python\assignments\01.assignment=02.py
Enter a number: 0
Factorial of given number is 1
>>>
= RESTART: C:\Users\Administrator\Desktop\Programming methodology in python\assignments\01.assignment=02.py
Enter a number: -4
Enter a positive number
>>>
```

Problem 3: Sum of Digits of a Positive Integer

Statement: Write a program that prints the sum of the digits of a positive integer n . For example, if $n = 1234$, then output: $1 + 2 + 3 + 4 = 10$.

Approach:

1. Take number input as string.
2. Initialize sum counter $c=0$.
3. Loop through each character, convert to int, add to c .
4. Print final sum.

Code

```
n = input('Enter a positive number:')

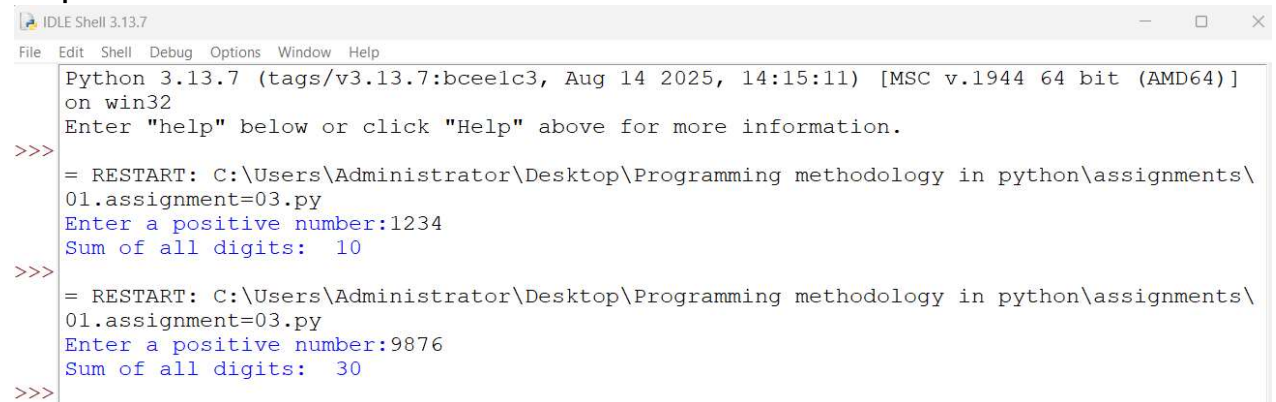
c = 0

for i in n:

    c = c + int(i)

print('Sum of all digits: ', c)
```

Sample Runs:

A screenshot of the IDLE Shell 3.13.7 window. The window title is 'IDLE Shell 3.13.7'. The menu bar includes 'File', 'Edit', 'Shell', 'Debug', 'Options', 'Window', and 'Help'. The shell shows the following interaction:
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)]
on win32
Enter "help" below or click "Help" above for more information.
>>>
= RESTART: C:\Users\Administrator\Desktop\Programming methodology in python\assignments\
01.assignment=03.py
Enter a positive number:1234
Sum of all digits: 10
>>>
= RESTART: C:\Users\Administrator\Desktop\Programming methodology in python\assignments\
01.assignment=03.py
Enter a positive number:9876
Sum of all digits: 30
>>>

Problem 4: Leap Year Checker

Statement: Write a Python program that prints whether a given year is a leap year or not. Input 1900 should output 'Not a leap year'.

Approach:

1. Take year input.
2. If divisible by 400 → Leap year.
3. Else if divisible by 4 but not 100 → Leap year.
4. Else → Not a leap year.
5. Print result.

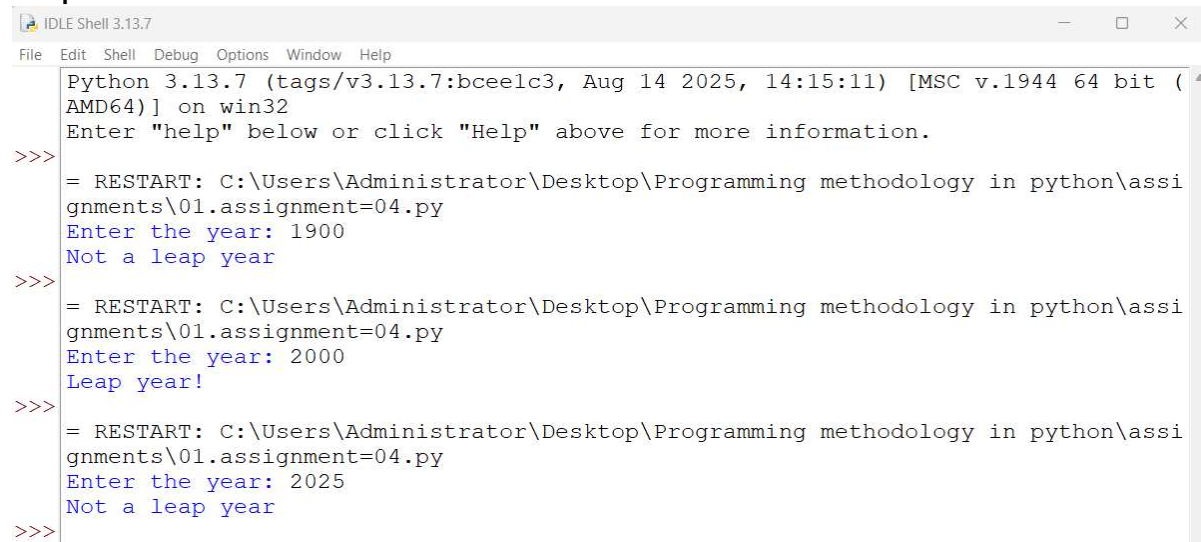
Code:

```
n = (int(input("Enter the year:
")))

if n % 400 == 0:
    print('Leap year!')

elif n%100 != 0 and n%4 == 0:
    print('Leap year!')

else:
    print('Not a leap year')
```

Sample Runs:A screenshot of the IDLE Shell 3.13.7 window. The window title is "IDLE Shell 3.13.7". The menu bar includes "File", "Edit", "Shell", "Debug", "Options", "Window", and "Help". The shell area shows the following text:

```
Python 3.13.7 (tags/v3.13.7:bce1c3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.

>>> = RESTART: C:\Users\Administrator\Desktop\Programming methodology in python\assig
gnments\01.assignment=04.py
Enter the year: 1900
Not a leap year

>>> = RESTART: C:\Users\Administrator\Desktop\Programming methodology in python\assig
gnments\01.assignment=04.py
Enter the year: 2000
Leap year!

>>> = RESTART: C:\Users\Administrator\Desktop\Programming methodology in python\assig
gnments\01.assignment=04.py
Enter the year: 2025
Not a leap year

>>>
```

Problem 5: Time-based Greeting Program

Statement: Write a Python program that takes as input the time of the day, and greets Good Morning, Good Afternoon, or Good Evening depending on the time.

Approach:

1. Take hour input and AM/PM.
2. If AM: 1-11 → Morning, 12 → Evening.
3. If PM: 12 or 1-5 → Afternoon, 6-11 → Evening.
4. Else → Invalid.
5. Print greeting.

Code:

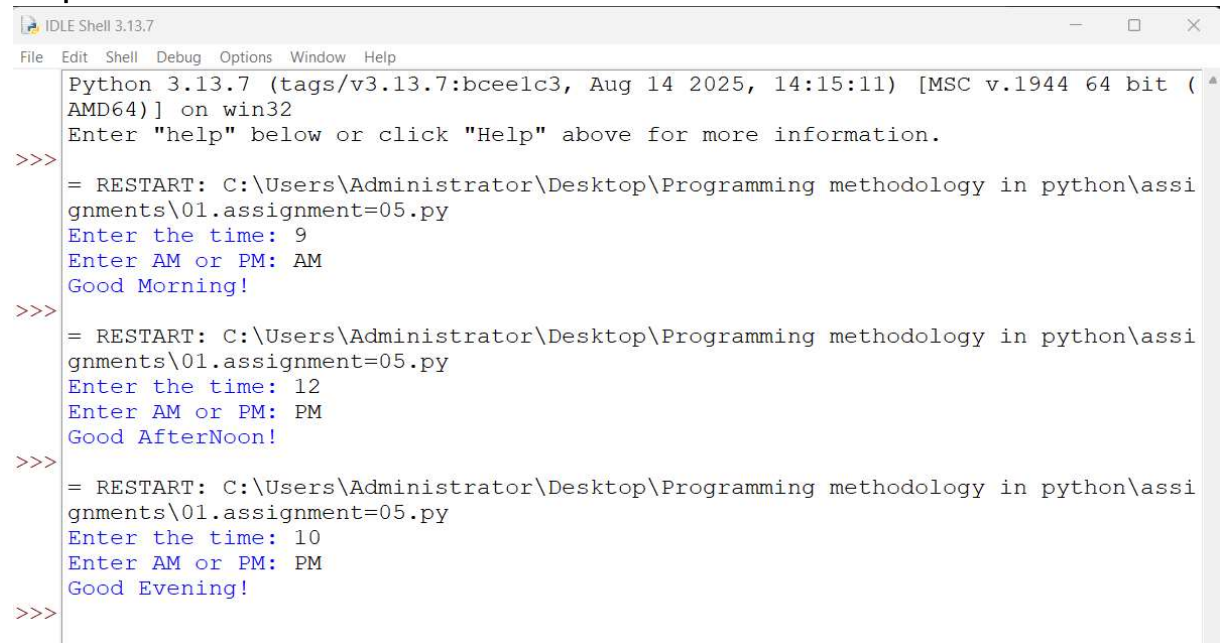
```
time = int(input("Enter the time: " ))

mer = input("Enter AM or PM: ")

if mer.upper() == 'AM':
    if time >= 1 and time <= 11:
        print("Good Morning!")
    elif time == 12:
        print("Good Evening!")

elif mer.upper() == 'PM':
    if time == 12 or (time >= 1 and time <= 5):
        print("Good AfterNoon!")
    elif time > 5 and time <= 11:
        print("Good Evening!")

else:
    print("Invalid Time!")
```

Sample Runs:

```
IDLE Shell 3.13.7
File Edit Shell Debug Options Window Help
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
= RESTART: C:\Users\Administrator\Desktop\Programming methodology in python\assig
nments\01.assignment=05.py
Enter the time: 9
Enter AM or PM: AM
Good Morning!
>>>
= RESTART: C:\Users\Administrator\Desktop\Programming methodology in python\assig
nments\01.assignment=05.py
Enter the time: 12
Enter AM or PM: PM
Good AfterNoon!
>>>
= RESTART: C:\Users\Administrator\Desktop\Programming methodology in python\assig
nments\01.assignment=05.py
Enter the time: 10
Enter AM or PM: PM
Good Evening!
>>>
```

Problem 6: Rock, Paper, Scissors Game

Statement: Write a Python program to play Rock, Paper, Scissors with the computer. Keep score of wins, losses, and ties.

Approach:

1. Import random module.

2. Define choices list.
3. Initialize counters.
4. Use while True loop for rounds.
5. Get user input, validate, pick random computer choice.
6. Compare and update counters.
7. Ask to play again, break if not.
8. Print final results.

Code:

```
import random

print("Let's Play Rock, Paper, Scissor game!")

choices = ["rock", "paper", "scissor"]
draw_count = comp_count = user_count = 0

while True:
    user = input("Enter rock, paper, or scissor: ").lower()
    if user not in choices:
        print("Invalid choice, try again.")
        continue

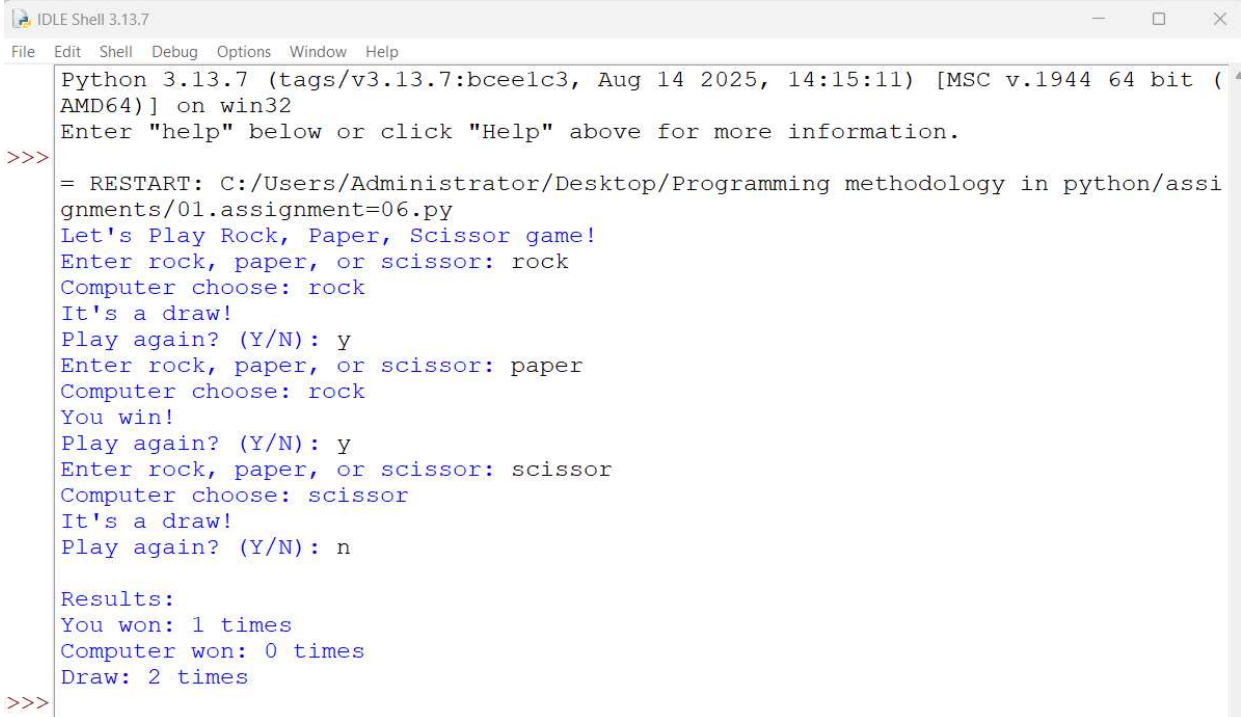
    comp = random.choice(choices)
    print("Computer choose:", comp)

    if user == comp:
        print("It's a draw!")
        draw_count += 1
    elif (user == "rock" and comp == "scissor") or \
        (user == "paper" and comp == "rock") or \
        (user == "scissor" and comp == "paper"):
        print("You win!")
        user_count += 1
    else:
        print("You lose!")
        comp_count += 1

    if input("Play again? (Y/N): ").upper() != 'Y':
        break

print()
print('Results: ')
print('You won:', user_count, 'times')
print('Computer won:', comp_count, 'times')
print('Draw:', draw_count, 'times')
```

Sample Runs:



```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
= RESTART: C:/Users/Administrator/Desktop/Programming methodology in python/assignments/01.assignment=06.py
Let's Play Rock, Paper, Scissor game!
Enter rock, paper, or scissor: rock
Computer choose: rock
It's a draw!
Play again? (Y/N): y
Enter rock, paper, or scissor: paper
Computer choose: rock
You win!
Play again? (Y/N): y
Enter rock, paper, or scissor: scissor
Computer choose: scissor
It's a draw!
Play again? (Y/N): n

Results:
You won: 1 times
Computer won: 0 times
Draw: 2 times
>>>
```

Thankyou.