

Python Assignment

Name: Shivam Gupta

Enrollment Number: 0176AL231120

Batch: 5

Batch Time: 10:30 AM

#Basic If-Else Problems :

#1 wap to check a number is positive , negative or zero.

```
'''
num = int(input("enter the numeber"))
if num>0:
    print("The number is positve")
elif num<0:
    print("The number is negative")
else :
    print("The number is zero")
```

#OUTPUT -

```
enter the numeber-45
The number is negative
'''
```

#2 Write a program to check whether a number is even or odd.

```
'''
num = int(input("enter the numeber : "))
if num % 2==0:
    print("The num is even")
else:
    print("The num is odd")
```

#OUTPUT -

```
enter the numeber : 2
The num is even
'''
```

#3 Write a program to check if a given year is a leap year or not.

```
'''
year = int(input("enter the year"))
if year%4==0:
    if year%100==0:
        if year%400==0:
            print(year,"is a leap year")
        else:
            print(year,"is not a leap year")
    else:
        print(year,"is a leap year")
else:
    print(year,"is not a leap year")
```

#OUTPUT -

```
enter the year2001
2001 is not a leap year
```

```
enter the year2004
2004 is a leap year
```

```
'''
```

```
#4 Write a program to find the greatest of two numbers.
```

```
'''
```

```
num1 = int(input("enter the first numeber"))
num2 = int(input("enter the first numeber"))
if num1 > num2:
    print(num1 , "is greatest")
else:
    print(num2 , "is greatest")
```

```
#OUTPUT -
```

```
enter the first numeber23
enter the first numeber32
32 is greatest
'''
```

```
#5 Write a program to check whether a person is eligible to vote (age
>= 18).
```

```
'''
```

```
name = str(input("enter the name :"))
age = int(input("enter the age : "))
if age >= 18:
    print("hurrahhh" , name , "is eligible to vote")
else:
    print("Sorry" , name , "is not eligible to vote")
```

```
#OUTPUT -
```

```
enter the name :SHIVAM
enter the age : 19
hurrahhh SHIVAM is eligible to vote
```

```
enter the name :RAM
enter the age : 17
Sorry RAM is not eligible to vote
'''
```

```
#6 Write a program to check whether a given character is a vowel or
consonant.
```

```
'''
```

```
char = str(input("enter the character :"))
char = char.lower()
if char == "a" or char == "e" or char == "i" or char == "o" or char
=="u" :
    print(char , "is a vowel")
else:
    print(char , "is a consonant")
```

```
#OUTPUT -
```

```
enter the character :A
a is a vowel
```

```
enter the character :X
x is a consonant
'''
```

#7 Write a program to check if a number is divisible by 5.

```
'''
num = int(input("enter the number : "))
if num % 5 == 0 :
    print(num , "is divisible by 5")
else:
    print(num , "is not divisible by 5")
```

#OUTPUT -

```
enter the number : 550
550 is divisible by 5
```

```
enter the number : 44
44 is not divisible by 5
'''
```

#8 Write a program to determine whether a given number is a single-digit, two-digit, or more than two-digit number.

```
'''
num = int(input("enter the number : "))
if num > 0 :
    if num < 10:
        print(num , " is a single-digit number .")
    elif num < 100:
        print(num , " is a two-digit number .")
    else:
        print(num , " is a more than two- digit number .")
else:
    if num > -10:
        print(num , " is a single-digit number .")
    elif num > -100:
        print(num , " is a two-digit number .")
    else:
        print(num , " is a more than two- digit number .")
```

#OUTPUT -

```
enter the number : 0
0 is a single-digit number .
```

```
enter the number : -68
-68 is a two-digit number .
```

```
enter the number : 100
100 is a more than two- digit number .
```

```
'''
```

#9 Write a program to check whether a student has passed or failed (passing marks = 40).

```
'''
```

```
name = str(input("enter the name "))
marks = float(input("enter the marks of the students : "))
if marks >= 40:
    print(name , "is passed .")
else:
    print(name , " is failed")
```

#OUTPUT -

```
enter the name shyama
enter the marks of the students : 29
shyama is failed
```

```
enter the name ram
enter the marks of the students : 40
ram is passed .
'''
```

#10 Write a program to find whether the entered number is a multiple of both 3 and 7.

```
'''
```

```
num = int(input("enter the number : "))
if num % 3 == 0 and num % 7 == 0:
    print(num , " is a multiple of both 3 and 7 . ")
else :
    print(num , " is not a multiple of both ")
```

#OUTPUT -

```
enter the number : 63
63 is a multiple of both 3 and 7 .
```

```
enter the number : 49
49 is not a multiple of both
```

```
'''
```

Ladder If & Nested If :

#1 Write a program to find the greatest among three numbers.

```
'''
```

```
num1 = int(input("enter the 1st number : "))
num2 = int(input("enter the 2nd number : "))
num3 = int(input("enter the 3rd number : "))
if num1 >= num2 and num1 >= num3 :
```

```

    print(num1 , " is greatest number.")
elif num2 >= num1 and num2 >= num3 :
    print(num2 , " is greatest number.")
else:
    print(num3 , " is greatest number.")

```

#OUTPUT -

```

enter the 1st number : 69
enter the 2nd number : 96
enter the 3rd number : 88
96 is greatest number.
'''

```

#2 Write a program to classify a person based on age: Child (<13), Teenager (13-19), Adult (20-59), Senior (60+).

```

'''
name = str(input("enter the name of the person : "))
age = int(input("enter the age of the person : "))
if age <13 :
    print(name , "is a child .")
elif age >= 13 and age < 20:
    print(name , "is a Teenager .")
elif age >=20 and age < 60:
    print(name , "is a Adult .")
else:
    print(name , "is a Senior .")

```

#output -

```

enter the name of the person : shivam
enter the age of the person : 20
shivam is a Adult .
'''

```

#3 Write a program to assign grades based on marks:

```

'''
90-100: A,
75-89: B,
50-74: C,
35-49: D,
<35: Fail.

name = str(input("enter the name of the person : "))
marks = int(input("enter the marks of the person : "))
if marks <= 100 and marks>= 90 :
    print(name , "is having A-GRADE with " , marks , " marks .")
elif marks <90 and marks >=75:
    print(name , "is having B-GRADE with " , marks , " marks .")
elif marks < 75 and marks >= 50:
    print(name , "is having C-GRADE with " , marks , " marks .")
elif marks <50 and marks >= 35:
    print(name , "is having D-GRADE with " , marks , " marks .")
else:
    print(name , " is Fail with " , marks , " marks .")

```

#OUTPUT -

```
enter the name of the person : Ram
enter the marks of the person : 79
Ram is having B-GRADE with 79 marks .
'''
```

#4 Write a program to check the type of triangle (equilateral, isosceles, or scalene) based on sides.

```
'''
s1 = int(input("enter the side : "))
s2 = int(input("enter the side : "))
s3 = int(input("enter the side : "))
if s1+s2>s3 and s2+s3>s1 and s3+s1>s2:
    if s1==s2==s3 :
        print("The Triangle is equilateral triangle with sides - "
,s1,",",s2,",",s3)
    elif s1 == s2 or s2 == s3 or s3==s1:
        print("The Triangle is isosceles triangle with sides - "
,s1,s2,s3)
    else:
        print("The Triangle is scalene triangle with sides - "
,s1,s2,s3)
else:
    print("The input sides do not form triangle")
'''
```

#OUTPUT -

```
enter the side : 45
enter the side : 45
enter the side : 69
The Triangle is isosceles triangle with sides - 45 45 69
'''
```

#5 Write a program to check if a character is uppercase, lowercase, digit, or special symbol.

```
'''
ch = input("enter the char :")
if len(ch) != 1:
    print("Entered character is not single digit.")
elif 'A' <= ch <= 'Z':
    print(ch , " is in uppercase.")
elif 'a' <= ch <= 'z':
    print(ch , " is in lowercase.")
elif '0' <= ch <= '9':
    print(ch , " is a digit.")
else:
    print(ch , " is a symbol.")
'''
```

#OUTPUT -

```
enter the char :5
5 is a digit.
'''
```

#6 Write a program to calculate electricity bill based on units:

```
'''
```

```
Up to 100 units: â, 15 per unit,  
101â€"200 units: â, 17 per unit,  
Above 200 units: â, 10 per unit.
```

```
unit = int(input("enter the unit : "))  
if unit <= 100 and unit>0:  
    print("Electricity Bill is :", unit*5)  
elif 101<=unit<=200:  
    print("Electricity Bill is :", unit*7)  
elif unit >200:  
    print("Electricity Bill is :", unit*10)  
else:  
    print("Invalid input")
```

#OUTPUT -

```
enter the unit : 101  
Electricity Bill is : 707  
'''
```

#7 Write a program to determine the largest of four numbers using nested if.

```
'''
```

```
num1 = int(input("enter the 1st number : "))  
num2 = int(input("enter the 2nd number : "))  
num3 = int(input("enter the 3rd number : "))  
num4 = int(input("enter the 4th number : "))  
if num1 >= num2 :  
    if num1 >= num3 :  
        if num1 >= num4 :  
            print(num1 , " is greatest number.")  
        else:  
            print(num4 , " is greatest number.")  
    else:  
        if num3 >= num4 :  
            print(num3 , " is greatest number.")  
        else:  
            print(num4 , " is greatest number.")  
else:  
    if num2 >= num3 :  
        if num2 >= num4 :  
            print(num2 , " is greatest number.")  
        else:  
            print(num4 , "is greatest number.")  
    else:  
        if num3 >= num4:  
            print(num3 , "is greatest number.")  
        else:  
            print(num4 , "is greatest number.")
```


#OUTPUT -

```
enter the 1st number : 32
enter the 2nd number : 2
enter the 3rd number : 1
enter the 4th number : 32
32 is greatest number.
'''
```

#8 Write a program to check if a given year is a century year and also a leap year.

```
'''
year = int(input("Enter a year: "))
if year % 100 == 0:
    print(year, "is a Century Year.")
    if year % 400 == 0:
        print(year, "is also a Leap Year.")
    else:
        print(year, "is not a Leap Year.")
else:
    print(year, "is not a Century Year.")
    if year % 4 == 0:
        print(year, "is a Leap Year.")
    else:
        print(year, "is not a Leap Year.")
'''
```

#OUTPUT -

```
Enter a year: 2000
2000 is a Century Year.
2000 is also a Leap Year.
'''
```

#9 Write a program to classify BMI value: Underweight (<18.5), Normal (18.5-24.9), Overweight (25-29.9), Obese (30+).

```
'''
name = str(input("enter the name of the person : "))
bmi = float(input("enter the bmi value of the person : "))
if bmi <18.5 :
    print(name , "is Underweight.")
elif 18.5 <= bmi <= 24.9:
    print(name , "is Normal.")
elif 25 <= bmi <=29.9:
    print(name , "is Overweight.")
else:
    print(name , "is Obese.")
'''
```

#OUTPUT -

```
enter the name of the person : shivam
enter the bmi value of the person : 24.9
shivam is Normal.
'''
```

#10 Write a program to display the smallest number among three using nested if.

```
'''
a = int(input("Enter first number: "))
b = int(input("Enter second number: "))
c = int(input("Enter third number: "))
if a < b:
    if a < c:
        smallest = a
    else:
        smallest = c
else:
    if b < c:
        smallest = b
    else:
        smallest = c
print("The smallest number is:", smallest)
'''
```

#OUTPUT -

```
Enter first number: 2
Enter second number: 69
Enter third number: 88
The smallest number is: 2
'''
```

For Loop Problems:

#1. Write a program using a for loop to print all Armstrong numbers between 100 and 999. (Armstrong number: # sum of cubes of digits equals the number itself. Example: 153 => $1^3 + 5^3 + 3^3 = 153$).

```
'''
print("Armstrong numbers between 100 and 999 are:")
for num in range(100, 1000):
    sum_of_cubes = 0
    temp = num
    while temp > 0:
        digit = temp % 10
        sum_of_cubes += digit ** 3
        temp //= 10
    if num == sum_of_cubes:
        print(num, end=" ")
'''
```

#OUTPUT -

```
Armstrong numbers between 100 and 999 are:
153 370 371 407
'''
```

#2 Write a program to generate and display the first n prime numbers using a for loop.

```
'''
n = int(input("Enter how many prime numbers you want: "))
count = 0
num = 2
```

```
print(f"First {n} prime numbers are:")
```

```
while count < n:
    is_prime = True
    for i in range(2, int(num ** 0.5) + 1):
        if num % i == 0:
            is_prime = False
            break
    if is_prime:
        print(num, end=" ")
        count += 1
    num += 1
```

#OUTPUT -

```
Enter how many prime numbers you want: 10
First 10 prime numbers are:
2 3 5 7 11 13 17 19 23 29
'''
```

#3 Write a program to display all numbers from 1 to 500 that are divisible by 3,
#but the sum of their digits should not exceed 10.

```
'''
print("All numbers from 1 to 500 that are divisible by 3, but the sum
of their digits should not exceed 10. are :")
for i in range(1, 501):
    if i % 3 == 0:
        d_sum = sum(int(d) for d in str(i))
        if d_sum <= 10:
            print(i, end=" ")
```

#OUTPUT -

```
All numbers from 1 to 500 that are divisible by 3, but the sum of
their digits should not exceed 10. are :
3 6 9 12 15 18 21 24 27 30 33 36 42 45 51 54 60 63 72 81 90 102 105
108 111 114 117 120 123 126 132 135 141
144 150 153 162 171 180 201 204 207 210 213 216 222 225 231 234 240
243 252 261 270 300 303 306 312 315 321
324 330 333 342 351 360 402 405 411 414 420 423 432 441 450
```

```
'''
```

#4 Write a program using a for loop to print a pyramid of stars (*)
of height n. Example for n=4:

```
'''
    *
   ***
  *****
```

```
n = int(input("Enter the height of the pyramid: "))
for i in range(1, n + 1):
    spaces = n - i
    stars = 2 * i - 1
    print(" " * spaces + "*" * stars)
```

#OUTPUT -

Enter the height of the pyramid: 5

```
  *
 ***
*****
*****
*****
'''
```

#5 Write a program to accept a string and check whether it is a pangram (contains all 26 alphabets at least once)
#using a for loop.

```
'''
def pangram(arr):
    arr=arr.lower()
    a=set()
    for ch in arr:
        if ch >='a' and ch <='z':
            a.add(ch)

    return len(a)==26
```

```
ar=input("enter the text : ")
print(pangram(ar))
```

#OUTPUT -

```
enter the text :The quick brown fox jumps over the lazy dog
True
'''
```

#6 Write a program using a for loop to print all twin primes between 1 and 100. (Twin primes: pairs of prime numbers with a difference of 2, e.g., (3,5), (11,13))

```
'''
print("Twin primes between 1 and 100 are:")

for n in range(2, 100):

    is_prime_n = True    #N
    for i in range(2, int(n**0.5) + 1):
        if n % i == 0:
            is_prime_n = False
```

```

        break

    is_prime_n2 = True    #N+2
    for j in range(2, int((n+2)**0.5) + 1):
        if (n + 2) % j == 0:
            is_prime_n2 = False
            break

    if is_prime_n and is_prime_n2 and (n + 2) <= 100:
        print(f"({n}, {n+2})")

```

#OUTPUT -

Twin primes between 1 and 100 are:

```

(3, 5)
(5, 7)
(11, 13)
(17, 19)
(29, 31)
(41, 43)
(59, 61)
(71, 73)
'''

```

#7 Write a program that accepts a number from the user and prints whether it is a Harshad number (number divisible by the sum of its digits) using a for loop.

```

'''
n=int(input("enter the number"))
d_sum = sum(int(d) for d in str(n))
if n % d_sum == 0:
    print(n , " is a Harshad number")
else:
    print(n , " is not a Harshad number")
'''

```

#OUTPUT -

```

enter the number111
111 is a Harshad number
'''

```

#8 Write a program to generate Pascal's Triangle up to n rows using a for loop

```

'''
n = int(input("Enter the number of rows: "))
for i in range(n):
    print(" " * (n - i), end="")
    val = 1
    for j in range(i + 1):
        print(val, end=" ")
        val = val * (i - j) // (j + 1)
    print()
'''

```

#OUTPUT -

Enter the number of rows: 5

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
'''
```

#9 Write a program using a for loop to display the sum of the series:

$1\hat{A}^2 + 2\hat{A}^2 + 3\hat{A}^2 + \dots + n\hat{A}^2$

```
'''
n = int(input("Enter the value of n: "))
sum_of_squares = 0
for i in range(1, n + 1):
    sum_of_squares += i ** 2
print(f"The sum of the series  $1\hat{A}^2 + 2\hat{A}^2 + \dots + \{n\}\hat{A}^2$  is: {sum_of_squares}")
```

#OUTPUT -

Enter the value of n: 4

The sum of the series $1\hat{A}^2 + 2\hat{A}^2 + \dots + 4\hat{A}^2$ is: 30

'''

#10 Write a program that accepts a number from the user and prints whether it is a Strong number (sum of

factorials of digits = number itself) using a for loop. Example: 145 => $1! + 4! + 5! = 145$.

```
'''
num = int(input("Enter a number: "))
temp = num
sum_of_factorials = 0
while temp > 0:
    digit = temp % 10
    fact = 1
    for i in range(1, digit + 1):
        fact *= i
    sum_of_factorials += fact
    temp //= 10
if sum_of_factorials == num:
    print(num, "is a Strong Number")
else:
    print(num, "is NOT a Strong Number")
```

#OUTPUT -

Enter the number : 145

145 is a Strong Number

'''

While Loop Problems:

#11. Write a program using a while loop to find the reverse of a number and check if the reversed number is prime. Example: Input = 73 → Reverse = 37 → Prime.

```
'''
n=(input("enter the number"))
nu = n[::-1]
num =int(nu)
if num > 1:
    for i in range(2, num):
        if num % i == 0:
            print(num, "is not a prime number")
            break
    else:
        print(num, "is a prime number")
else:
    print(num, "is not a prime number")
```

#OUTPUT -

```
enter the number13
31 is a prime number
'''
```

#12. Write a program that continues to accept numbers from the user until the sum of digits of all numbers entered becomes greater than 100.

```
'''
total_sum = 0
while total_sum <= 100:
    num = int(input("Enter a number: "))
    digit_sum = 0
    temp = num
    while temp > 0:
        digit_sum += temp % 10
        temp //= 10
    total_sum += digit_sum
    print(f"Sum of digits of {num} = {digit_sum}, Total sum so far = {total_sum}")
```

```
print("\nStopping input as total sum of digits is greater than 100.")
```

#OUTPUT -

```
Enter a number: 45
Sum of digits of 45 = 9, Total sum so far = 9
Enter a number: 99
Sum of digits of 99 = 18, Total sum so far = 27
Enter a number: 88
Sum of digits of 88 = 16, Total sum so far = 43
Enter a number: 16
Sum of digits of 16 = 7, Total sum so far = 50
```

```

Enter a number: 50
Sum of digits of 50 = 5, Total sum so far = 55
Enter a number: 99
Sum of digits of 99 = 18, Total sum so far = 73
Enter a number: 99
Sum of digits of 99 = 18, Total sum so far = 91
Enter a number: 99
Sum of digits of 99 = 18, Total sum so far = 109

Stopping input as total sum of digits is greater than 100.

```

```
'''
```

#13. Write a program using a while loop to check whether a number is a Duck number (a number containing zero #but not starting with zero, e.g., 202, 1203).

```
'''
```

```

num = input("Enter a number: ")
if num[0] == '0':
    print(num, "is NOT a Duck Number (starts with zero)")
else:
    temp = int(num)
    is_duck = False
    while temp > 0:
        digit = temp % 10
        if digit == 0:
            is_duck = True
            break
        temp //= 10
    if is_duck:
        print(num, "is a Duck Number")
    else:
        print(num, "is NOT a Duck Number (no zero found)")

```

#OUTPUT -

```

Enter a number: 001
001 is NOT a Duck Number (starts with zero)
Enter a number: 202
202 is a Duck Number
'''

```

#14. Write a program using a while loop to accept a number and check if it is a Happy number. (A number is happy if # repeatedly replacing it with the sum of squares of its digits eventually reaches 1). Example: 19 is a happy number.

```
'''
```

```

num = int(input("Enter a number: "))
visited = set() # To store already seen numbers to detect cycles

temp = num
while temp != 1 and temp not in visited:
    visited.add(temp)
    sum_of_squares = 0

```



```

while temp > 0:
    digit = temp % 10
    sum_of_squares += digit ** 2
    temp //= 10
temp = sum_of_squares

if temp == 1:
    print(num, "is a Happy Number")
else:
    print(num, "is NOT a Happy Number")

```

#OUTPUT -

```

Enter a number: 19
19 is a Happy Number
'''

```

#15. Write a program using a while loop to find the largest prime factor of a given number.

```

'''
num = int(input("Enter a number: "))
n = num
largest_prime = 0
i = 2
while i <= n:
    if n % i == 0:
        largest_prime = i
        n //= i
    else:
        i += 1
print("The largest prime factor of", num, "is:", largest_prime)

```

#OUTPUT -

```

Enter a number: 84
The largest prime factor of 84 is: 7
'''

```

#16. Write a program to repeatedly accept a string from the user until the string entered is a palindrome.

```

'''
while True:
    s = input("Enter a string: ")
    if s == s[::-1]:
        print(f"'{s}' is a palindrome. Program ends.")
        break
    else:
        print(f"'{s}' is not a palindrome. Try again.")

```

#OUTPUT -

```

Enter a string: SHIVAM
'SHIVAM' is not a palindrome. Try again.
Enter a string: SHIV

```

```
'SHIV' is not a palindrome. Try again.
Enter a string: YASH
'YASH' is not a palindrome. Try again.
Enter a string: NAMAN
'NAMAN' is a palindrome. Program ends.
'''
```

#17. Write a program using a while loop to compute the sum of digits of a number until the result becomes a single-digit number (Digital root). Example: 9875 => 9+8+7+5=29 => 2+9=11 => 1+1=2.

```
'''
num = int(input("Enter a number: "))
temp = num
while temp > 9:
    sum_of_digits = 0
    while temp > 0:
        sum_of_digits += temp % 10
        temp //= 10
    temp = sum_of_digits
print("The digital root of", num, "is:", temp)
```

#OUTPUT -

```
Enter a number: 6969
The digital root of 6969 is: 3
'''
```

#18. Write a program using a while loop to generate the Collatz sequence for a given number. (Rule: If n is even # => n/2, if odd => 3n+1. Continue until n=1).

```
'''
n = int(input("Enter a number: "))
print("Collatz sequence:")
while n != 1:
    print(n, end=" -> ")
    if n % 2 == 0:
        n //= 2
    else:
        n = 3 * n + 1
print(1)
```

#OUTPUT -

```
Enter a number: 6
Collatz sequence:
6 -> 3 -> 10 -> 5 -> 16 -> 8 -> 4 -> 2 -> 1
'''
```

#19. Write a program using a while loop to accept a number and check whether it is a Kaprekar number.
(Kaprekar number: if square of the number can be split into two parts whose sum equals the number.
Example: $45^2=2025 \Rightarrow 20+25=45$).

```
'''
```

```

num = int(input("Enter a number: "))
sq = num ** 2
sq_str = str(sq)
length = len(sq_str)
i = length - len(str(num))
if i <= 0:
    left = 0
    right = int(sq_str)
else:
    left = int(sq_str[:i])
    right = int(sq_str[i:])

if left + right == num:
    print(num, "is a Kaprekar Number")
else:
    print(num, "is NOT a Kaprekar Number")

```

#OUTPUT -

```

Enter a number: 45
45 is a Kaprekar Number
Enter a number: 12
12 is NOT a Kaprekar Number
'''

```

#20. Write a program to simulate an ATM machine using a while loop where a user can:

```

'''
â€¢ Check balance
â€¢ Deposit money
â€¢ Withdraw money (only if balance is sufficient)
â€¢ Exit
Continue until the user chooses to exit.
'''

```

```

balance = 1000
while True:
    print("\n=== ATM Menu ===")
    print("1. Check Balance")
    print("2. Deposit Money")
    print("3. Withdraw Money")
    print("4. Exit")
    choice = input("Enter your choice (1-4): ")

    if choice == '1':
        print(f"Your current balance is: ${balance}")

    elif choice == '2':
        amount = float(input("Enter amount to deposit: $"))
        if amount > 0:
            balance += amount
            print(f"${amount} deposited successfully. New balance: ${balance}")
        else:
            print("Invalid amount!")

```

```

elif choice == '3':
    amount = float(input("Enter amount to withdraw: $"))
    if amount > balance:
        print("Insufficient balance!")
    elif amount <= 0:
        print("Invalid amount!")
    else:
        balance -= amount
        print(f"${amount} withdrawn successfully. Remaining
balance: ${balance}")

elif choice == '4':
    print("Thank you for using the ATM. Goodbye!")
    break

else:
    print("Invalid choice! Please enter 1-4.")

```

#OUTPUT -

```

=== ATM Menu ===
1. Check Balance
2. Deposit Money
3. Withdraw Money
4. Exit
Enter your choice (1-4): 2
Enter amount to deposit: $2550
$2550.0 deposited successfully. New balance: $3550.0

```

```

=== ATM Menu ===
1. Check Balance
2. Deposit Money
3. Withdraw Money
4. Exit
Enter your choice (1-4): 1
Your current balance is: $3550.0

```

```

=== ATM Menu ===
1. Check Balance
2. Deposit Money
3. Withdraw Money
4. Exit
Enter your choice (1-4): 3
Enter amount to withdraw: $255555
Insufficient balance!

```

```

=== ATM Menu ===
1. Check Balance
2. Deposit Money
3. Withdraw Money
4. Exit
Enter your choice (1-4): 4
Thank you for using the ATM. Goodbye!
'''

```