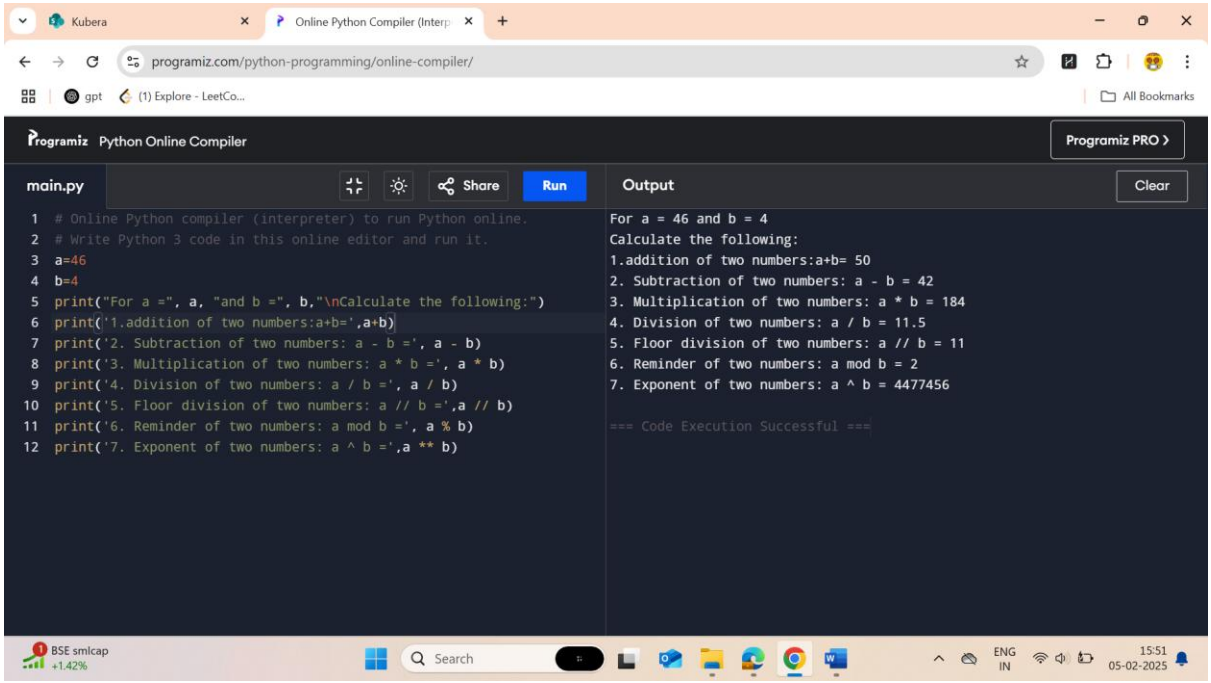
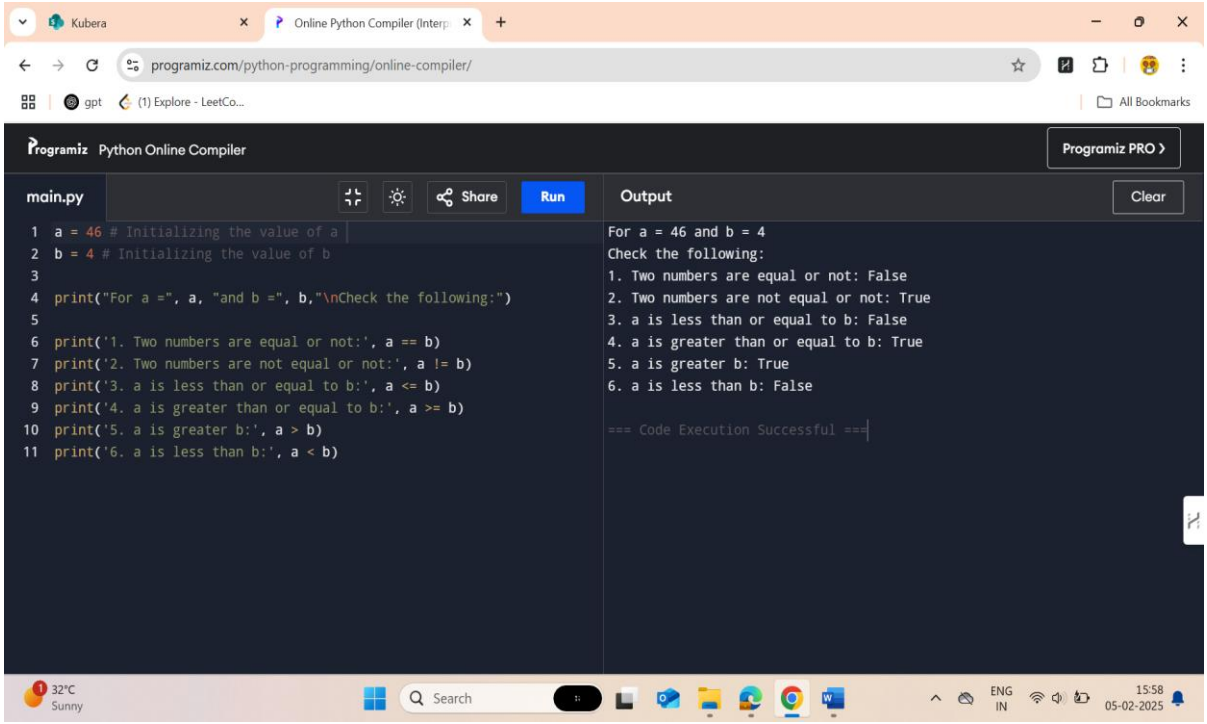
Python Basics

Python operators:

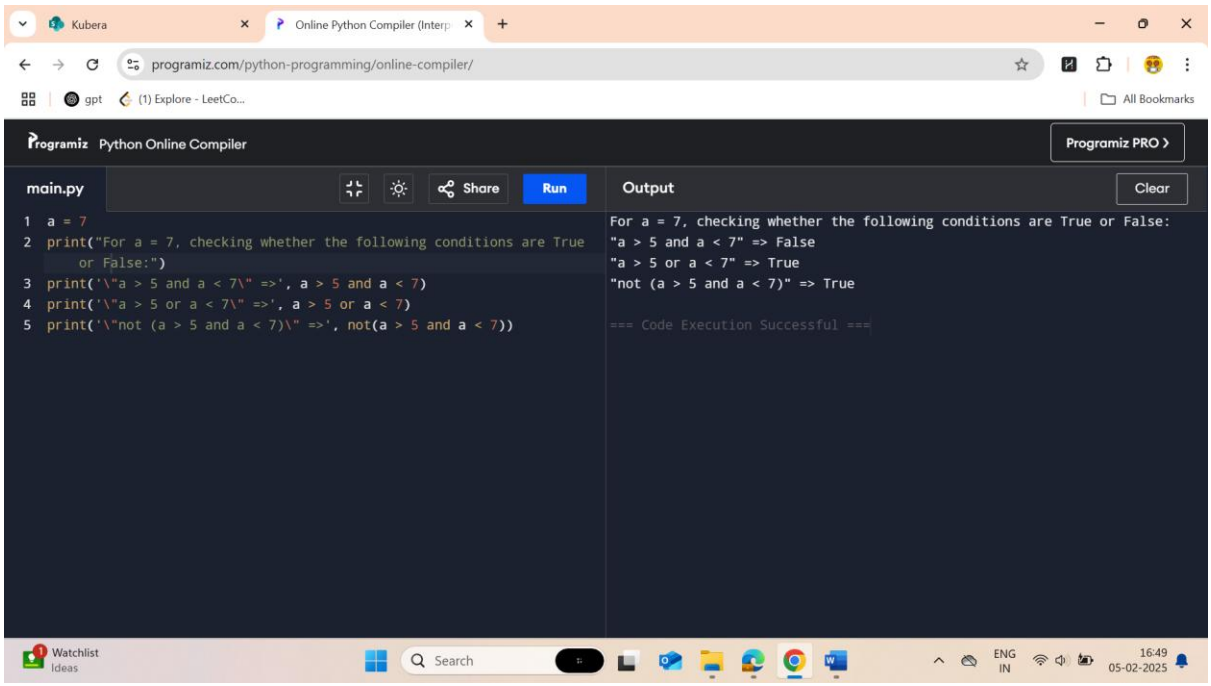
Example1:Arithemetic operations:



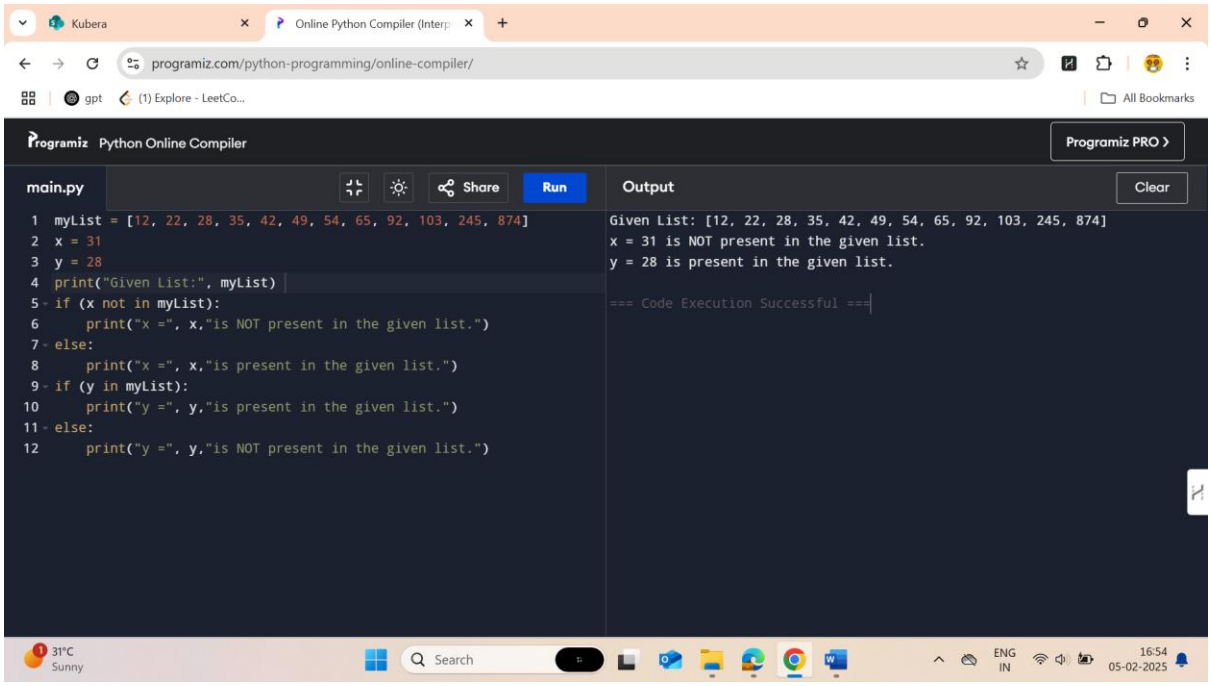
Example 2: Comparison operators in Python.



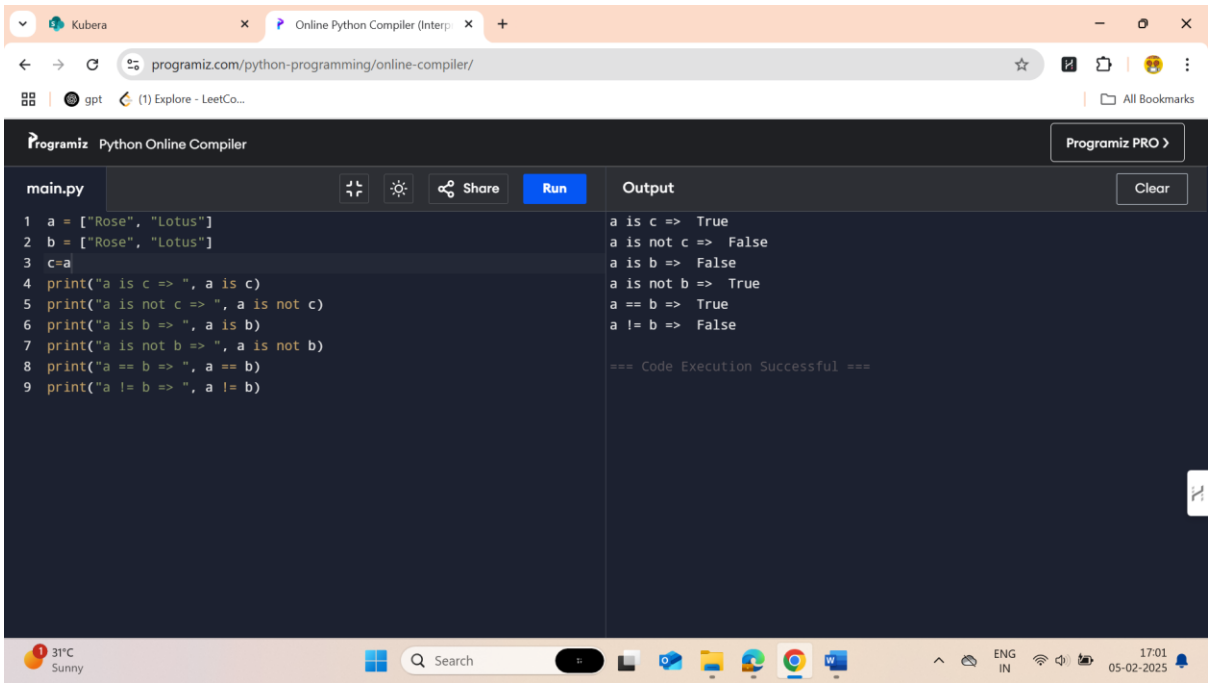
Example 3: examples of Logical operators in Python



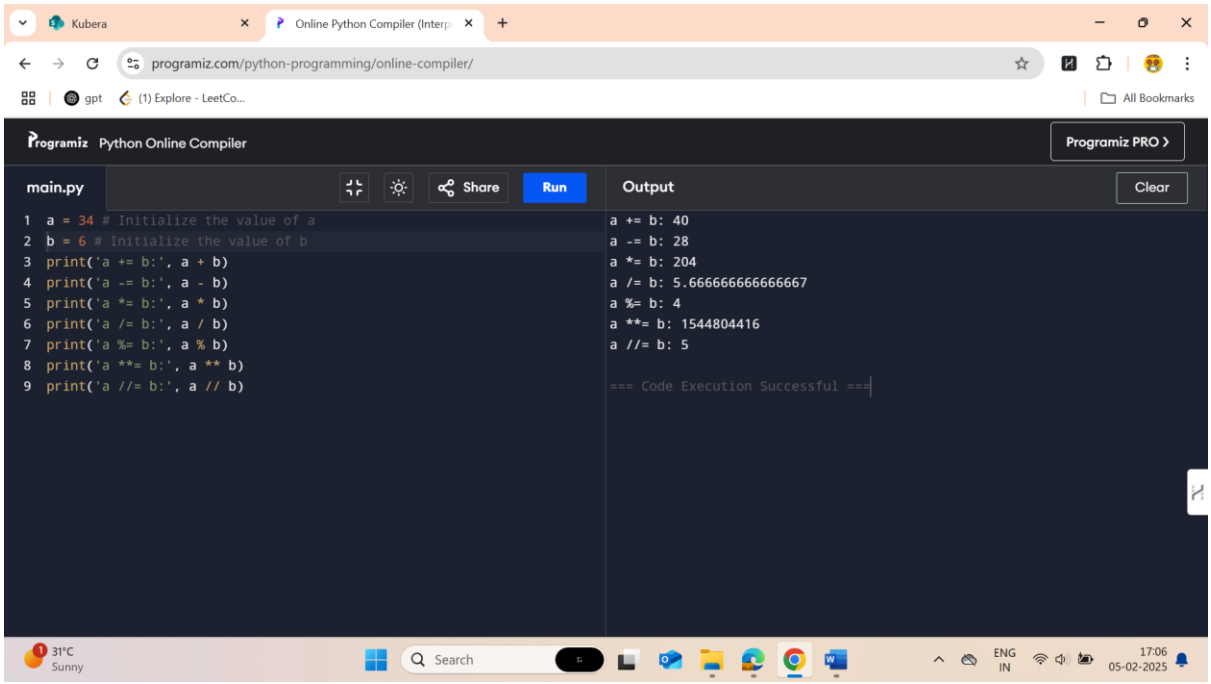
Example 4: examples of Membership operators in Python



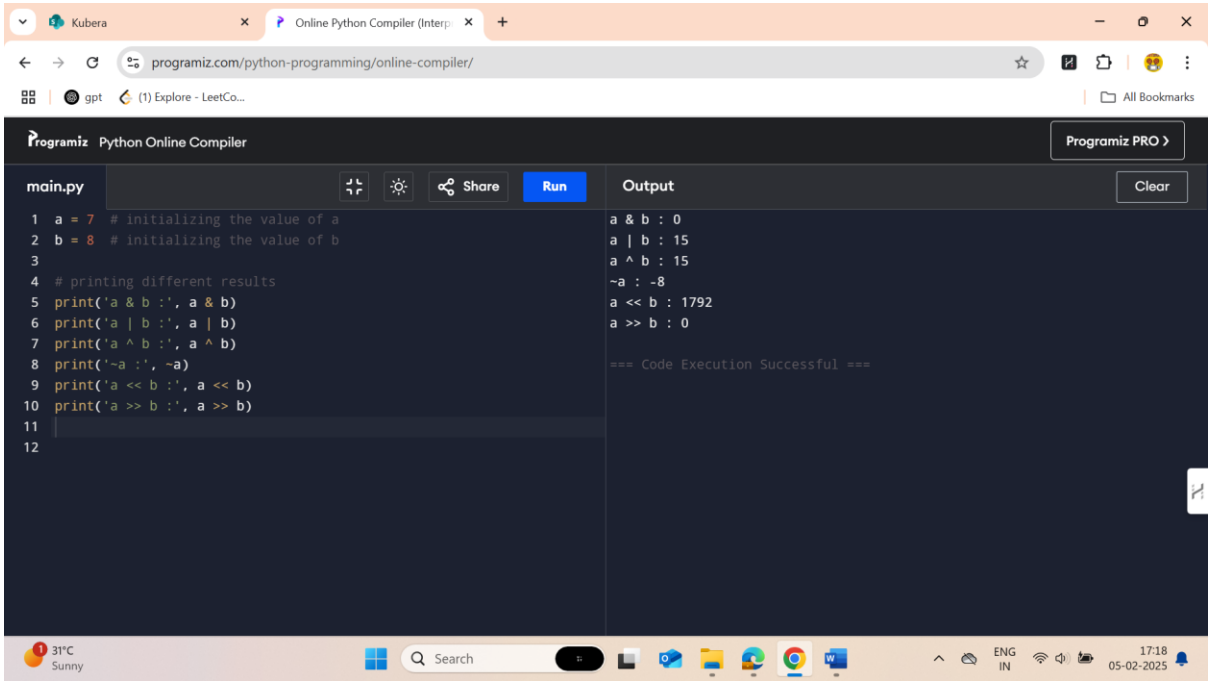
Example 5:examples of Identity operators in Python



Example 6: examples of Assignment operators in Python

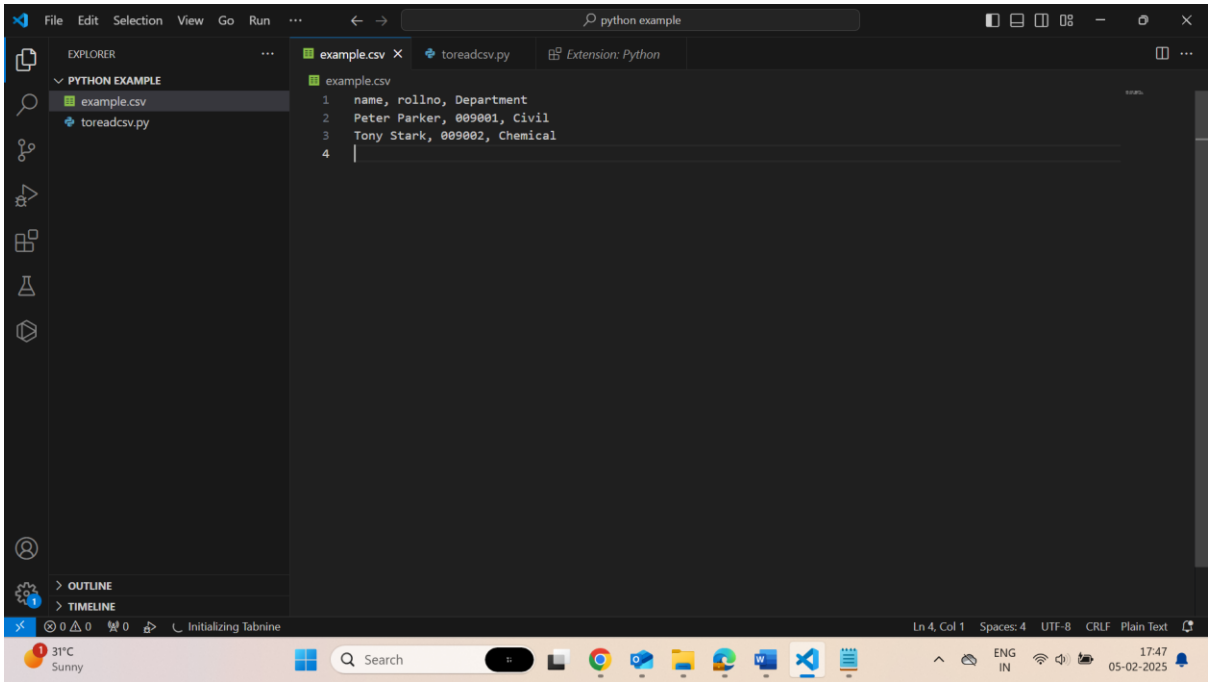


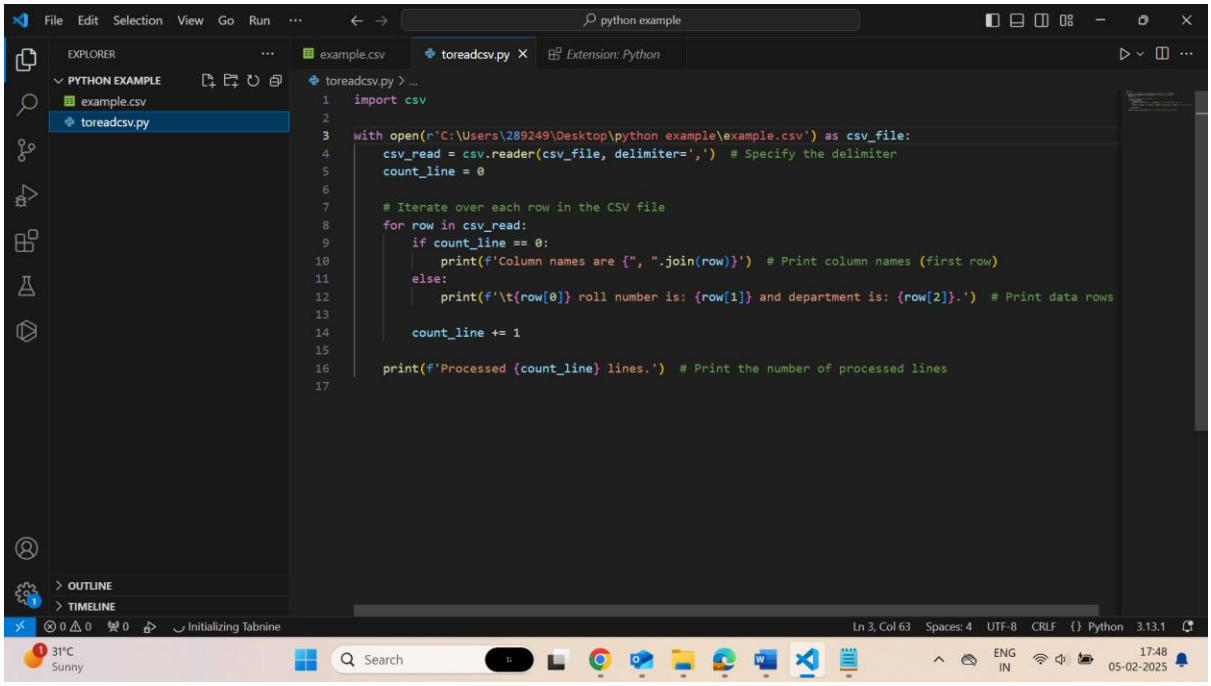
Example 7: examples of Bitwise operators in Python



How to read CSV file in Python?

Example –



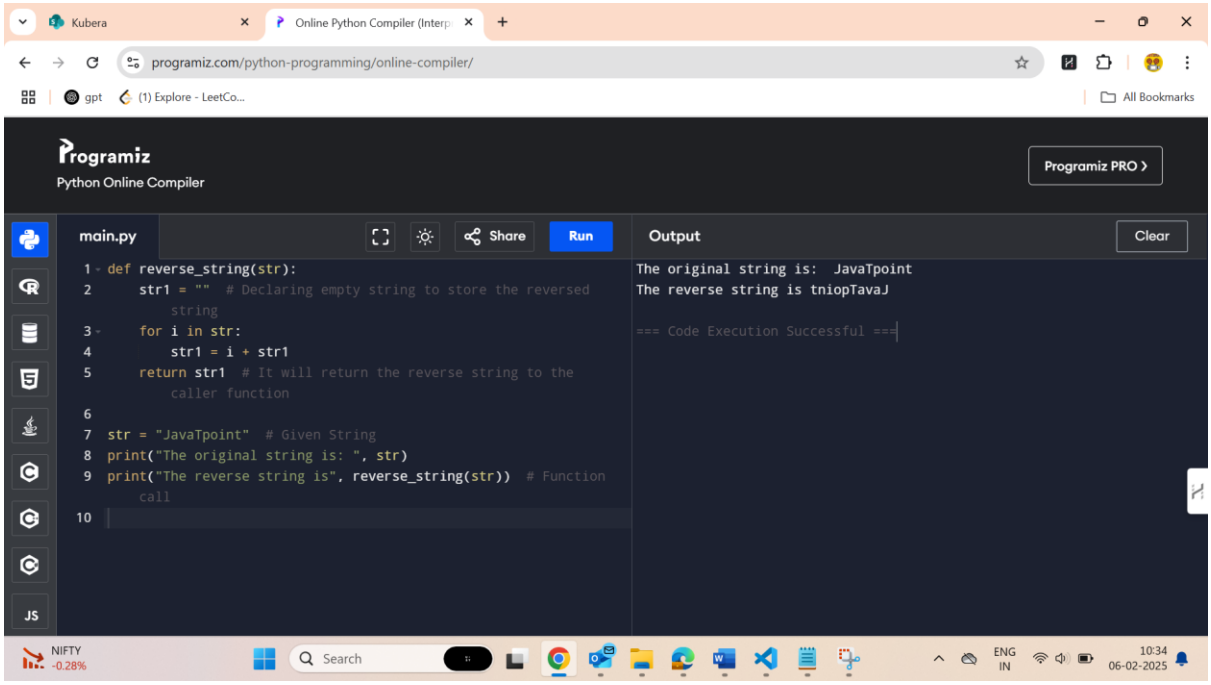


Output:

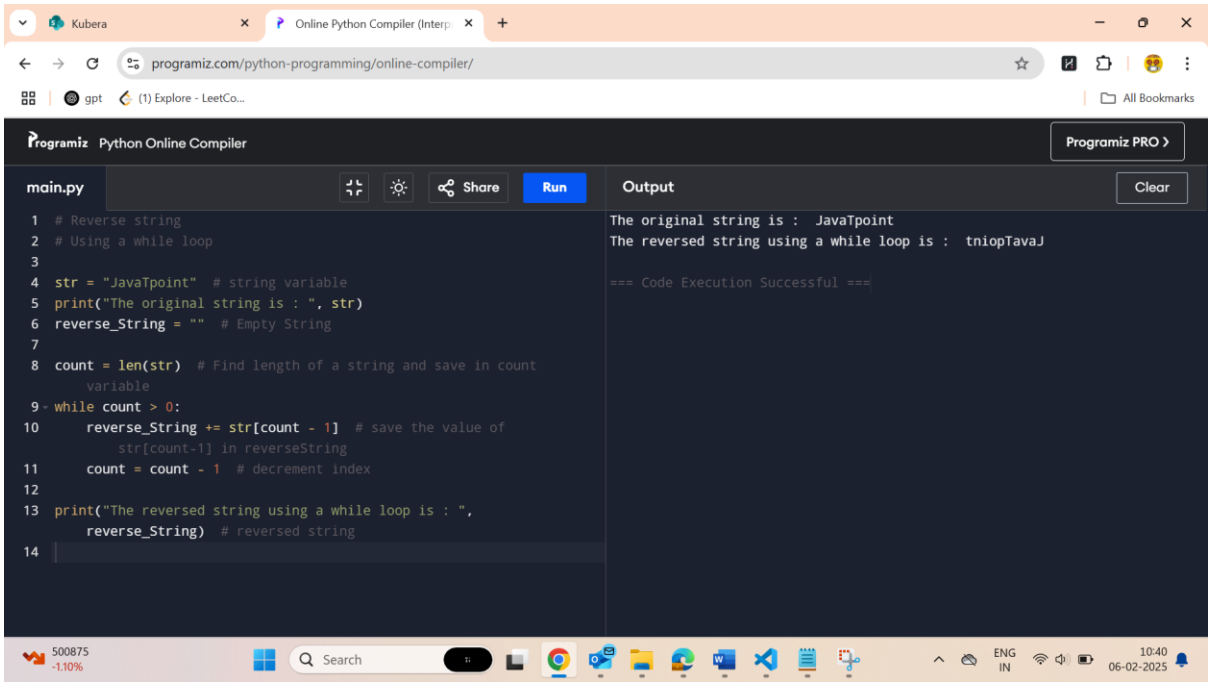
```
PS C:\Users\289249\Desktop\python example>  c:; cd 'c:\Users\289249\Desktop\p
ograms\Python\Python313\python.exe' 'c:\Users\289249\.vscode\extensions\ms-py
py\adapter/../..\debugpy\launcher' '65401' '--' 'c:\Users\289249\Desktop\pyth
Column names are name,  rollno,  Department
        Peter Parker roll number is:  009001 and department is:  Civil.
        Tony Stark roll number is:  009002 and department is:  Chemical.
Processed 3 lines.
PS C:\Users\289249\Desktop\python example>
```

Reverse A String :

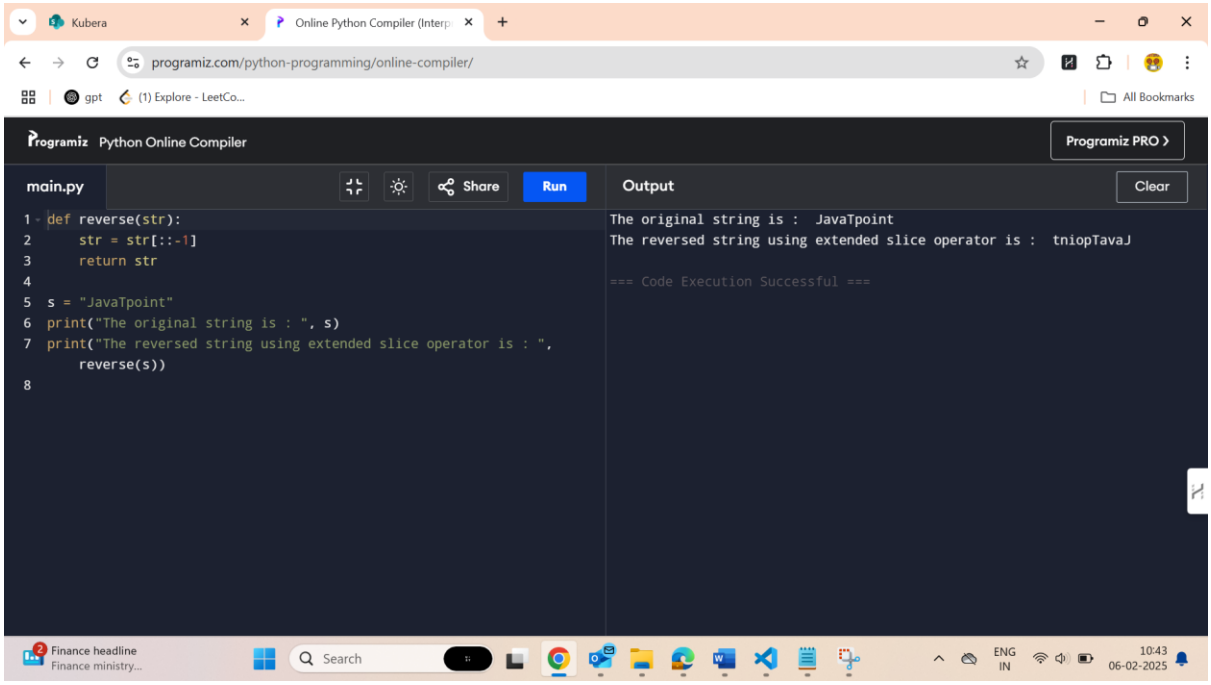
Example 1:Using For



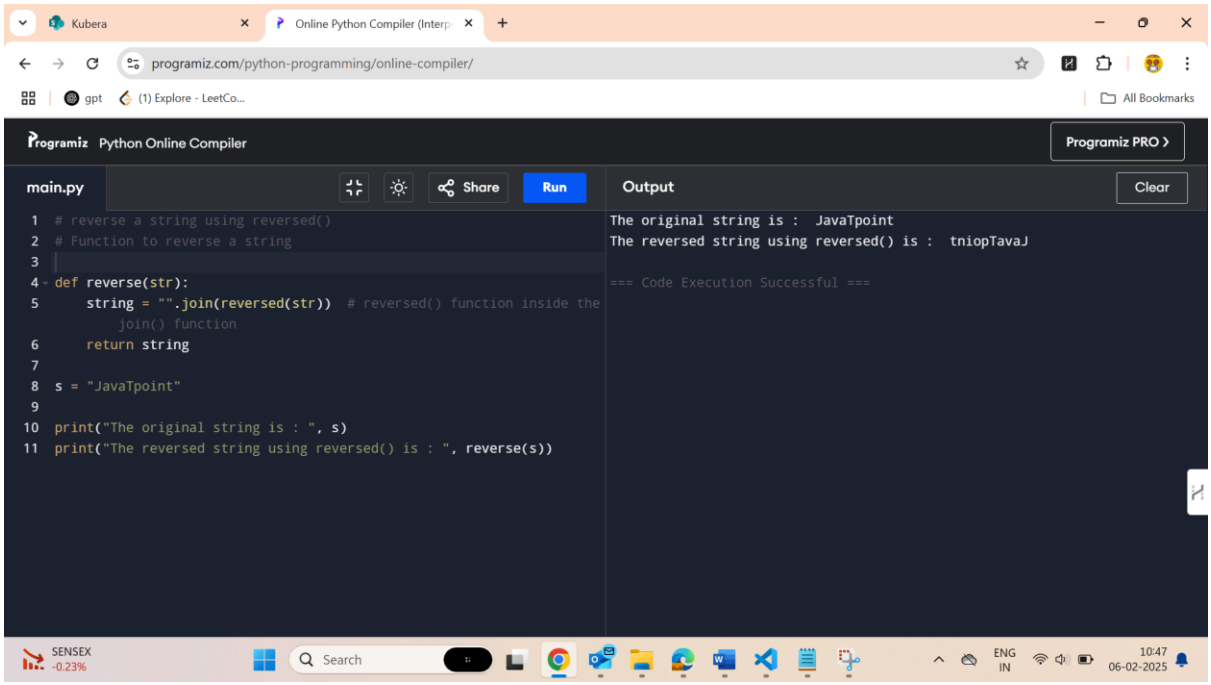
Example 2: Using While



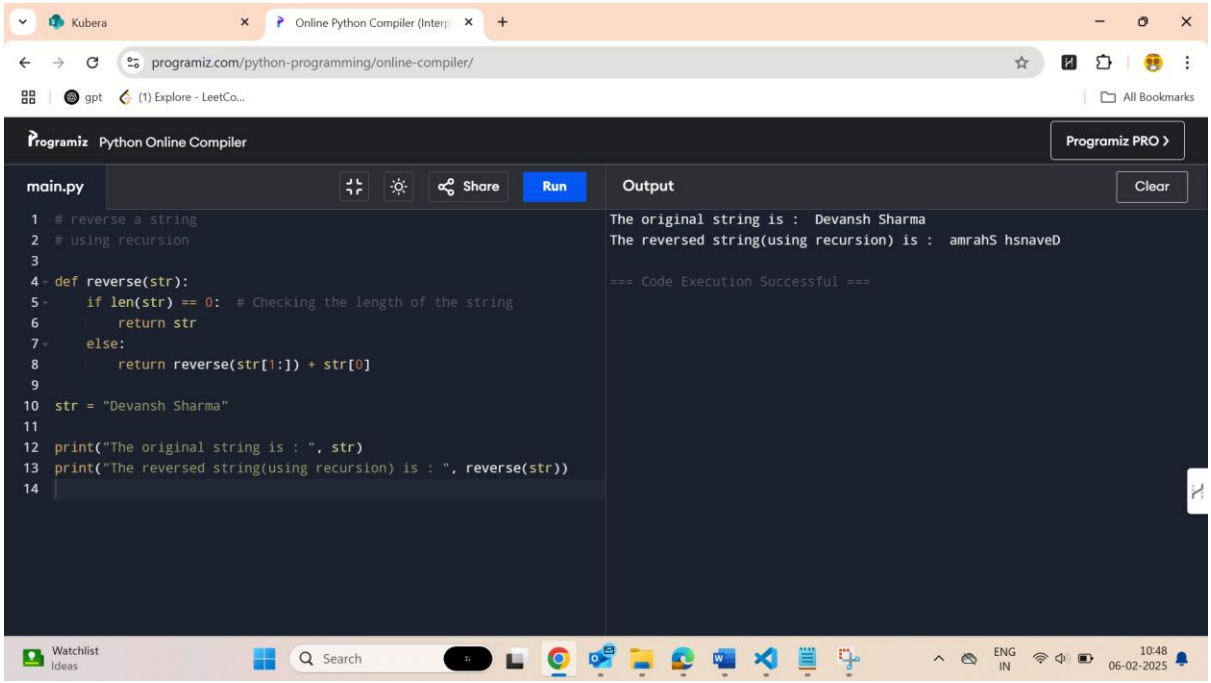
Example 3: Using the slice operator



Example 4: Using the reversed() function

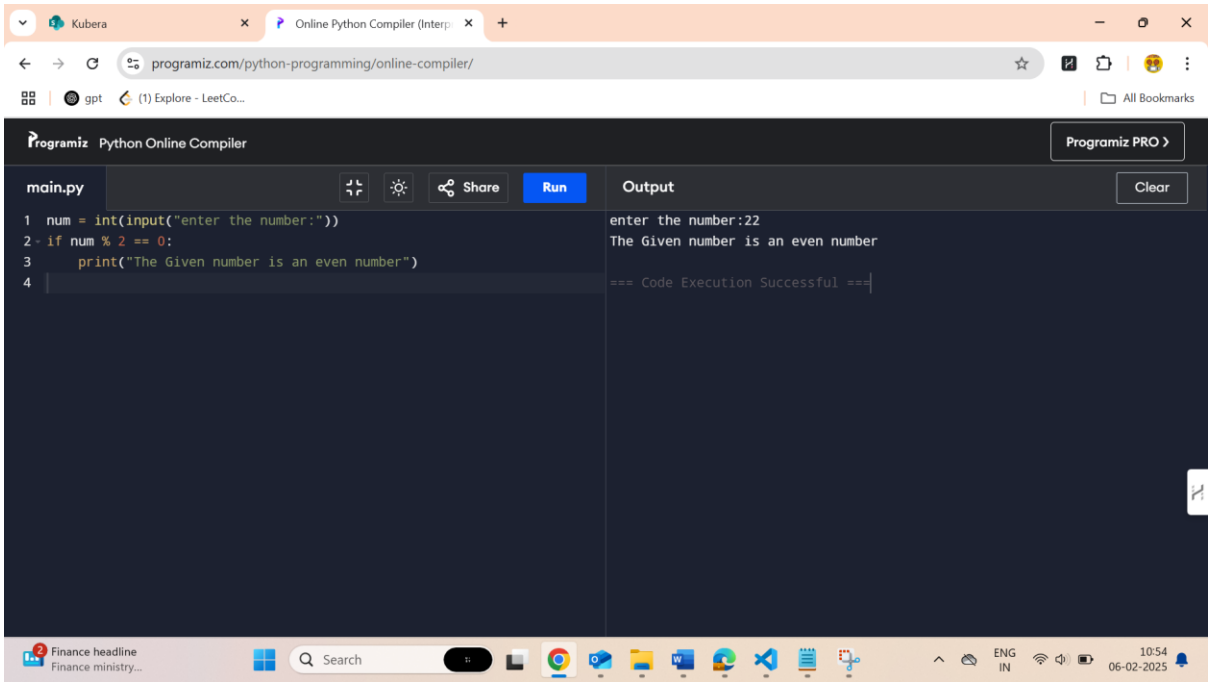


Example 5: Using recursion()

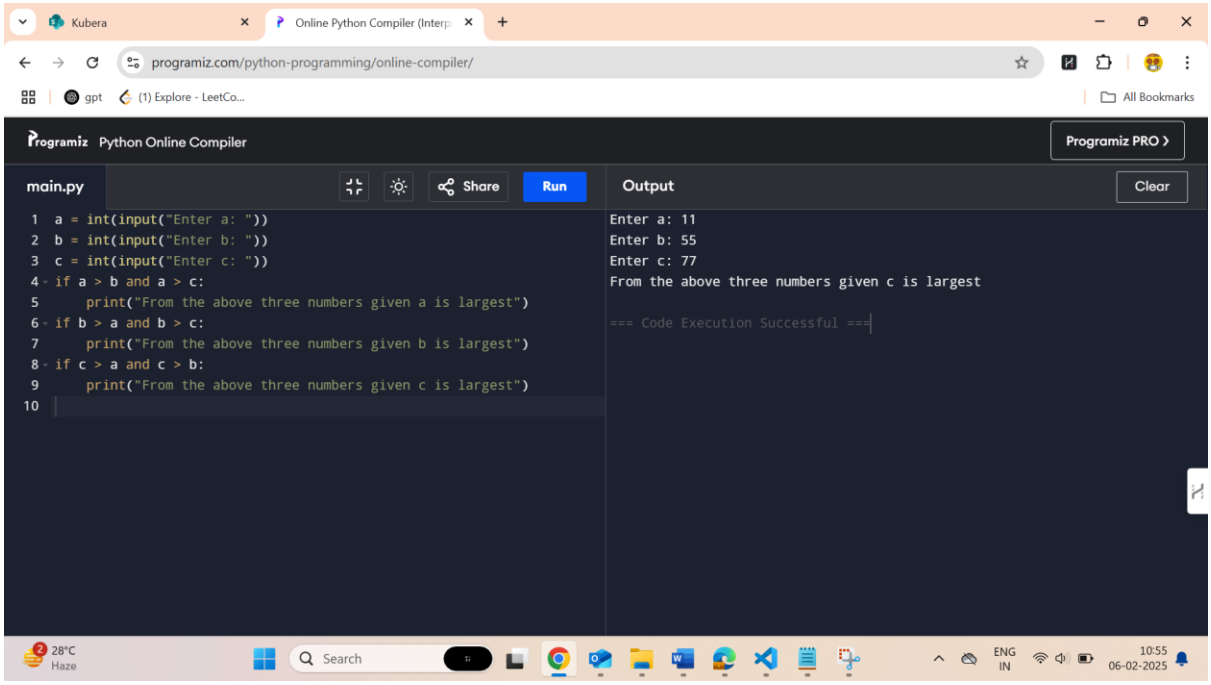


Python If else:

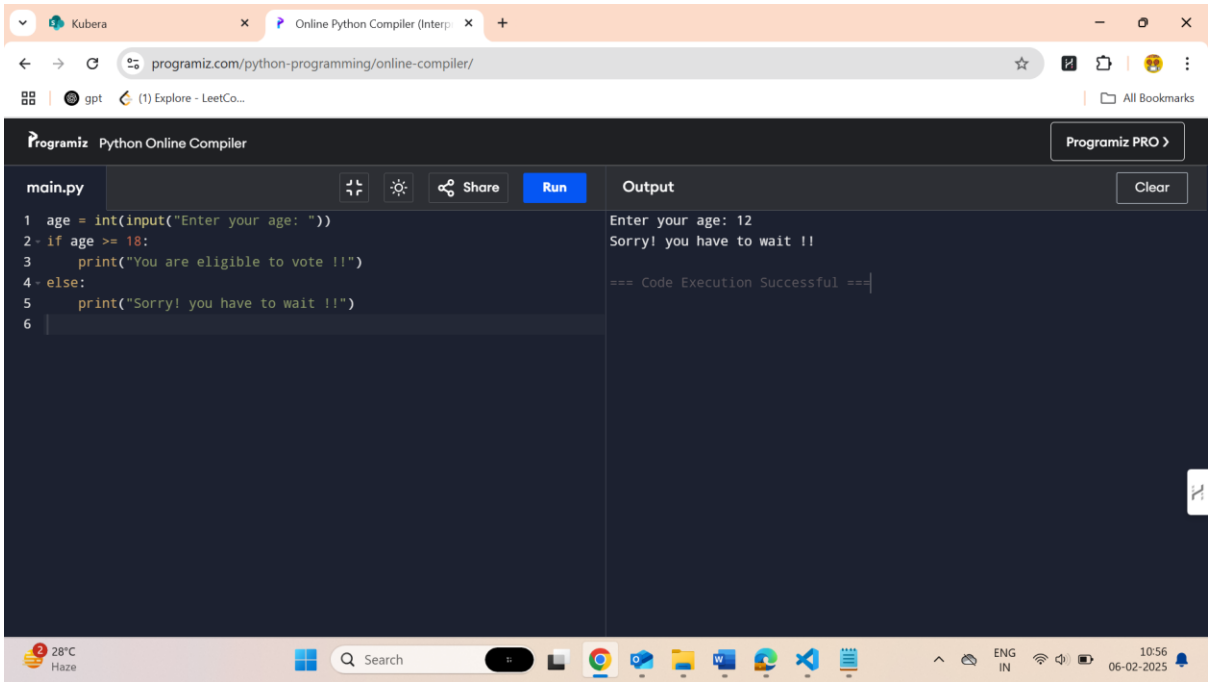
Example 1: Simple Python program to understand the if statement



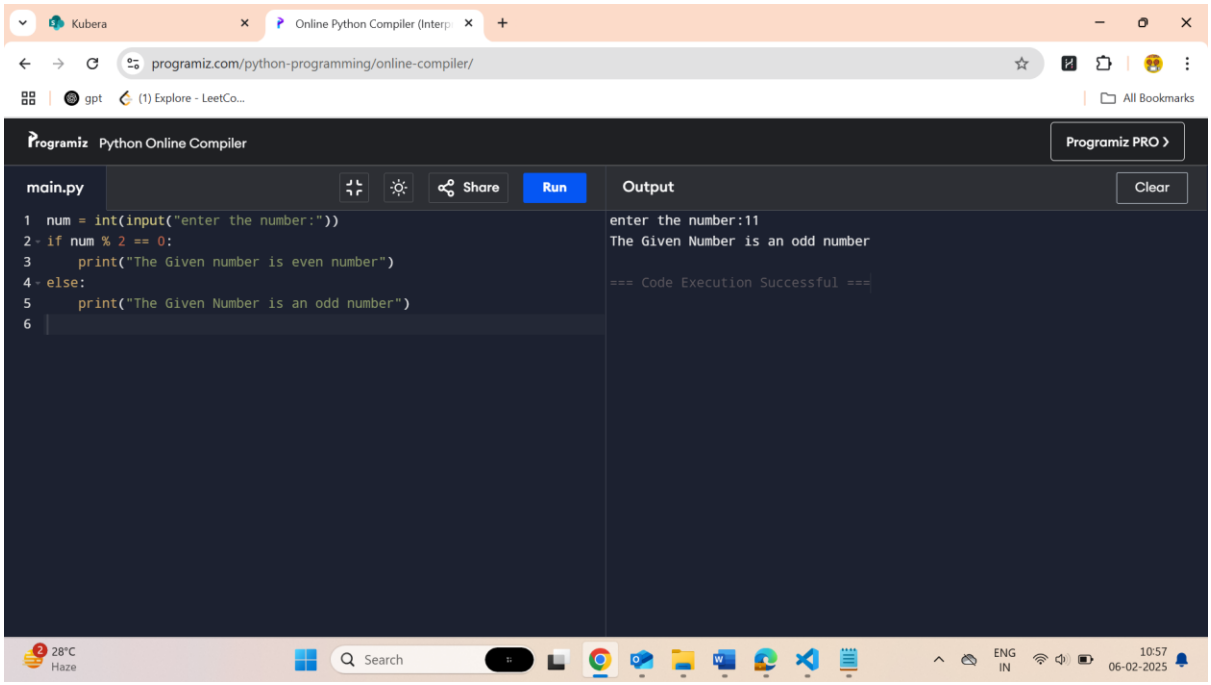
Example 2: Program to print the largest of the three numbers



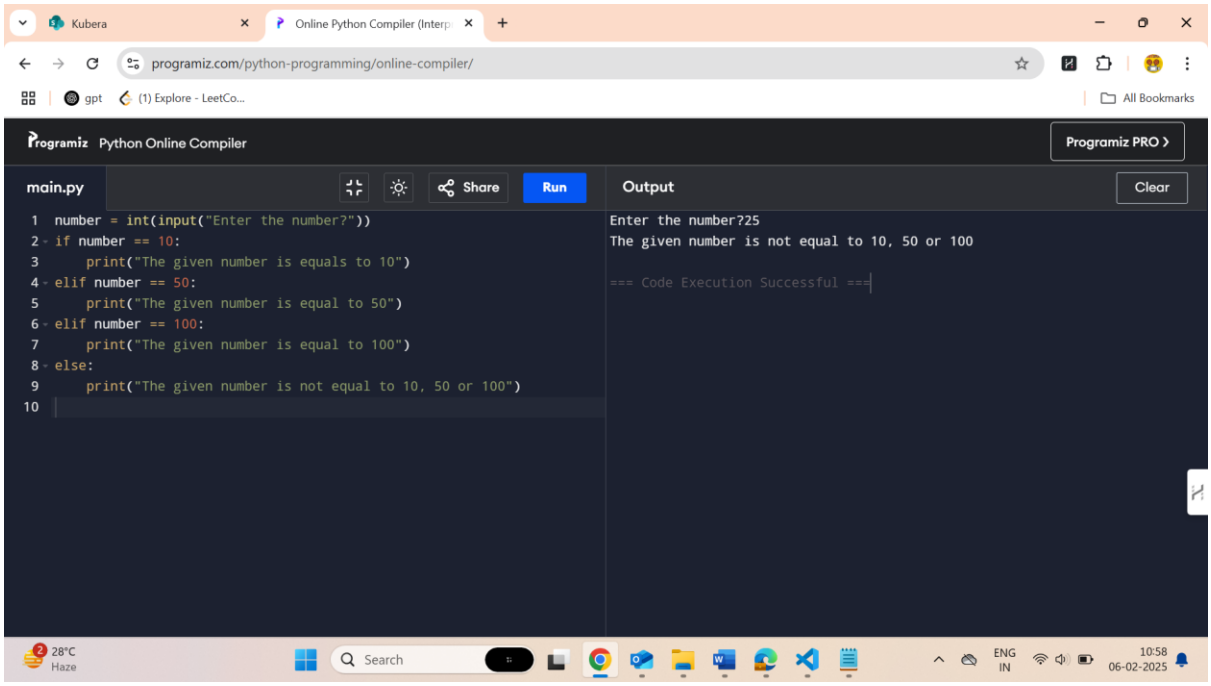
Example 3: Program to check whether a person is eligible to vote or not (if-else)



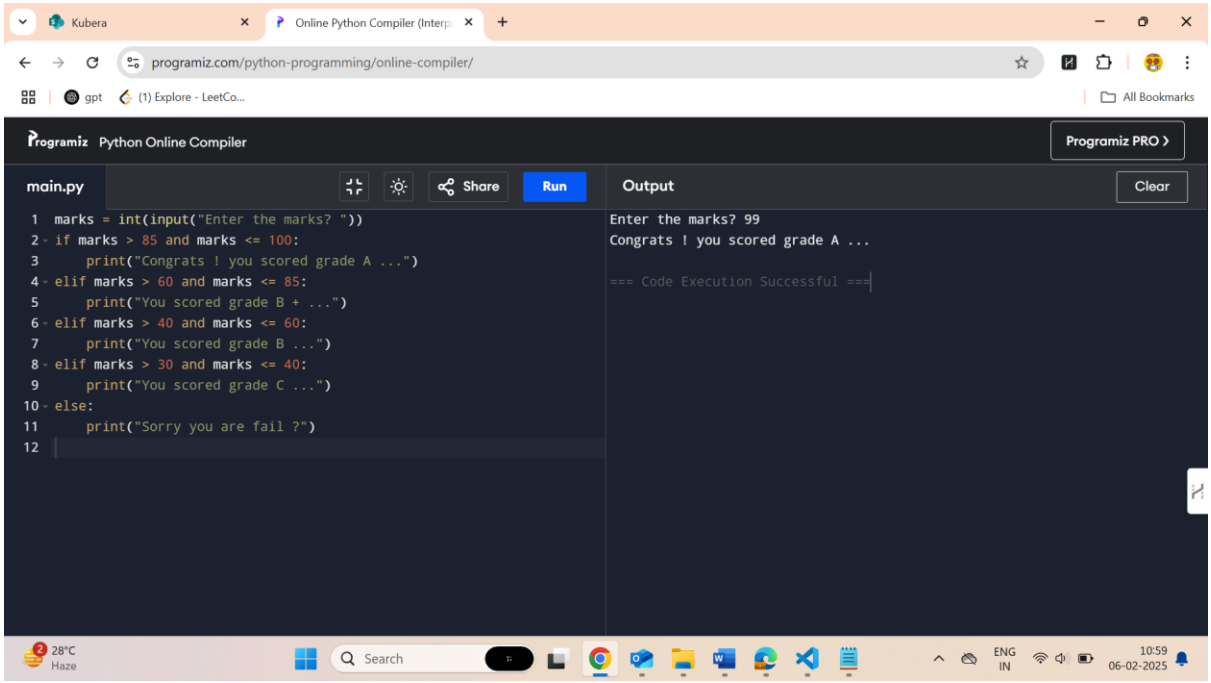
Example 4: Program to check whether a number is even or not (if-else)



Example 5: Simple Python program to understand elif statement



Example 6: Python program using elif for grading

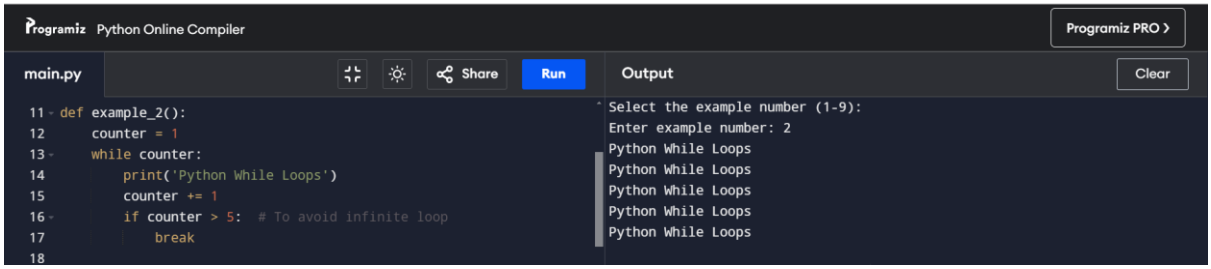


LOOPS:

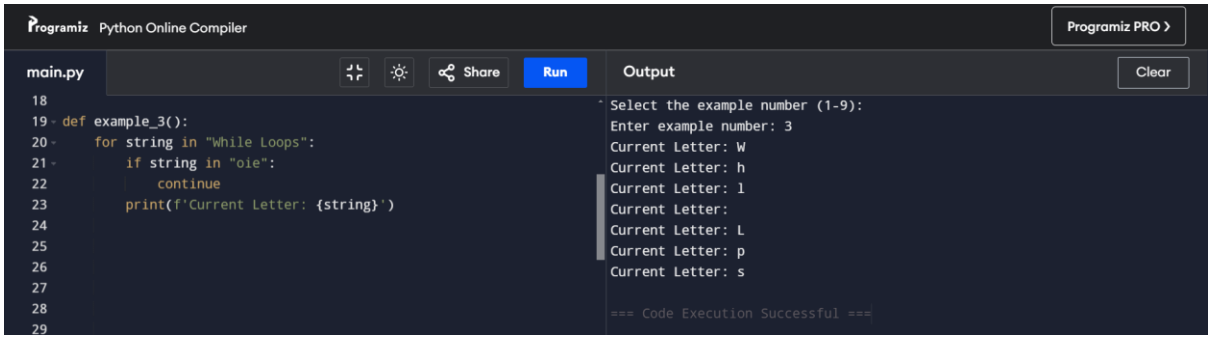
Examples:

1.

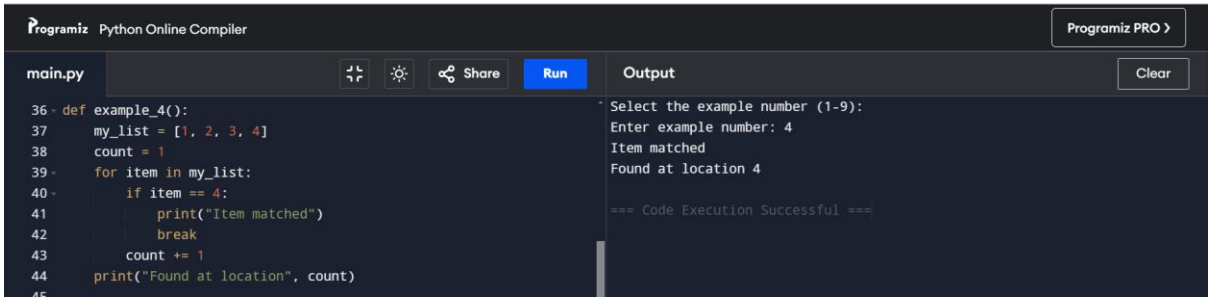
2.



3.


Programiz Python Online Compiler
Programiz PRO
main.py
Share
Run
Output
Clear
def example_3():
for string in "While Loops":
if string in "oie":
continue
print(f'Current Letter: {string}')
Select the example number (1-9):
Enter example number: 3
Current Letter: W
Current Letter: h
Current Letter: l
Current Letter:
Current Letter: L
Current Letter: p
Current Letter: s
=== Code Execution Successful ===

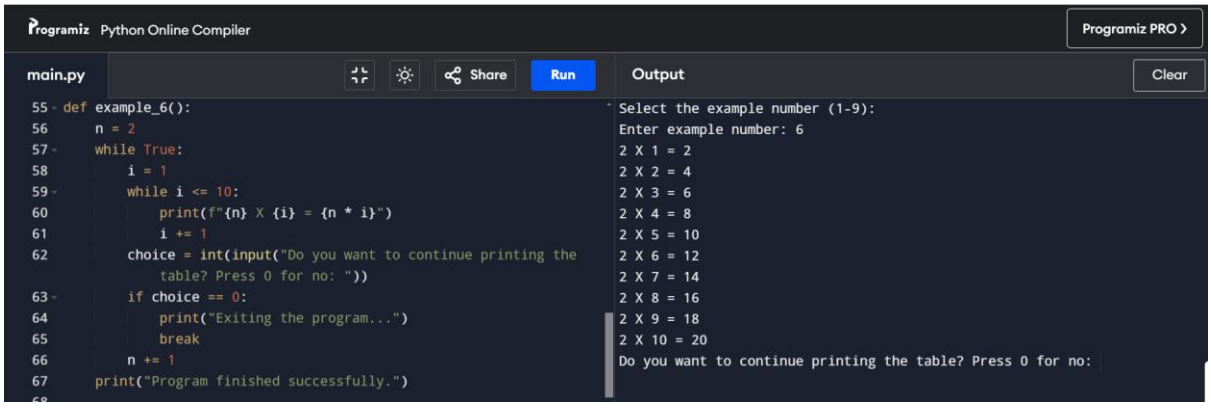
4.


Programiz Python Online Compiler
Programiz PRO
main.py
Share
Run
Output
Clear
def example_4():
my_list = [1, 2, 3, 4]
count = 1
for item in my_list:
if item == 4:
print("Item matched")
break
count += 1
print("Found at location", count)
Select the example number (1-9):
Enter example number: 4
Item matched
Found at location 4
=== Code Execution Successful ===

5.


Programiz Python Online Compiler
Programiz PRO
main.py
Share
Run
Output
Clear
def example_5():
i = 0
while 1:
print(i, " ", end="")
i += 1
if i == 10:
break
print("came out of while loop")
Select the example number (1-9):
Enter example number: 5
0 1 2 3 4 5 6 7 8 9 came out of while loop
=== Code Execution Successful ===

6.


Programiz Python Online Compiler
Programiz PRO
main.py
Share
Run
Output
Clear
def example_6():
n = 2
while True:
i = 1
while i <= 10:
print(f"{n} X {i} = {n * i}")
i += 1
choice = int(input("Do you want to continue printing the table? Press 0 for no: "))
if choice == 0:
print("Exiting the program...")
break
n += 1
print("Program finished successfully.")
Select the example number (1-9):
Enter example number: 6
2 X 1 = 2
2 X 2 = 4
2 X 3 = 6
2 X 4 = 8
2 X 5 = 10
2 X 6 = 12
2 X 7 = 14
2 X 8 = 16
2 X 9 = 18
2 X 10 = 20
Do you want to continue printing the table? Press 0 for no:

7.

Programiz Python Online Compiler — Programiz PRO >

main.py — Share — Run

```
71
72 def example_7():
73     for iterator in range(10, 21):
74         if iterator == 15:
75             continue
76         print(iterator)
77
78
79
80
81
82
```

Output — Clear

```
Select the example number (1-9):
Enter example number: 7
10
11
12
13
14
16
17
18
19
20
```

8.

Programiz Python Online Compiler — Programiz PRO >

main.py — Share — Run

```
86
87 def example_8():
88     string = "JavaTpoint"
89     iterator = 0
90
91     while iterator < len(string):
92         if string[iterator] == 'a':
93             iterator += 1
94             continue
95         print(string[iterator])
96         iterator += 1
97
98
```

Output — Clear

```
Select the example number (1-9):
Enter example number: 8
J
v
T
p
o
i
n
t

=== Code Execution Successful ===
```

9.

Programiz Python Online Compiler — Programiz PRO >

main.py — Share — Run

```
101 def example_9():
102     numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
103     sq_num = [num ** 2 for num in numbers if num % 2 == 0]
104     print(sq_num)
105
106
107
108
```

Output — Clear

```
Select the example number (1-9):
Enter example number: 9
[4, 16, 36, 64, 100]

=== Code Execution Successful ===
```

10.

Programiz Python Online Compiler — Programiz PRO >

main.py — Share — Run

```
1 def while_loop_multiple_conditions():
2     # Example 10: While Loop with Multiple Conditions
3     num1 = 9
4     num2 = 14
5     maximum_value = 4
6     counter = 0
7     # While loop with multiple conditions
8     result = []
9     while (counter < num1 or counter < num2) and not counter >=
          maximum_value:
10        result.append(f"Number of iterations: {counter}")
11        counter += 1
12    return result
```

Output — Clear

```
Select the example number (10-18):
Enter example number: 10
Number of iterations: 0
Number of iterations: 1
Number of iterations: 2
Number of iterations: 3

=== Code Execution Successful ===
```

11.

Programiz Python Online Compiler — Programiz PRO

main.py — Share — Run — Output — Clear

```python
def single_statement_while():
    # Example 11: Single Statement While Loop
    counter = 1
    # Single statement while loop
    result = []
    while counter:
        result.append('Python While Loops')
        counter += 1
        if counter > 5:  # Preventing infinite loop
            break
    return result
```

```
Select the example number (10-18):
Enter example number: 11
Python While Loops
Python While Loops
Python While Loops
Python While Loops
Python While Loops

=== Code Execution Successful ===
```

12.

Programiz Python Online Compiler — Programiz PRO

main.py — Share — Run — Output — Clear

```python
def continue_statement_example():
    # Example 12: Continue Statement Example (for loop)
    result = []
    for string in "While Loops":
        if string in "oie":
            continue
        result.append(f'Current Letter: {string}')
    return result
```

```
Select the example number (10-18):
Enter example number: 12
Current Letter: W
Current Letter: h
Current Letter: l
Current Letter:
Current Letter: L
Current Letter: p
Current Letter: s
```

13.

Programiz Python Online Compiler — Programiz PRO

main.py — Share — Run — Output — Clear

```python
def break_statement_example():
    # Example 13: Break Statement Example (for loop)
    my_list = [1, 2, 3, 4]
    count = 1
    result = []
    for item in my_list:
        if item == 4:
            result.append("Item matched")
            break
        count += 1
    result.append("Found at location " + str(count))
    return result
```

```
Select the example number (10-18):
Enter example number: 13
Item matched
Found at location 4

=== Code Execution Successful ===
```

14.

Programiz Python Online Compiler — Programiz PRO

main.py — Share — Run — Output — Clear

```python
def break_statement_while():
    # Example 14: Break Statement Example (while loop)
    i = 0
    result = []
    while 1:
        result.append(f"{i} ")
        i += 1
        if i == 10:
            break
    result.append("came out of while loop")
    return result
```

```
Select the example number (10-18):
Enter example number: 14
0
1
2
3
4
5
6
7
8
9
came out of while loop
```

15.

Programiz Python Online Compiler — Programiz PRO

main.py — Share — Run

```
def break_statement_nested_loops():
    # Example 15: Break Statement Example (nested loops)
    n = 2
    result = []
    while True:
        i = 1
        while i <= 10:
            result.append(f"{n} X {i} = {n * i}")
            i += 1
        choice = int(input("Do you want to continue printing the
            table? Press 0 for no: "))
        if choice == 0:
            result.append("Exiting the program...")
            break
        n += 1
    result.append("Program finished successfully.")
    return result
```

Output — Clear

```
Select the example number (10-18):
Enter example number: 15
Do you want to continue printing the table? Press 0 for no: 0
2 X 1 = 2
2 X 2 = 4
2 X 3 = 6
2 X 4 = 8
2 X 5 = 10
2 X 6 = 12
2 X 7 = 14
2 X 8 = 16
2 X 9 = 18
2 X 10 = 20
Exiting the program...
Program finished successfully.

=== Code Execution Successful ===
```

16.

Programiz Python Online Compiler — Programiz PRO

main.py — Share — Run

```
def continue_statement_for_loop():
    # Example 16: Continue Statement Example (for loop with 10-20)
    result = []
    for iterator in range(10, 21):
        if iterator == 15:
            continue
        result.append(str(iterator))
    return result
```

Output — Clear

```
Select the example number (10-18):
Enter example number: 16
10
11
12
13
14
16
17
18
19
20
```

17.

Programiz Python Online Compiler — Programiz PRO

main.py — Share — Run

```
def continue_statement_while_loop():
    # Example 17: Continue Statement Example (while loop with
        string)
    string = "JavaTpoint"
    iterator = 0
    result = []
    while iterator < len(string):
        if string[iterator] == 'a':
            iterator += 1
            continue
        result.append(string[iterator])
        iterator += 1
    return result
```

Output — Clear

```
Select the example number (10-18):
Enter example number: 17
J
v
T
p
o
i
n
t

=== Code Execution Successful ===
```

18.

Programiz Python Online Compiler — Programiz PRO

main.py — Share — Run

```
def continue_statement_list_comprehension():
    # Example 18: Continue Statement Example (list comprehension)
    numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
    sq_num = [num ** 2 for num in numbers if num % 2 == 0]
    return sq_num
```

Output — Clear

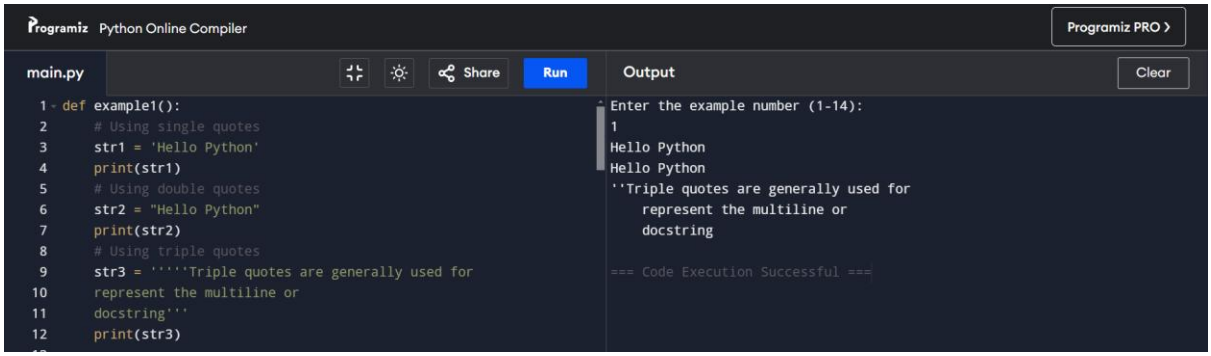
```
Select the example number (10-18):
Enter example number: 18
4
16
36
64
100

=== Code Execution Successful ===
```

Python Strings and Tuple

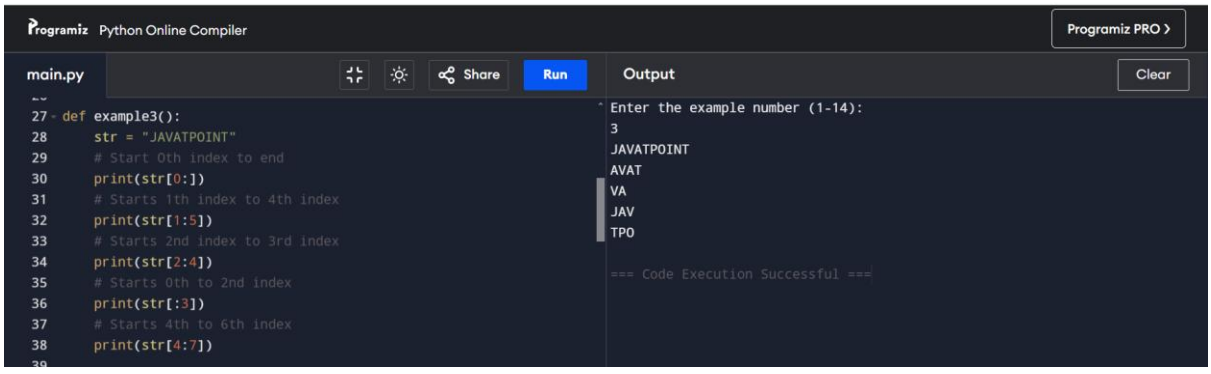
examples:

1.

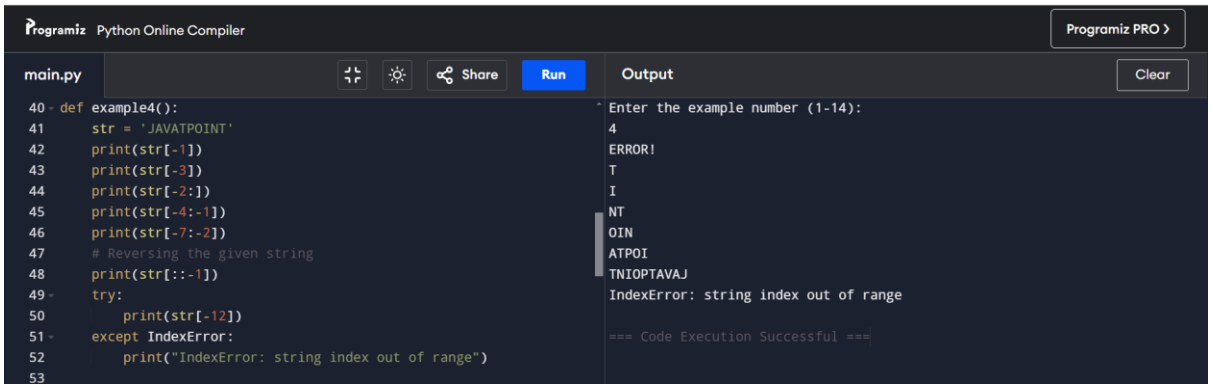


2.

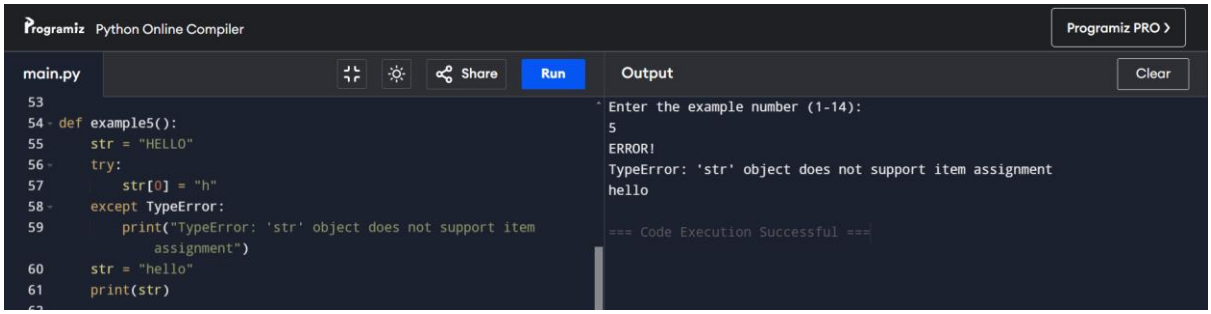
3.



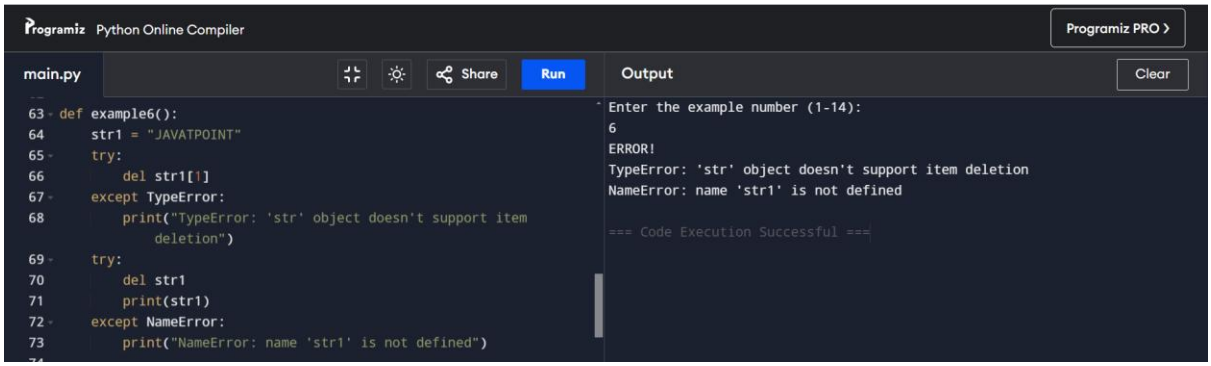
4.


Programiz Python Online Compiler
Programiz PRO
main.py
Share
Run
Output
Clear
def example4():
str = 'JAVATPOINT'
print(str[-1])
print(str[-3])
print(str[-2:])
print(str[-4:-1])
print(str[-7:-2])
# Reversing the given string
print(str[::-1])
try:
print(str[-12])
except IndexError:
print("IndexError: string index out of range")
Enter the example number (1-14):
4
ERROR!
T
I
NT
OIN
ATPOI
TNIOPTAVAJ
IndexError: string index out of range
=== Code Execution Successful ===

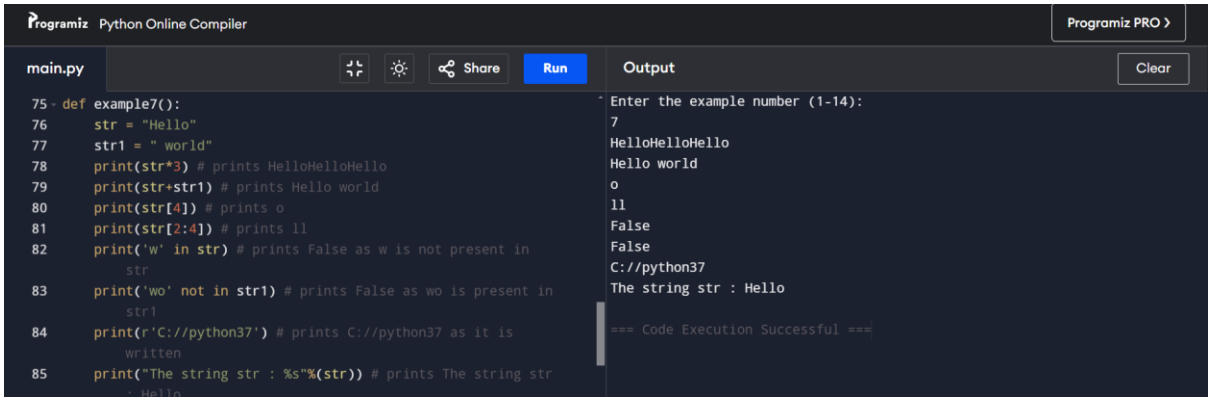
5.


Programiz Python Online Compiler
Programiz PRO
main.py
Share
Run
Output
Clear
def example5():
str = "HELLO"
try:
str[0] = "h"
except TypeError:
print("TypeError: 'str' object does not support item assignment")
str = "hello"
print(str)
Enter the example number (1-14):
5
ERROR!
TypeError: 'str' object does not support item assignment
hello
=== Code Execution Successful ===

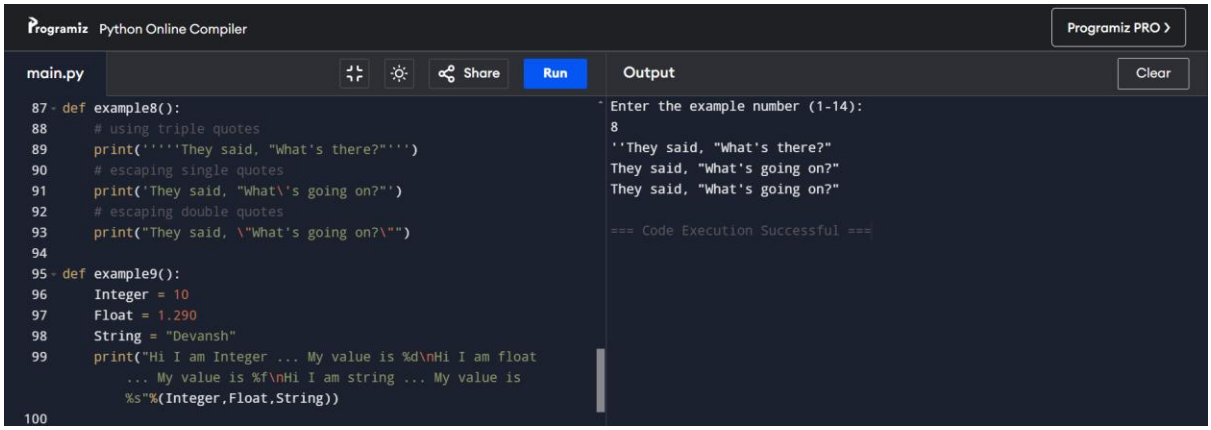
6.


Programiz Python Online Compiler
Programiz PRO
main.py
Share
Run
Output
Clear
def example6():
str1 = "JAVATPOINT"
try:
del str1[1]
except TypeError:
print("TypeError: 'str' object doesn't support item deletion")
try:
del str1
print(str1)
except NameError:
print("NameError: name 'str1' is not defined")
Enter the example number (1-14):
6
ERROR!
TypeError: 'str' object doesn't support item deletion
NameError: name 'str1' is not defined
=== Code Execution Successful ===

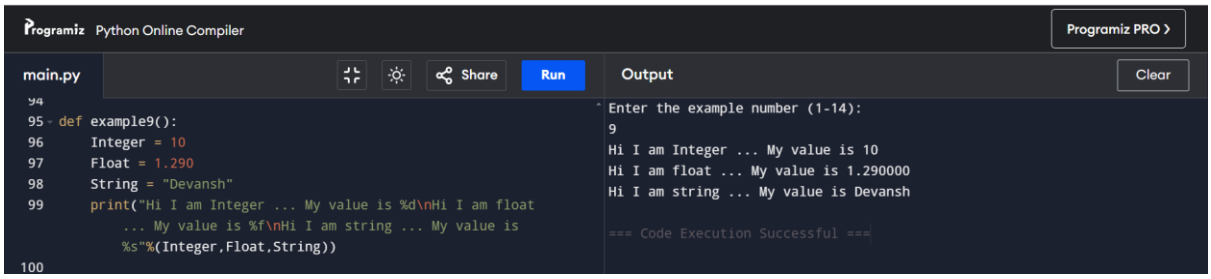
7.


Programiz Python Online Compiler
Programiz PRO
main.py
Share
Run
Output
Clear
def example7():
str = "Hello"
str1 = " world"
print(str*3) # prints HelloHelloHello
print(str+str1) # prints Hello world
print(str[4]) # prints o
print(str[2:4]) # prints ll
print('w' in str) # prints False as w is not present in str
print('wo' not in str1) # prints False as wo is present in str1
print(r'C://python37') # prints C://python37 as it is written
print("The string str : %s"%(str)) # prints The string str : Hello
Enter the example number (1-14):
7
HelloHelloHello
Hello world
o
ll
False
False
C://python37
The string str : Hello
=== Code Execution Successful ===

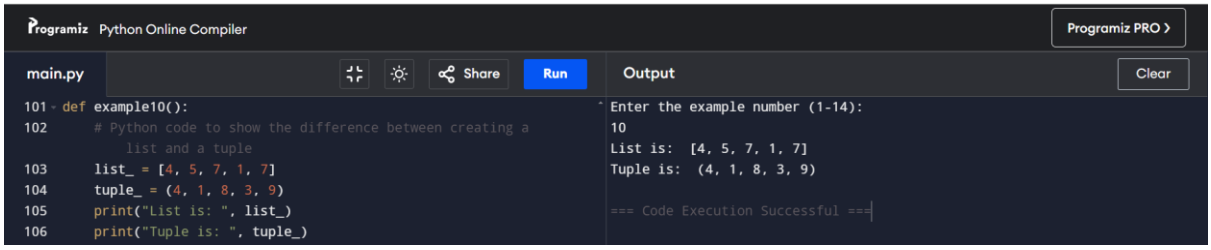
8.


Programiz Python Online Compiler
Programiz PRO
main.py
Share
Run
Output
Clear
def example8():
# using triple quotes
print('''They said, "What's there?"''')
# escaping single quotes
print('They said, "What\'s going on?"')
# escaping double quotes
print("They said, \"What's going on?\"")
def example9():
Integer = 10
Float = 1.290
String = "Devansh"
print("Hi I am Integer ... My value is %d\nHi I am float ... My value is %f\nHi I am string ... My value is %s"%(Integer,Float,String))
Enter the example number (1-14):
8
''They said, "What's there?"
They said, "What's going on?"
They said, "What's going on?"
=== Code Execution Successful ===

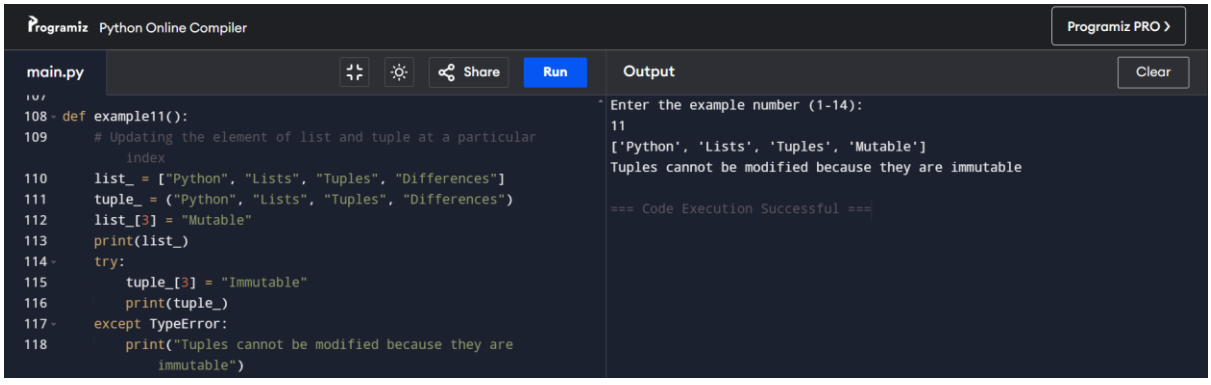
9.


Programiz Python Online Compiler
Programiz PRO
main.py
Share
Run
Output
Clear
def example9():
Integer = 10
Float = 1.290
String = "Devansh"
print("Hi I am Integer ... My value is %d\nHi I am float ... My value is %f\nHi I am string ... My value is %s"%(Integer,Float,String))
Enter the example number (1-14):
9
Hi I am Integer ... My value is 10
Hi I am float ... My value is 1.290000
Hi I am string ... My value is Devansh
=== Code Execution Successful ===

10.


Programiz Python Online Compiler
Programiz PRO
main.py
Share
Run
Output
Clear
def example10():
# Python code to show the difference between creating a list and a tuple
list_ = [4, 5, 7, 1, 7]
tuple_ = (4, 1, 8, 3, 9)
print("List is: ", list_)
print("Tuple is: ", tuple_)
Enter the example number (1-14):
10
List is: [4, 5, 7, 1, 7]
Tuple is: (4, 1, 8, 3, 9)
=== Code Execution Successful ===

11.


Programiz Python Online Compiler
Programiz PRO
main.py
Share
Run
Output
Clear
def example11():
# Updating the element of list and tuple at a particular index
list_ = ["Python", "Lists", "Tuples", "Differences"]
tuple_ = ("Python", "Lists", "Tuples", "Differences")
list_[3] = "Mutable"
print(list_)
try:
tuple_[3] = "Immutable"
print(tuple_)
except TypeError:
print("Tuples cannot be modified because they are immutable")
Enter the example number (1-14):
11
['Python', 'Lists', 'Tuples', 'Mutable']
Tuples cannot be modified because they are immutable
=== Code Execution Successful ===

12.


Programiz Python Online Compiler
Programiz PRO
main.py
Share
Run
Output
Clear
def example12():
# Code to show the difference in the size of a list and a tuple
list_ = ["Python", "Lists", "Tuples", "Differences"]
tuple_ = ("Python", "Lists", "Tuples", "Differences")
# printing sizes
print("Size of tuple: ", tuple_.__sizeof__())
print("Size of list: ", list_.__sizeof__())
Enter the example number (1-14):
12
Size of tuple: 56
Size of list: 72
=== Code Execution Successful ===