	Name: Mohmadhusen Ahmadbhai Khimani **Enrollment No:** 22FOTCA11071 **Roll No:** 14 **Div:** 6-BCA-B **Subject:** Python Programming **Subject Code:** BCA619

—---

Practical Preparation for Exam TCIE-2

—---

Questions:	1. **Write a Python program to print "Hello, World!"** 2. **Write a program to take user input for their name and greet them.** 3. **Write a Python program to swap two numbers without using a third variable.** 4. **Write a program to check whether a number is even or odd.** 5. **Write a program to find the largest**

among three numbers.

Control Statements & Loops

6. **Write a program to print numbers from 1 to 10 using a for loop.**
7. **Write a program to find the sum of first n natural numbers using a while loop.**
8. **Write a program to print the Fibonacci series up to n terms.**
9. **Write a program to check if a given number is a prime number.**
10. **Write a program to generate the multiplication table of a given number.**

Functions & Recursion

11. **Write a function to find the factorial of a number using recursion.**
12. **Write a function to check if a string is a palindrome.**
13. **Write a function to calculate the sum of digits of a number.**
14. **Write a function to reverse a string without using built-in functions.**
15. **Write a function to find the**

greatest common divisor (GCD) of two numbers.

Lists & Tuples

16. **Write a program to find the largest element in a list.**
17. **Write a program to remove duplicate elements from a list.**
18. **Write a program to find the second largest element in a list.**
19. **Write a program to count the occurrences of an element in a list.**
20. **Write a program to find the sum of all elements in a tuple.**

Dictionaries & Sets

21. **Write a program to create a dictionary and display its keys and values.**
22. **Write a program to merge two dictionaries into one.**
23. **Write a program to count occurrences of words in a given sentence.**
24. **Write a program to find the intersection of two sets.**
25. **Write a program to check if a**

key exists in a dictionary.

String Handling

31. **Write a program to count the number of vowels in a given string.**
32. **Write a program to remove all special characters from a string.**
33. **Write a program to find the longest word in a given sentence.**
34. **Write a program to check if two strings are anagrams of each other.**
35. **Write a program to replace a substring within a given string.**

Object-Oriented Programming (OOP)

36. **Write a Python class with a constructor and a method to display student details.**
37. **Write a program to implement single inheritance in Python.**
38. **Write a program to implement method overloading using default arguments.**
39. **Write a program to implement**

	operator overloading for the `+` operator. **40. Write a program to create a class with private attributes and access them using getter and setter methods.** **Exception Handling** **41. Write a program to demonstrate the use of `try`, `except`, and `finally`.** **42. Write a program to handle division by zero exception.** **43. Write a program to raise a custom exception if a given number is negative.** **44. Write a program to handle multiple exceptions in Python.** **45. Write a program that reads a file and handles the exception if the file is not found.**

Code:	# Hello, World!
	print("Hello, World!")
	
	# Taking user input and greeting
	name = input("Enter your name: ")
	print(f"Hello, {name}!")

```
# Swapping two numbers without a third variable
a, b = 5, 10
a, b = b, a
print("After swapping:", a, b)

# Check if a number is even or odd
num = int(input("Enter a number: "))
print("Even" if num % 2 == 0 else "Odd")

# Find the largest among three numbers
a, b, c = 3, 7, 5
print("Largest:", max(a, b, c))

# Print numbers from 1 to 10 using a for loop
for i in range(1, 11):
    print(i)

# Sum of first n natural numbers using a while loop
n = int(input("Enter n: "))
sum_n = 0
while n > 0:
    sum_n += n
    n -= 1
print("Sum:", sum_n)

# Fibonacci series up to n terms
```

```
def fibonacci(n):
    a, b = 0, 1
    for _ in range(n):
        print(a, end=" ")
        a, b = b, a + b

fibonacci(int(input("Enter n: ")))

# Check if a number is prime
num = int(input("Enter a number: "))
if num > 1 and all(num % i != 0 for i in range(2, int(num**0.5) + 1)):
    print("Prime")
else:
    print("Not Prime")

# Multiplication table of a given number
n = int(input("Enter a number: "))
for i in range(1, 11):
    print(f"{n} x {i} = {n*i}")

# Factorial using recursion
def factorial(n):
    return 1 if n == 0 else n * factorial(n - 1)
print(factorial(int(input("Enter n: "))))

# Check if a string is palindrome
def is_palindrome(s):
    return s == s[::-1]
print(is_palindrome(input("Enter a string: ")))
```

```
# Sum of digits of a number
def sum_digits(n):
    return sum(int(digit) for digit in str(n))
print(sum_digits(int(input("Enter a number:
"))))

# Reverse a string without built-in functions
def reverse_string(s):
    return "".join(reversed(s))
print(reverse_string(input("Enter a string:
")))

# Find GCD of two numbers
def gcd(a, b):
    while b:
        a, b = b, a % b
    return a
print(gcd(int(input("Enter first number: ")),
int(input("Enter second number: "))))

# Find the largest element in a list
lst = [3, 7, 2, 9, 5]
print(max(lst))

# Remove duplicates from a list
print(list(set(lst)))

# Find second largest element in a list
print(sorted(set(lst))[-2])
```

```
# Count occurrences of an element in a list
print(lst.count(int(input("Enter element to count: "))))

# Sum of all elements in a tuple
tup = (1, 2, 3, 4, 5)
print(sum(tup))

# Create a dictionary and display keys & values
d = {"name": "Alice", "age": 25}
print(d.keys(), d.values())

# Merge two dictionaries
d1 = {"a": 1, "b": 2}
d2 = {"c": 3, "d": 4}
d1.update(d2)
print(d1)

# Count occurrences of words in a sentence
sentence = input("Enter a sentence: ").split()
print({word: sentence.count(word) for word in set(sentence)})

# Intersection of two sets
s1, s2 = {1, 2, 3}, {2, 3, 4}
print(s1 & s2)

# Check if a key exists in a dictionary
```

```
print("age" in d)

# Count vowels in a string
s = input("Enter a string: ")
print(sum(1 for ch in s if ch.lower() in
"aeiou"))

# Remove special characters from a string
import re
s = input("Enter a string: ")
print(re.sub(r'[^a-zA-Z0-9 ]', '', s))

# Find longest word in a sentence
print(max(input("Enter a sentence: ").split(),
key=len))

# Check if two strings are anagrams
s1, s2 = input("Enter first string: "),
input("Enter second string: ")
print(sorted(s1) == sorted(s2))

# Replace a substring
s = input("Enter string: ")
old, new = input("Old substring: "),
input("New substring: ")
print(s.replace(old, new))

# Class with constructor and method
class Student:
    def __init__(self, name, age):
        self.name = name
        self.age = age
    def display(self):
        print(f"Name: {self.name}, Age:
{self.age}")
Student("Alice", 20).display()

# Single inheritance
class A:
    def show(self):
        print("A")
class B(A):
    pass
B().show()

# Method overloading using default
arguments
def add(a, b=0):
    return a + b
print(add(5), add(5, 3))

# Operator overloading
class Num:
    def __init__(self, val):
        self.val = val
    def __add__(self, other):
        return Num(self.val + other.val)
n1, n2 = Num(5), Num(3)
print((n1 + n2).val)
```

```
# Private attributes with getter and setter
class Person:
    def __init__(self, name):
        self.__name = name
    def get_name(self):
        return self.__name
    def set_name(self, name):
        self.__name = name
p = Person("Alice")
p.set_name("Bob")
print(p.get_name())

# Exception handling with try, except, finally
try:
    x = int(input("Enter a number: "))
    print(10 / x)
except ZeroDivisionError:
    print("Cannot divide by zero!")
finally:
    print("Done!")

# Custom exception for negative numbers
class NegativeError(Exception):
    pass
num = int(input("Enter a number: "))
if num < 0:
    raise NegativeError("Negative number
not allowed!")

# Handling multiple exceptions
```

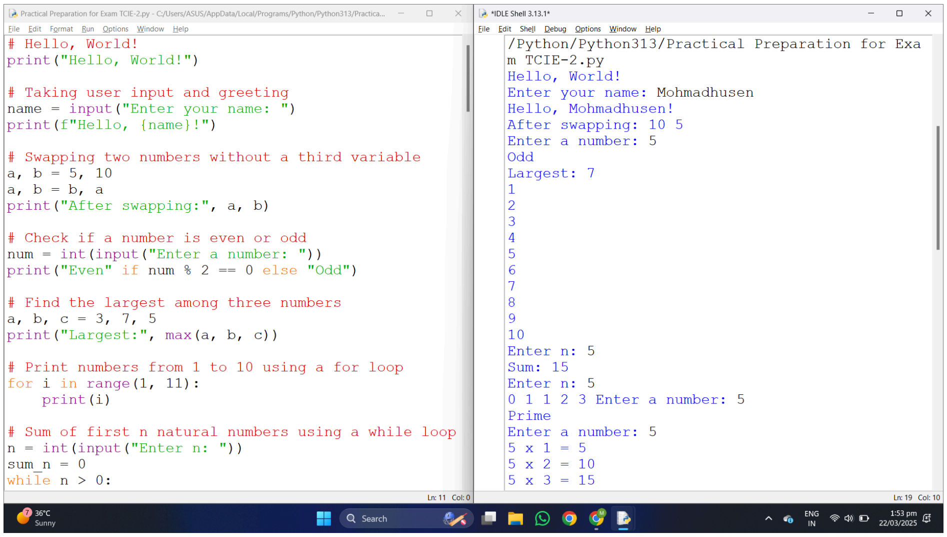
```
try:
    print(10 / int(input("Enter a number: ")))
except (ZeroDivisionError, ValueError):
    print("Invalid input!")

# File handling with exception handling
try:
    with open("file.txt") as f:
        print(f.read())
except FileNotFoundError:
    print("File not found!")
```

Output:

```
Hello, World!
Enter your name: Mohmadhusen
Hello, Mohmadhusen!
After swapping: 10 5
Enter a number: 5
Odd
Largest: 7
1
2
3
4
5
6
7
8
9
10
```

```
Enter n: 5
Sum: 15
Enter n: 5
0 1 1 2 3 Enter a number: 5
Prime
Enter a number: 5
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
5 x 6 = 30
5 x 7 = 35
5 x 8 = 40
5 x 9 = 45
5 x 10 = 50
Enter n: 5
120
Enter a string: 5
True
Enter a number: 5
5
Enter a string: 5
5
Enter first number: 5
Enter second number: 5
5
9
[2, 3, 5, 7, 9]
7
```

```
Enter element to count: 5
1
15
dict_keys(['name', 'age'])
dict_values(['Alice', 25])
{'a': 1, 'b': 2, 'c': 3, 'd': 4}
Enter a sentence: 5
{'5': 1}
{2, 3}
True
Enter a string: 5
0
Enter a string: 5
5
Enter a sentence: 5
5
Enter first string: 5
Enter second string: 5
True
Enter string: 5
Old substring: 5
New substring: 5
5
Name: Alice, Age: 20
A
5 8
8
Bob
Enter a number: 5
2.0
```

	Done! Enter a number: 5 Enter a number: 5 2.0 File not found!
Full Screen:	

Practical Preparation for Exam TCIE-2.py - C:/Users/ASUS/AppData/Local/Programs/Python/Python313/Practica...

```
# Hello, World!
print("Hello, World!")

# Taking user input and greeting
name = input("Enter your name: ")
print(f"Hello, {name}!")

# Swapping two numbers without a third variable
a, b = 5, 10
a, b = b, a
print("After swapping:", a, b)

# Check if a number is even or odd
num = int(input("Enter a number: "))
print("Even" if num % 2 == 0 else "Odd")

# Find the largest among three numbers
a, b, c = 3, 7, 5
print("Largest:", max(a, b, c))

# Print numbers from 1 to 10 using a for loop
for i in range(1, 11):
    print(i)

# Sum of first n natural numbers using a while loop
n = int(input("Enter n: "))
sum_n = 0
while n > 0:
```

Ln: 11 Col: 0

IDLE Shell 3.13.1

```
/Python/Python313/Practical Preparation for Exa
m TCIE-2.py
Hello, World!
Enter your name: Mohmadhusen
Hello, Mohmadhusen!
After swapping: 10 5
Enter a number: 5
Odd
Largest: 7
1
2
3
4
5
6
7
8
9
10
Enter n: 5
Sum: 15
Enter n: 5
0 1 1 2 3 Enter a number: 5
Prime
Enter a number: 5
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
```

Ln: 19 Col: 10

File Edit Format Run Options Window Help
File Edit Shell Debug Options Window Help
36°C Sunny
Search
ENG IN
1:53 pm 22/03/2025

*****