```
# 1. Check Python Version
import sys
print("Python Version:", sys.version)

# 2. Variables and Data Types
integer_var = 42
float_var = 3.14
string_var = "Hello, Python!"
boolean_var = True
list_var = [1, 2, 3]
print("Values:", integer_var, float_var, string_var, boolean_var, list_var)
print("Types:", type(integer_var), type(float_var), type(string_var),
type(boolean_var), type(list_var))

# 3. Data Type Conversion
int_to_str = str(100)
float_to_int = int(5.67)
str_to_float = float("123.45")
print("Integer to String:", int_to_str, type(int_to_str))
print("Float to Integer:", float_to_int, type(float_to_int))
print("String to Float:", str_to_float, type(str_to_float))

# 4. Arithmetic and Comparison Operators
a, b = 10, 5
print("Addition:", a + b)
```

```
print("Subtraction:", a - b)
print("Multiplication:", a * b)
print("Division:", a / b)
print("Modulus:", a % b)
print("Exponentiation:", a ** b)
print("Comparison (a > b):", a > b)

# 5. Logical and Identity Operators
num = 8
print("Is positive and even?", num > 0 and num % 2 == 0)
print("Is num an integer?", isinstance(num, int))

# 6. Decision Making (if-else)
num = int(input("Enter a number: "))
if num > 0:
    print("Positive")
elif num < 0:
    print("Negative")
else:
    print("Zero")

# 7. Check Even or Odd
num = int(input("Enter a number: "))
print("Even" if num % 2 == 0 else "Odd")
```

```
# 8. Find the Largest of Three Numbers
a, b, c = map(int, input("Enter three numbers: ").split())
largest = max(a, b, c)
print("Largest:", largest)

# 9. Print Numbers from 1 to N
N = int(input("Enter N: "))
print("Using for loop:")
for i in range(1, N + 1):
    print(i, end=" ")
print("\nUsing while loop:")
i = 1
while i <= N:
    print(i, end=" ")
    i += 1
print()

# 10. Print Multiplication Table
num = int(input("Enter a number: "))
for i in range(1, 11):
    print(f"{num} x {i} = {num * i}")

# 11. Sum of First N Natural Numbers
N = int(input("Enter N: "))
sum_n = sum(range(1, N + 1))
```

```
print("Sum:", sum_n)

# 12. Factorial of a Number
num = int(input("Enter a number: "))
factorial = 1
for i in range(1, num + 1):
    factorial *= i
print(f"Factorial of {num} is {factorial}")

# 13. Fibonacci Series
N = int(input("Enter N for Fibonacci: "))
a, b = 0, 1
print(a, end=" ")
for _ in range(N - 1):
    print(b, end=" ")
    a, b = b, a + b
print()

# 14. Armstrong Number
num = int(input("Enter a number: "))
num_str = str(num)
power = len(num_str)
sum_digits = sum(int(d) ** power for d in num_str)
print(f"{num} is {'an' if num == sum_digits else 'not an'} Armstrong number")
```

```
# 15. Prime Number Check
num = int(input("Enter a number: "))
is_prime = num > 1 and all(num % i != 0 for i in range(2, int(num ** 0.5) + 1))
print(f"{num} is {'prime' if is_prime else 'not prime'}")

# 16. Reverse a Number
num = int(input("Enter a number: "))
reverse = int(str(num)[::-1])
print("Reversed:", reverse)

# 17. String Operations
text = input("Enter a string: ")
print("Uppercase:", text.upper())
print("Lowercase:", text.lower())
print("Reversed:", text[::-1])

# 18. Palindrome Check (String & Number)
input_str = input("Enter a string or number: ")
is_palindrome = input_str == input_str[::-1]
print(f"{input_str} is {'a' if is_palindrome else 'not a'} palindrome")

# 19. Count Vowels and Consonants in a String
text = input("Enter a string: ").lower()
vowels = sum(1 for c in text if c in 'aeiou')
consonants = sum(1 for c in text if c.isalpha() and c not in 'aeiou')
```

```
print("Vowels:", vowels, "Consonants:", consonants)

# 20. Remove Duplicates from a List
lst = list(map(int, input("Enter numbers separated by space: ").split()))
unique_lst = list(dict.fromkeys(lst))
print("List without duplicates:", unique_lst)

# 21. Function to Find Factorial
def factorial(n):
    return 1 if n <= 1 else n * factorial(n - 1)
num = int(input("Enter a number: "))
print(f"Factorial of {num} is {factorial(num)}")

# 22. Function to Check Armstrong Number
def is_armstrong(num):
    num_str = str(num)
    power = len(num_str)
    return num == sum(int(d) ** power for d in num_str)
num = int(input("Enter a number: "))
print(f"{num} is {'an' if is_armstrong(num) else 'not an'} Armstrong number")

# 23. Function to Find Maximum in a List
def find_max(lst):
    return max(lst)
lst = list(map(int, input("Enter numbers separated by space: ").split()))
```

```
print("Maximum:", find_max(lst))

# 24. Read and Write to a File
with open("sample.txt", "w") as f:
    f.write("Hello, Python File Handling!")
with open("sample.txt", "r") as f:
    print("File content:", f.read())

# 25. Count Words in a File
with open("sample.txt", "w") as f:
    f.write("This is a test file with some words.")
with open("sample.txt", "r") as f:
    words = f.read().split()
print("Word count:", len(words))

# 26. Copy Contents of One File to Another
with open("source.txt", "w") as f:
    f.write("Content to copy.")
with open("source.txt", "r") as src, open("dest.txt", "w") as dst:
    dst.write(src.read())
with open("dest.txt", "r") as f:
    print("Copied content:", f.read())

# 27. Create a Class and Object
class Car:
```

```
    def __init__(self, brand, model):
        self.brand = brand
        self.model = model
    def display(self):
        return f"{self.brand} {self.model}"
my_car = Car("Toyota", "Camry")
print(my_car.display())

# 28. Inheritance in Python
class Animal:
    def speak(self):
        return "I make a sound"
class Dog(Animal):
    def speak(self):
        return "Woof!"
dog = Dog()
print(dog.speak())

# 29. Method Overloading (Simulated)
class Math:
    def add(self, a, b, c=0):
        return a + b + c
math = Math()
print("Add two numbers:", math.add(2, 3))
print("Add three numbers:", math.add(2, 3, 4))
```

```
# 30. Method Overriding
class Parent:
    def show(self):
        return "Parent method"
class Child(Parent):
    def show(self):
        return "Child method"
child = Child()
print(child.show())

# 31. Polymorphism Example
class Cat:
    def speak(self):
        return "Meow"
class Dog:
    def speak(self):
        return "Woof"
def animal_sound(animal):
    print(animal.speak())
cat, dog = Cat(), Dog()
animal_sound(cat)
animal_sound(dog)

# 32. Matrix Addition
```

```
matrix1 = [[1, 2], [3, 4]]
matrix2 = [[5, 6], [7, 8]]
result = [[matrix1[i][j] + matrix2[i][j] for j in range(2)] for i in range(2)]
print("Matrix Addition Result:", result)

# 33. Transpose of a Matrix
matrix = [[1, 2], [3, 4]]
transpose = [[matrix[j][i] for j in range(2)] for i in range(2)]
print("Transpose:", transpose)

# 34. Merge Two Sorted Lists
list1 = list(map(int, input("Enter sorted list 1: ").split()))
list2 = list(map(int, input("Enter sorted list 2: ").split()))
merged = []
i, j = 0, 0
while i < len(list1) and j < len(list2):
    if list1[i] <= list2[j]:
        merged.append(list1[i])
        i += 1
    else:
        merged.append(list2[j])
        j += 1
merged += list1[i:] + list2[j:]
print("Merged sorted list:", merged)
```

```
# 35. Find Second Largest Number in a List
lst = list(map(int, input("Enter numbers: ").split()))
unique_lst = sorted(set(lst), reverse=True)
print("Second largest:", unique_lst[1] if len(unique_lst) > 1 else "No second largest")

# 36. Find GCD and LCM of Two Numbers
def gcd(a, b):
    while b:
        a, b = b, a % b
    return a
def lcm(a, b):
    return (a * b) // gcd(a, b)
a, b = map(int, input("Enter two numbers: ").split())
print("GCD:", gcd(a, b), "LCM:", lcm(a, b))

# 37. Generate a Random Password
import random
import string
length = 12
chars = string.ascii_letters + string.digits + string.punctuation
password = ''.join(random.choice(chars) for _ in range(length))
print("Random Password:", password)

# 38. Implement a Simple Calculator
```

```
def calculator(a, b, op):
    if op == '+': return a + b
    if op == '-': return a - b
    if op == '*': return a * b
    if op == '/': return a / b if b != 0 else "Error: Division by zero"
a, op, b = input("Enter expression (e.g., 5 + 3): ").split()
a, b = float(a), float(b)
print("Result:", calculator(a, b, op))

# 39. Count Frequency of Words in a String
from collections import Counter
text = input("Enter a string: ").lower().split()
freq = Counter(text)
for word, count in freq.items():
    print(f"{word}: {count}")
```