☑ 1. Introduction to Python & Overview

Q1. Print "Welcome to Python Programming!"
Q2. Print your name, college, and course in 3 separate lines.
Q3. Write a Python program to display the result of 5 + 3 * 2.

☑ 2. Variables & Assignment

Q4. Create variables for your name, age, and city, and print them.
Q5. Swap values of two variables: a = 5, b = 10.
Q6. Take input from the user for name and print: “Hello, [name]!”

☑ 3. Basic Data Types

Q7. Create and print one variable of type int, float, str, bool, None.
Q8. Use type() to print data types of above variables.
Q9. Convert a float to an int and string to int (if valid).

☑ 4. Control Flow + Conditional Statements

Q10. Check if a number is even or odd.
Q11. If marks > 90, print “Excellent”, 60–90: “Good”, else “Needs Improvement”.
Q12. Find the largest among 3 numbers entered by the user.

☑ 5. Loops and Iterations

Q13. Print numbers from 1 to 10 using a for loop.
Q14. Print even numbers from 2 to 20 using a while loop.
Q15. Print a multiplication table of a number using loops.

☑ 6. List Comprehensions

Q16. Create a list of squares from 1 to 10 using list comprehension.
Q17. Create a list of all even numbers from 1 to 50.
Q18. Filter a list to keep only numbers > 10: [5, 12, 3, 21, 7].

☑ 7. Functions & Modules

Q19. Create a function greet() that prints “Welcome to Python!”
Q20. Write a function that returns the cube of a number.
Q21. Use math module to find the square root and factorial of a number.

☑ 8. Arguments & Parameters

Q22. Create a function that accepts name and prints greeting.
Q23. Create a function with default parameter age=18.
Q24. Create a function that takes 2 numbers and returns their sum.

☑ 9. Modules & Packages

Q25. Import random module and print a random number.
Q26. Use math.pi to print the area of a circle.
Q27. Use datetime to print current date and time.

☑ 10. Data Structures: Lists & Tuples

Q28. Create a list of 5 fruits and print the third fruit.
Q29. Append a new fruit and print updated list.
Q30. Create a tuple of 3 colors and try to change one element (observe error).

☑ 11. Data Structures: Dictionaries & Sets

Q31. Create a dictionary with keys: name, age, city. Print each key-value.
Q32. Add a new key college to the dictionary.
Q33. Create a set of 5 unique numbers and try adding duplicates.

☑ 12. Object-Oriented Programming

Q34. Create a class Student with attributes name and age, and method display().
Q35. Create a child class GraduateStudent inheriting from Student.
Q36. Use __init__ and __str__() method in your class.

☑ 13. File Handling & Exception Handling

Q37. Write your name into a file named info.txt.
Q38. Read and print the content from the file.
Q39. Handle file not found using try-except.

☑ 14. Python Libraries: NumPy

Q40. Create a NumPy array of numbers from 1 to 10.
Q41. Find the mean and reshape it into 2x5 matrix.
Q42. Square each element using vectorized operation.

☑ 15. Python Libraries: Pandas

Q43. Create a DataFrame with columns: Name, Age, City.
Q44. Print only names where Age > 20.
Q45. Add a new column “Marks” and display DataFrame.

☑ 16. Python Libraries: Matplotlib

Q46. Plot a line graph for x = [1, 2, 3, 4], y = [2, 4, 6, 8].
Q47. Add labels for X, Y and graph title.
Q48. Save the graph as an image.

That’s 48 Python assignment questions with increasing complexity, designed to cover beginner-to-intermediate topics.

Solutions

```
# 1. Introduction to Python & Overview
print("Welcome to Python Programming!")
print("Mahendra Bella")
print("SV College of Engineering")
print("B.Tech CSE - AIML")
print(5 + 3 * 2)


# 2. Variables & Assignment
name = "Mahendra"
age = 21
city = "Kadapa"
print(name, age, city)
a, b = 5, 10
a, b = b, a
print(a, b)
user_name = input("Enter your name: ")
print("Hello,", user_name)
```

```
# 3. Basic Data Types
x = 10
y = 3.14
z = "Python"
b = True
n = None
print(type(x), type(y), type(z), type(b), type(n))
print(int(3.7))
print(int("7"))

# 4. Control Flow + Conditionals
num = int(input("Enter a number: "))
print("Even" if num % 2 == 0 else "Odd")
marks = int(input("Enter marks: "))
if marks > 90:
    print("Excellent")
elif marks >= 60:
    print("Good")
else:
    print("Needs Improvement")
a, b, c = 3, 5, 1
print("Largest is", max(a, b, c))

# 5. Loops and Iterations
for i in range(1, 11):
    print(i)
i = 2
while i <= 20:
    print(i)
    i += 2
n = int(input("Enter a number: "))
```

```
for i in range(1, 11):
    print(f"{n} x {i} = {n*i}")

# 6. List Comprehensions
squares = [i**2 for i in range(1, 11)]
print(squares)
evens = [i for i in range(1, 51) if i % 2 == 0]
print(evens)
filtered = [i for i in [5, 12, 3, 21, 7] if i > 10]
print(filtered)

# 7. Functions & Modules
def greet():
    print("Welcome to Python!")
greet()
def cube(n):
    return n ** 3
print(cube(3))
import math
print(math.sqrt(16), math.factorial(5))

# 8. Arguments & Parameters
def hello(name):
    print("Hello,", name)
hello("Mahendra")
def user_info(age=18):
    print("Age:", age)
user_info()
def add(a, b):
    return a + b
print(add(4, 6))
```

```
# 9. Modules & Packages
import random
print(random.randint(1, 100))
import math
r = 7
area = math.pi * r ** 2
print("Area:", area)
from datetime import datetime
print(datetime.now())

# 10. Lists & Tuples
fruits = ["apple", "banana", "cherry", "mango", "kiwi"]
print(fruits[2])
fruits.append("orange")
print(fruits)
colors = ("red", "green", "blue")
# colors[0] = "yellow"  # Error: Tuples are immutable

# 11. Dictionaries & Sets
d = {"name": "Mahendra", "age": 21, "city": "Kadapa"}
for k, v in d.items():
    print(k, ":", v)
d["college"] = "SVCE"
print(d)
s = {1, 2, 3, 3, 4, 5}
print(s)

# 12. OOP - Classes, Inheritance
class Student:
    def __init__(self, name, age):
        self.name = name
        self.age = age
    def display(self):
        print(self.name, self.age)
s1 = Student("Mahendra", 21)
s1.display()
class GraduateStudent(Student):
    def __init__(self, name, age, course):
        super().__init__(name, age)
        self.course = course
    def display(self):
        print(self.name, self.age, self.course)
gs = GraduateStudent("Mahendra", 21, "AI/ML")
gs.display()

# 13. File Handling & Exceptions
with open("info.txt", "w") as f:
    f.write("Mahendra Bella\n")
with open("info.txt", "r") as f:
    print(f.read())
try:
    with open("nonexistent.txt") as f:
        print(f.read())
except FileNotFoundError:
    print("File not found!")

# 14. NumPy
import numpy as np
a = np.arange(1, 11)
print(a)
print(a.mean())
print(a.reshape(2, 5))
print(a ** 2)

# 15. Pandas
import pandas as pd
data = {"Name": ["A", "B", "C"], "Age": [21, 25, 19], "City": ["Kadapa", "Hyd", "Delhi"]}
df = pd.DataFrame(data)
print(df[df["Age"] > 20])
df["Marks"] = [88, 92, 79]
print(df)

# 16. Matplotlib
import matplotlib.pyplot as plt
x = [1, 2, 3, 4]
y = [2, 4, 6, 8]
plt.plot(x, y)
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.title("Simple Line Graph")
plt.savefig("line_graph.png")
plt.show()
```