**Unit-1**

1. Introduction to Python.

2. How do you define a variable and assign a value to it in Python?

3. What are the different comment lines supported by the python?

4. What are the inbuilt data type?

5. What are the input output functions used in python explain?

6. Difference between list, tuple, set, dictionary?

7. what are Types of operators in python, explain membership and bitwise operator in details.

8. What is string? Explain all the string function.

9. Explain all List inbuilt functions. How do you remove an item from list.

10. What are the types of function? Explain in details.

11. What is the use of range function in for loop?

12. What are the condition control statements? With example

13. Jump statements in python.

14. Write a python program to display prime numbers till N value.

15. Write a program to check for palindrome.

**Unit-2**

1. What are the types of inheritance? Explain in details.

2. Difference between compile time and runtime polymorphism.

3. Explain different access modifiers in details.

4. Explain multiple inheritance? With suitable program.

5. What is constructor? Explain its type.

6. What is super function? Explain with suitable program.

7. What is self keyword? What is the use self keyword.

8. What is try catch and finally clause explain with example.

9. What is abstraction? How it used, explain in details.

10. How to read and write file. Explain in details.

11. Write a program to create a files called even.txt and odd.txt and write the first 50 even numbers into even, and first 50 odd numbers into odd.

**Unit-3**

1. What is a module in Python, and how does it differ from a package?

2. Can you name three popular Python libraries used for data analysis?

3. Explain the concept of namespace collision and how it's avoided in Python modules.

4. How does PyCharm facilitate integration with Git for version control?

5. Write a simple Python script demonstrating the use of a custom module.

6. Use PyCharm to create a new Git repository and commit a change to a Python file.

7. Compare and contrast the advantages of using modules versus packages in Python development.

8. Analyze the benefits of using PyTest for testing Python code compared to traditional unit testing methods.

9. Assess the effectiveness of PyCharm as an IDE for Python development, considering factors such as ease of use, features, and community support.

10. Evaluate the advantages and disadvantages of using MySQL versus MongoDB for CRUD operations in Python applications.

11. Design a Python package structure for a web development project, including necessary modules and sub-packages.

12. Develop a PyCharm plugin to automate a repetitive task in Python development.

13. Explain the process of connecting Python with a MySQL database and performing CRUD operations.

14. How does PyCharm assist in managing Python virtual environments?

15. Write Python code to connect to a MongoDB database and perform a read and write operations on a collection.

16. Compare the advantages and disadvantages of using NoSQL databases like MongoDB versus traditional SQL databases like MySQL for Python applications.

17. Evaluate the significance of version control systems like Git in collaborative Python development projects.