**Here are the interview questions:**

**1.String :**

1. \*\*Reverse a String\*\*

2. \*\*Check if a String is a Palindrome\*\*

3. \*\*Count the Number of Vowels in a String\*\*

4. \*\*Find the Frequency of Characters in a String\*\*

5. \*\*Check if Two Strings are Anagrams\*\*

6. \*\*Remove Duplicate Characters from a String\*\*

7. \*\*Check if a String Contains Only Digits\*\*

8. \*\*Convert the First Letter of Each Word to Uppercase\*\*

**2.List:**

1. \*\*Reverse a List Without Using Built-in Functions\*\*

2. \*\*Find the Largest and Smallest Elements in a List\*\*

3. \*\*Remove Duplicates from a List\*\*

4. \*\*Check if a List is Empty\*\*

5. \*\*Find the Second Largest Element in a List\*\*

6. \*\*Count the Frequency of Each Element in a List\*\*

7. \*\*Flatten a Nested List\*\*

8. \*\*Merge Two Lists and Remove Duplicates\*\*

9. \*\*Find the Intersection of Two Lists\*\*

10. \*\*Rotate a List by `n` Positions\*\*

11. \*\*Check if Two Lists are Identical\*\*

12. \*\*Split a List into Even and Odd Numbers\*\*

13. \*\*Find the Cumulative Sum of a List\*\*

14. \*\*Sort a List Without Using the `sort()` Method\*\*

15. \*\*Find All Pairs in a List That Sum Up to a Given Number\*\*

**3.Set:**

1. \*\*Remove Duplicates from a List Using a Set\*\*

2. \*\*Find the Union of Two Sets\*\*

3. \*\*Find the Intersection of Two Sets\*\*

4. \*\*Find the Difference Between Two Sets\*\*

5. \*\*Check if a Set is a Subset of Another Set\*\*

6. \*\*Check if Two Sets are Disjoint\*\*

7. \*\*Remove an Element from a Set\*\*

8. \*\*Add an Element to a Set\*\*

9. \*\*Find the Symmetric Difference Between Two Sets\*\*

10. \*\*Check if a Set is Empty\*\*

11. \*\*Convert a List to a Set and Vice Versa\*\*

12. \*\*Check if Two Sets are Equal\*\*

13. \*\*Get All Unique Characters from a String Using a Set\*\*

14. \*\*Count the Number of Unique Elements in a List\*\*

15. \*\*Iterate Over a Set and Print Each Element\*\*

**4. \*\*Tuple Questions:\*\***

1. \*\*Access Elements in a Tuple\*\*

2. \*\*Check if an Element Exists in a Tuple\*\*

3. \*\*Count the Occurrences of an Element in a Tuple\*\*

4. \*\*Find the Index of an Element in a Tuple\*\*

5. \*\*Convert a List to a Tuple and Vice Versa\*\*

6. \*\*Unpack a Tuple into Variables\*\*

7. \*\*Concatenate Two Tuples\*\*

8. \*\*Check if Two Tuples are Identical\*\*

9. \*\*Sort a Tuple\*\*

10. \*\*Find the Length of a Tuple\*\*

11. \*\*Create a Tuple with Single Element (and Explain the Syntax)\*\*

12. \*\*Reverse a Tuple\*\*

13. \*\*Convert a Tuple of Tuples to a Single Tuple\*\*

14. \*\*Iterate Over a Tuple and Print Each Element\*\*

15. \*\*Create a Tuple Without Using Parentheses (Tuple Packing)\*\*

**5. \*\*Dictionary Questions:\*\***

1. \*\*Access the Value Associated with a Key\*\*

2. \*\*Check if a Key Exists in a Dictionary\*\*

3. \*\*Iterate Over a Dictionary and Print All Key-Value Pairs\*\*

4. \*\*Merge Two Dictionaries\*\*

5. \*\*Remove a Key from a Dictionary\*\*

6. \*\*Find the Maximum and Minimum Values in a Dictionary\*\*

7. \*\*Sort a Dictionary by Keys or Values\*\*

8. \*\*Convert Two Lists (Keys and Values) into a Dictionary\*\*

9. \*\*Get a List of All Keys and Values Separately\*\*

10. \*\*Count the Frequency of Each Character in a String Using a Dictionary\*\*

11. \*\*Update the Value of an Existing Key in a Dictionary\*\*

12. \*\*Get the Default Value for a Non-Existent Key Without Raising an Error\*\*

13. \*\*Reverse the Keys and Values in a Dictionary\*\*

14. \*\*Create a Dictionary Using Dictionary Comprehension\*\*

15. \*\*Remove All Entries from a Dictionary (Clear the Dictionary)\*\*

`**if-else` statements question:**

1. \*\*Check if a Number is Positive, Negative, or Zero\*\*

2. \*\*Determine if a Person is Eligible to Vote Based on Age\*\*

3. \*\*Check if a Year is a Leap Year\*\*

4. \*\*Find the Largest of Three Numbers\*\*

5. \*\*Check if a Character is a Vowel or Consonant\*\*

6. \*\*Determine if a Given Number is Even or Odd\*\*

7. \*\*Check if a String is Empty or Not\*\*

8. \*\*Determine the Grade Based on a Score (e.g., A, B, C, D, F)\*\*

9. \*\*Check if Two Numbers are Equal, Greater, or Lesser\*\*

10. \*\*Check if a Number is Divisible by Both 3 and 5\*\*

11. \*\*Implement a Simple Calculator Using `if-elif-else` Statements\*\*

12. \*\*Check if a Number is Within a Certain Range (e.g., between 1 and 100)\*\*

13. \*\*Determine if a String Starts with a Vowel\*\*

14. \*\*Check if a Given Year is a Century Year\*\*

15. \*\*Determine if a Person is a Child, Teen, or Adult Based on Age\*\*

**\*\*For Loop Questions:\*\***

1. \*\*Print All Elements in a List Using a For Loop\*\*

2. \*\*Calculate the Sum of All Numbers in a List\*\*

3. \*\*Find the Factorial of a Given Number\*\*

4. \*\*Print the Multiplication Table of a Given Number\*\*

5. \*\*Print the Fibonacci Sequence Up to `n` Terms\*\*

6. \*\*Count the Number of Even and Odd Numbers in a List\*\*

7. \*\*Reverse a String Using a For Loop\*\*

8. \*\*Find the Maximum and Minimum Values in a List\*\*

9. \*\*Print All Prime Numbers Within a Given Range\*\*

10. \*\*Iterate Over a Dictionary and Print Each Key-Value Pair\*\*

11. \*\*Find the Length of Each Word in a List of Strings\*\*

12. \*\*Create a List of Squares for Numbers From 1 to 10\*\*

13. \*\*Filter Out Only Positive Numbers From a List\*\*

14. \*\*Print Each Character of a String Separately\*\*

15. \*\*Check if an Element Exists in a List Without Using the `in` Keyword\*\*

**\*\*While Loop Questions:\*\***

1. \*\*Print Numbers From 1 to 10 Using a While Loop\*\*

2. \*\*Calculate the Sum of Digits of a Given Number\*\*

3. \*\*Reverse a Number Using a While Loop\*\*

4. \*\*Print a Countdown From 10 to 1\*\*

5. \*\*Find the Greatest Common Divisor (GCD) of Two Numbers\*\*

6. \*\*Keep Taking Input From the User Until They Enter 'exit'\*\*

7. \*\*Check if a Number is a Palindrome\*\*

8. \*\*Generate the Fibonacci Sequence Until a Specified Number\*\*

9. \*\*Calculate the Power of a Number Without Using the `\*\*` Operator\*\*

10. \*\*Print All Even Numbers Between 1 and 100\*\*

11. \*\*Sum of Natural Numbers Until a Given Number\*\*

12. \*\*Implement a Simple Menu-Driven Program Using While Loop\*\*

13. \*\*Simulate a Basic Password Check That Limits to 3 Attempts\*\*

14. \*\*Keep Multiplying a Number by 2 Until It Becomes Greater Than 1000\*\*

15. \*\*Print the Digits of a Number in Reverse Order\*\*

list of practice questions on **single**, **multiple**, **multilevel**, and **hierarchical** inheritance.

**Single Inheritance**

1. **Vehicle Example:**
   * Create a Vehicle class with attributes like brand and model.
   * Create a Car class that inherits from Vehicle and adds an attribute number\_of\_doors.
   * Write a program to demonstrate creating and displaying details of a Car.
2. **Shape Example:**
   * Create a Shape class with a method calculate\_area().
   * Create a Circle class that inherits from Shape and implements the calculate\_area() method.
   * Test it by creating a Circle object and calculating its area.
3. **Animal Example:**
   * Create an Animal class with attributes name and species, and a method speak().
   * Create a Dog class that inherits Animal and overrides the speak() method to return "Woof!".

**Multiple Inheritance**

1. **Person and Worker Example:**
   * Create a Person class with attributes like name and age.
   * Create a Worker class with attributes like company and position.
   * Create an Employee class that inherits from both Person and Worker and adds an employee\_id attribute.
2. **Device Example:**
   * Create a Keyboard class with a method type().
   * Create a Screen class with a method display().
   * Create a Laptop class that inherits from both and implements methods to type and display content.
3. **Sport and Music Example:**
   * Create a Sport class with a method play\_sport().
   * Create a Music class with a method play\_music().
   * Create a TalentedPerson class that inherits from both and demonstrates the ability to play a sport and music.

**Multilevel Inheritance**

1. **Library Example:**
   * Create a LibraryItem base class with attributes title and author.
   * Create a Book class that inherits LibraryItem and adds attributes like publisher.
   * Create an EBook class that inherits Book and adds an attribute file\_size.
2. **Family Tree Example:**
   * Create a Grandparent class with an attribute family\_name.
   * Create a Parent class that inherits Grandparent and adds an attribute occupation.
   * Create a Child class that inherits Parent and adds an attribute school.
3. **Education Example:**
   * Create a Person class with basic details like name and age.
   * Create a Student class that inherits from Person and adds attributes like grade.
   * Create a GraduateStudent class that inherits from Student and adds an attribute thesis\_title.

**Hierarchical Inheritance**

1. **Bank Account Example:**
   * Create a BankAccount base class with methods like deposit() and withdraw().
   * Create a SavingsAccount class that inherits from BankAccount and adds an attribute interest\_rate.
   * Create a CheckingAccount class that inherits from BankAccount and adds an attribute overdraft\_limit.
2. **Animal Example:**
   * Create an Animal base class with methods eat() and sleep().
   * Create a Bird class that inherits Animal and adds a fly() method.
   * Create a Fish class that inherits Animal and adds a swim() method.
3. **Shape Example:**
   * Create a Shape base class with a method calculate\_perimeter().
   * Create a Rectangle class that inherits Shape and implements the perimeter calculation.
   * Create a Triangle class that inherits Shape and implements the perimeter calculation.

**Here are a variety of \*\*for loop pattern questions\*\* for practice, ranging from simple to complex:**

### \*\*Basic Patterns\*\*

1. \*\*Right-Angled Triangle (Stars):\*\*

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

2. \*\*Inverted Right-Angled Triangle:\*\*

\*\*\*\*\*

\*\*\*\*

\*\*\*

\*\*

3. \*\*Number Triangle:\*\*

1

12

123

1234

4. \*\*Same Number Row:\*\*

1

22

333

4444

55555

5. \*\*Reverse Number Triangle:\*\*

54321

5432

543

54

5

```

---

### \*\*Intermediate Patterns\*\*

6. \*\*Pyramid:\*\*

```

\*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*

```

7. \*\*Inverted Pyramid:\*\*

```

\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*

\*\*\*

\*

```

8. \*\*Diamond:\*\*

```

\*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*

\*\*\*

\*

```

9. \*\*Number Pyramid:\*\*

```

1

121

12321

1234321

123454321

```

10. \*\*Hollow Square:\*\*

```

\*\*\*\*\*

\* \*

\* \*

\* \*

\*\*\*\*\*

```

---

### \*\*Advanced Patterns\*\*

11. \*\*Floyd's Triangle:\*\*

```

1

2 3

4 5 6

7 8 9 10

11 12 13 14 15

```

12. \*\*Pascal's Triangle:\*\*

```

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

```

13. \*\*Butterfly Pattern:\*\*

```

\* \*

\*\* \*\*

\*\*\* \*\*\*

\*\*\*\* \*\*\*\*

\*\*\*\*\*\*\*\*\*

\*\*\*\* \*\*\*\*

\*\*\* \*\*\*

\*\* \*\*

\* \*

```

14. \*\*Zig-Zag Pattern:\*\*

```

\* \*

\* \* \* \*

\* \* \*

```

15. \*\*Checkerboard Pattern:\*\*

```

\* \* \* \*

\* \* \*

\* \* \* \*

\* \* \*

```

---

### \*\*Challenges\*\*

16. \*\*Hollow Diamond in a Square:\*\*

```

\*\*\*\*\*\*\*\*\*

\*\*\* \*\*\*

\*\* \*\*

\* \*

\*\* \*\*

\*\*\* \*\*\*

\*\*\*\*\*\*\*\*\*

```

17. \*\*Alphabet Pyramid:\*\*

```

A

ABA

ABCBA

ABCDCBA

ABCDEDCBA

```

18. \*\*Hourglass:\*\*

```

\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*

\*\*\*

\*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*

```

19. \*\*Wave Pattern:\*\*

```

\* \*

\* \*

\*

\* \*

\* \*

```

20. \*\*Binary Triangle:\*\*

```

1

01

101

0101

10101