Python support Engineer role Questions

**1**.These test your core understanding.

| **Topic** | **Sample Questions** |
| --- | --- |
| **Introduction** | What is Python? Why is it popular? |
| **Data Types** | What are the different data types in Python? |
| **Type Conversion** | How can you convert a string into an integer in Python? |
| **Operators** | What is the difference between == and is? |
| **Input/Output** | How do you take input from a user in Python? |
| **Indentation** | Why is indentation important in Python? |

**🧩 2. Data Structures (List, Tuple, Set, Dictionary)**

Most commonly asked in fresher interviews.

| **Type** | **Example Questions** |
| --- | --- |
| **List** | How do you add/remove elements from a list? |
|  | Difference between append() and extend()? |
| **Tuple** | How is a tuple different from a list? |
| **Set** | What is a set? Can it contain duplicate elements? |
| **Dictionary** | How do you access keys and values from a dictionary? |
| **Common** | How do you iterate through a list or dictionary? |

**🧮 3. String Handling**

String questions are very common.

| **Question** | **Example** |
| --- | --- |
| 1.How do you reverse a string in Python? | s[::-1] |
| 2. How do you check if a string is palindrome? | By comparing string with its reverse. |
| 3. What are string methods? | .upper(), .lower(), .split(), .replace() |
| 4. How to count occurrences of a character in a string? | s.count('a') |

**🔁 4. Loops and Conditional Statements**

| **Question** | **Example** |
| --- | --- |
| 1.What are different types of loops in Python? | for and while loops |
| 2. What is the difference between break, continue, and pass? |  |
| 3. How can you use else with loops? |  |
| 4. Write a program to print even numbers from 1 to 20. |  |

**⚙️ 5. Functions and Modules**

| **Question** | **Example** |
| --- | --- |
| 1.What is a function? |  |
| 2. Difference between return and print? |  |
| 3. What are default and keyword arguments? |  |
| 4. What is \*args and \*\*kwargs? |  |
| 5. What are modules and how do you import them? |  |
| 6. Difference between a module and a package? |  |

**🧱 6. Exception Handling**

| **Question** | **Example** |
| --- | --- |
| 1.What is an exception? |  |
| 2. Difference between error and exception? |  |
| 3. How do you handle exceptions in Python? | Using try, except, finally |
| 4. Example: Handle division by zero error. |  |

**🧰 7. File Handling**

| **Question** | **Example** |
| --- | --- |
| 1.How do you read and write a file in Python? |  |
| 2.What modes are available in open() function? |  |
| 3.What does with open() do? |  |
| 4.How do you count lines in a text file? |  |

**🧑‍💻 8. OOPs Concepts**

Even for support roles, basic OOP is often asked.

| **Question** | **Example** |
| --- | --- |
| 1.What is a class and object? |  |
| 2. What is inheritance? Give an example. |  |
| 3. What is polymorphism? |  |
| 4. What is encapsulation? |  |
| 5. What is the difference between a class method and an instance method? |  |

**🔄 9. Debugging & Support-Based Questions**

As a **Python Support Engineer**, debugging & log analysis are key.

| **Question** | **Example** |
| --- | --- |
| 1.How do you debug Python code? |  |
| 2. What is traceback in Python? |  |
| 3. How do you handle ModuleNotFoundError? |  |
| 4. How do you check installed packages? (pip list) |  |
| 5. What command is used to check Python version? |  |
| 6. If your script isn’t running, how will you find the issue? |  |

**🌐 10. Real-World / Scenario-Based Questions**

| **Question** | **Example** |
| --- | --- |
| 1.A Python script fails due to missing module — what will you do? |  |
| 2. How will you handle incorrect file paths in a script? |  |
| 3. What steps will you take if your script shows syntax errors? |  |
| 4. How do you optimize a slow Python script? |  |
| 5. What’s the difference between a bug and an exception? |  |

**🧾 Bonus: Common HR / General Questions**

| **Question** | **Example** |
| --- | --- |
| 1.Tell me about your project using Python. |  |
| 2. What’s your role in debugging or support tasks? |  |
| 3. Why do you want to join as a Python Support Engineer? |  |
| 4. What are your strengths and weaknesses? |  |
| 5. How do you handle pressure or deadlines? |  |