



Accenture Interview Questions and Answers

Introduction

Many students aim for a job at Accenture. The process is competitive, and hundreds apply for the same role. The interview usually starts with an aptitude or coding test. After that, a technical discussion follows. Finally, the HR round checks personality and attitude. Each round looks at different skills. Without preparation, it is easy to feel lost. Using real questions for practice helps more than reading random notes. In this guide, you will find common Accenture interview questions with simple answers. Read them, speak them aloud, and focus on explaining your thoughts clearly.

Accenture Interview Process for Freshers

Accenture's hiring process is not complicated, but it can feel heavy when you are the one sitting for it. Most freshers go through a few stages. Each one tests something slightly different.

- **Online Test:** This usually comes first. Aptitude, reasoning, sometimes coding. The timer runs fast. Many students rush. Reading slowly actually helps more than speed.
- **Technical Round:** This part feels more like a conversation. They ask about your projects. Why you chose a certain method. What went wrong. You might get a small problem to solve on the spot. They observe how you approach it.
- **HR Round:** This is where personality shows. Questions about strengths, weaknesses, long term plans. Nothing too complex, but they notice confidence and clarity.
- **Group Discussion:** Not always included. If it happens, the focus is simple. Can you listen. Can you speak without interrupting. Can you add value without trying to dominate.

Preparation matters. But over-preparing scripted answers does not help. Understanding your own experiences helps more.

Accenture Eligibility Criteria for Students and Freshers

Before you apply to Accenture, it helps to pause and check one thing. Are you actually eligible? Many students skip this step. Later they realise a small detail stopped them from moving forward.

Here's what usually matters.

- **Degree:** Most technical roles prefer BE, BTech, BCA, BSc, MCA or similar backgrounds. If your degree is unrelated, chances become slimmer unless the role is non technical.
- **Academic Marks:** They often expect a minimum percentage in 10th, 12th and graduation. It does not mean you need top ranks. But very low scores or big drops between years may raise questions.
- **Backlogs:** Active backlogs can create issues. Some drives allow a few, some do not. Even if allowed, you usually must clear them before joining.
- **Year Gaps:** A small gap is not a big problem if you can explain it honestly. Health reasons, exam preparation, skill courses. That is fine. A long unexplained break makes interviewers cautious.
- **Basic Skills:** For tech roles, knowing at least one programming language helps. Not just theory. Even small project experience makes a difference. For non tech roles, communication and clarity matter more.

Eligibility rules can change slightly depending on the role and hiring drive. So always read the official notification carefully. When you meet the criteria, you walk into the process with one less thing to worry about. That itself builds confidence.

List of Accenture Interview Questions for Freshers

1. Explain dictionary.
2. Define Tuple.
3. What is XML?

4. What is a classifier in Python?
5. In Java, what are lambda expressions?
6. How is the Java “static” keyword used?
7. Is it possible to implement more than one interface in a Java class?
8. Explain the diamond problem in Java.
9. Distinguish between “++var” and “var++”.
10. What is normalization in the database?

Accenture Technical Interview Questions and Answers for Freshers

1. Explain dictionary.

Data will be kept in a dictionary as key-value pairs, which is an unordered data collection. Key values are used to access elements. Dictionary entries are not limited in number. It serves as an example object.

2. Define Tuple.

A tuple is a collection of ordered data. Numerical index values are used to retrieve elements. There can only be a predetermined number of values in a tuple. When a function returns more than one value, it is helpful.

3. What is XML?

The markup language known as Extensible Markup Language, or XML, provides formatting specifications for documents that are simple for both

computers and humans to understand. Developing information formats, describing the data, and sharing structured data online are all beneficial.

4. What is a classifier in Python?

An algorithm that determines an input element's class based on a collection of features is called a classifier. To gain understandability regarding the relationship between input variables and class, it typically uses training data, which are huge datasets used to train an algorithm or machine learning model. It is mostly employed in supervised and machine learning.

5. In Java, what are lambda expressions?

A function that can be created without being included in any classes is known as a lambda expression. Java 8 brought it.

Its purpose is to offer an interface implementation with a working interface. Writing the implementation code suffices to provide the implementation; redefinition of the method is not necessary. As a result, a lot of code is saved.

**Take your Knowledge
Test Report**

Check your Score



6. How is the Java “static” keyword used?

Java has a non-access modifier called the static keyword that is helpful for managing memory.

Static members can be accessed directly using the class name; there's no need to construct an instance of the class.

For memory management, the static keyword can be used with variables, blocks, functions, and nested classes.

7. Is it possible to implement more than one interface in a Java class?

It is permissible to implement more than one interface in a single class. Multiple interfaces can be implemented into a class in Java to enable multiple inheritances. During implementation, a comma (,) operator is used to separate each interface name.

Syntax:

```
public class ClassName implements Interface1, Interface2,..., InterfaceN
```

```
{
```

```
//Code
```

```
}
```

Example:

```
public class SLA implements X, Y
```

```
{
```

```
//Code
```

```
}
```

8. Explain the diamond problem in Java.

The “Diamond Problem” typically arises when there are several inheritances. The diamond problem in Java arises when you try to implement several interfaces since Java does not enable multiple inheritances.

The compiler encounters uncertainty on which function to invoke when two interfaces with identical signatures are implemented in the same class, leading to a compile-time error. It is referred to as a “diamond problem” because of the way its structure resembles a diamond.

9. Distinguish between “++var” and “var++”.

The “var” variable’s value can be increased by using the expressions “var++” and “++var”.

“var++” is used to evaluate an expression first and then increase its value by 1; this process is known as the post-incrementation of a variable.

The usage of “++var,” which raises a variable’s value to one before the expression is evaluated, indicates that the variable has been pre-incremented.

Example

```
/* C program to explain the difference between var++ and ++var */
```

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
    int x,y;
```

```
x=7, y=1;
```

```
printf("%d %d\n", x++, x); //will generate 7, 8 as output
```

```
printf("%d %d", ++y, y); //will generate 2, 2 as output
```

```
}
```

10. What is normalization in the database?

Normalization, also known as data normalization or database normalization, is a database organization approach that reduces redundant data and enhances data integrity.

We can define a relationship between these tables or columns and arrange the data in tables and columns through database normalization.

**Take your Knowledge
Test Report**

Check your Score



List of Accenture Interview Questions for Experienced

11. What do Python's Pandas mean?
12. What are short, long and medium-term schedules?
13. Create a C++ program to produce the Fibonacci series.

14. Enumerate the Coffman's circumstances that result in an impasse.
15. Explain encapsulation in Java.
16. What is the difference between horizontal and vertical scaling?
17. How does indexing improve database performance?
18. How does garbage collection work in managed environments?
19. What is the difference between multithreading and multiprocessing?
20. How does REST maintain stateless communication?

Accenture Technical Interview Questions and Answers for Experienced

11. What do Python's Pandas mean?

Pandas is an open-source Python software package that is helpful for analyzing and manipulating data.

Numerous data structures and procedures, including the ability to modify time series and numerical tables, are available.

It is regarded as one of the most crucial tools to be able to handle and it can work with various file formats.

Among the qualities or functions that pandas offer are:

- **axes:** A list of row axis labels is returned.
- **empty:** If the series is empty, it returns true; if not, it returns false.
- **size:** The number of elements in the underlying data is returned.

- **values:** The series is returned as ndarray.
- **head():** It gives back the first n rows of a data frame or series.
- **tail():** Returns the n rows at the conclusion of a data frame or series.

12. What are short, long and medium-term schedules?

Short-term scheduler: The most common executor, a short-term scheduler, often called a dispatcher, chooses the exact process that should run next. The scheduler is triggered each time an event takes place. One process could be interrupted as a result of preemption.

Medium-term scheduler: The switching function includes medium-term scheduling. This has to do with processes that are either halted or blocked. Until they are prepared to act, they are swapped out of real memory. Memory-management criteria are used to make the swapping-in decision.

Long-term scheduler: These are programs that are allowed into the system to be processed. It regulates how much multiprogramming is done. After acceptance, a job turns into a procedure.

13. Create a C++ program to produce the Fibonacci series.

Every integer in the Fibonacci series is equal to the sum of the two numbers before it. In a Fibonacci series, the first two numbers are always 0 and 1.

The Fibonacci series is as follows:

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55....

```
#include<iostream.h>
```

```
#include<conio.h>
```

```
void main()
```

```

{

    int num1,num2,nextnum,n,i;

    cout<<"Enter the value for range:"; //Fibonacci series range value will be
inputted

    cin>>n;

    num1=0;

    num2=1;

    cout<<"Fibonacci series is:"<<endl;

    if(n==1)

        cout<<num1<<endl; //Single value will be printed if range value is 1

    else if(n==2)

        cout<<num1<<"\t"<<num2<<endl; //Two values will be printed if the
range value is two

    else

    {

        cout<<num1<<"\t"<<num2<<"\t";

```

```

    for(i=3;i<=n;i++) //Fibonacci series will be printed based on range limit

    {

        nextnum=num1+num2;

        cout<<nextnum<<"\t";

        num1=num2;

        num2=nextnum;

    }

}

getch();

}

```

14. Enumerate the Coffman's circumstances that result in an impasse.

Mutual Exclusion: One process at a time can only use a critical resource.

Hold & Wait: Some resources may be assigned to a process while others are awaited.

No Pre-emption: A process cannot be forced to relinquish any resource it is holding.

Circular Wait: A closed chain of processes known as a “circular wait” occurs when one process in the chain always has at least one resource that another process in the chain needs.

Take your Knowledge
Test Report

Check your Score



15. Explain encapsulation in Java.

The technique of enclosing code and data into a single unit is called encapsulation. A medicine-containing capsule serves as the best illustration of encapsulation.

- In Java, a class is considered fully enclosed if all of its data members are declared as private. This allows us to access the class using getter and setter methods.
- The Java Bean class is an illustration of a fully enclosed class.
- Encapsulation, also known as data-hiding, prevents its data from being seen by other classes.

Example

```
class EncapsulationEg{  
  
    private String empname;  
  
    private int empage;  
  
    private int empid;
```

```
public String getEmpName() //getter method
```

```
{
```

```
    return empname;
```

```
}
```

```
public int getEmpAge()
```

```
{
```

```
    return empage;
```

```
}
```

```
public int getEmpId()
```

```
{
```

```
    return empid;
```

```
}
```

```
public void setEmpName(String setvalue) //setter methods
```

```
{
```

```
    empname=setvalue;
```

```
}
```

```
public void setEmpAge(int setvalue){
```

```
    empage=setvalue;
```

```
}
```

```
public void setEmpId(int setvalue){
```

```
    empid=setvalue;
```

```
}
```

```
}
```

```
public class TestEncapsulation{
```

```
    public static void main(String args[]){
```

```
        EncapsulationEg en= new EncapsulationEg();
```

```
        en.setEmpName("Alvin");
```

```
        en.setEmpAge(22);
```

```
        en.setEmpId(12568);
```

```
        System.out.println("Employee Name: " +en.getEmpAge());
```

```
        System.out.println("Employee Age: " + en.getEmpAge());

        System.out.println("Employee ID: " + en.getEmpId());

    }

}
```

16. What is the difference between horizontal and vertical scaling?

Vertical scaling is the straightforward approach. You increase the power of a single machine. More RAM. Better CPU. It works for a while, especially for smaller systems. But there is a limit. Hardware cannot scale forever.

Horizontal scaling feels more flexible. Instead of upgrading one server, you add more servers and distribute traffic. Load balancers help manage requests. This approach supports growth better and improves fault tolerance. Most modern distributed systems prefer this model.

17. How does indexing improve database performance?

When a table grows large, searching without an index becomes slow because the database scans every row. An index works like a quick lookup structure. It narrows the search area instead of checking everything.

But adding indexes everywhere is not smart. Each insert or update also needs to update the index. So the decision depends on query patterns. Columns frequently used in filters or joins usually benefit the most.

18. How does garbage collection work in managed environments?

In simple terms, the runtime keeps track of objects in memory. If nothing is pointing to an object anymore, it becomes useless. The system eventually clears it.

Most modern runtimes divide memory into generations. New objects go to a young space. If they survive long enough, they move to older spaces. This helps because most objects die early. So cleaning the young space is faster.

It is automatic, yes. But it is not magic. If you create too many objects in tight loops or hold references longer than needed, performance drops. Garbage collection reduces manual memory handling. It does not remove responsibility from the developer.

19. What is the difference between multithreading and multiprocessing?

Threads live inside the same process. They share memory. That makes data exchange fast. It also makes bugs easier to introduce. If two threads modify the same variable without control, things break in unpredictable ways.

Processes are heavier. Each one has its own memory space. That separation improves stability. If one crashes, others may continue running. The trade off is communication overhead. Passing data between processes costs more than sharing memory between threads.

So the choice depends on the problem. CPU bound tasks sometimes benefit from processes. I/O heavy systems often use threads.

**Take your Knowledge
Test Report**

Check your Score



20. How does REST maintain stateless communication?

REST does not store session data on the server side. Each request carries everything required. Headers, tokens, identifiers. Nothing is assumed from previous calls.

Because of that, any server instance can handle any request. You do not need sticky sessions. Scaling becomes simpler.

The downside is that clients must send authentication data each time. That slightly increases request size. But in distributed systems, stateless design usually makes operations cleaner and easier to manage.

Conclusion

We hope these Accenture interview questions and answers will be helpful for you. It is advised that you study frequently asked interview questions, familiarize yourself with Accenture's mission and offerings, and hone your problem-solving and communication skills in advance of the interview. Shape your career by enrolling in our [**placement training institute in Chennai.**](#)