Top 22 Java Interview Questions and Answers for 2019

Prepared by DevTeam.Space

Our Dev Teams Are Trusted By The Best

AIRBUS UBS Disney mi SAMSUNG Yandex Taxi McKESSON

STARTUPS FROM Y Combinator 500startups

DevTeam.Space is a data-driven agile software development platform

Successful businesses and entrepreneurs rely on DevTeam.Space for their most innovative projects

As Seen On

Inc. lifehacker CNBC QUARTZ TECH!CO

HUFFPOST NBC NEWS

Whether you're an entrepreneur, a project manager in a large enterprise, or a CTO, you are well aware that the success of your project rests on your ability to find top developers.

In this guide, you'll find example interview questions and answers you can use when seeking a new Java developer. You'll also get some practical advice on how to use these questions to reliably identify the best people for the job.

First Things First: Select Your Job Requirements

Ultimately, an interview is about finding someone capable of producing the Java code you need while working effectively with your team. To do that, you need to figure out your unique job requirements.

Some example requirements include:

- **Programming skills -** E.g. Object-Oriented Design
- **Java-specific skills**
- **Library/toolkit experience** - E.g. JUnit
- **Design skills** - Performance optimization, building scalable applications, concurrency
- **Communication skills** - Discussing problems and constraints
- **Initiative -** If they'll have to figure out solutions by themselves

Avoid making a shopping list of your perfect Java developer. Instead, focus on what your candidate will really be doing day-to-day. You should keep your requirements list as short as possible. Cut anything they can do without or learn on the job.

With clearly stated requirements, you`ll be able to choose the right Java coding interview questions and have a much better idea of what kinds of responses you are looking for.

In this article, we'll categorize the Java coding interview questions based on the level of expertise you're looking for – Junior, Mid-level, or Senior. Some of the questions will overlap of course, but it's really important that you pay enough attention to the specifics of each of these categories and arrange your interview accordingly.

- Junior Level Interview Questions
- Mid-Level Developer Interview Questions
- Senior Developer Interview Questions

Junior Level Interview Questions

A Junior developer is someone that's just starting their career with less than 2 years of experience. Junior developers demand the lowest salary, but they are limited in the tasks they can take on, make more frequent mistakes and take more time to solve problems. When hiring a junior, you should consider that they'll need a guidance of a more experienced Java professional, who can delegate the tasks and then cross-check the execution done by the juniors.

Common Skill Requirements for Junior Java Developers

- Good basic programming skills
- Foundational Java knowledge
- Object-Oriented Programming
- Learning on the job
- Following instructions and receiving feedback
- Thinking like a programmer

Example Java Basic Interview Questions and Answers

Note: Important keywords are underlined in the answers. Bonus points if the candidate mentions them!

Question 1: What is inheritance?

Requirement: Object-Oriented Programming

Answer: It's the means by which a new class is derived from an existing class. These new classes can inherit or acquire the properties and methods of other classes. In other words, a subclass takes on the characteristics of its superclass or ancestor.

Question 2: Explain public static void main (String args[]).

Requirement: Foundational Java knowledge

Answer: This is a good basic Java interview question since it's one of the first things you have to memorize when you code in this language.

public: This is an access modifier, which means it specifies who can access this method. It means that this method is accessible by any class.

static: This identifies your program as class-based – that is, it can be accessed without first creating the instance of a class.

void: This defines the method's return type, and it says that the method won't return any value.
main: This is the name of the method. JVM searches for this method as a starting point for an application with only this particular signature.

String args[]: The parameters that can be passed to the main method.

Question 3: What is an object?
Requirement: Object-Oriented Programming

Answer: In Java, an object is an instance of a class. It has a state (which is stored in fields, or variables), and a behavior (stored in methods, or functions).

Question 4: What is object cloning in Java?
Requirement: Foundational Java knowledge, Object-Oriented Programming

Answer: Object cloning means creating an exact copy of an object using the Clone() method. It creates a new instanceof the object and initializes all fields to be the same.

Question 5: How would you learn about a new Java class?
Requirement: Learning on the job

Answer: Here you are finding out how the candidate approaches learning. You want to see that they have an idea of what information they will need and where to find it. For Java classes, that means looking up tutorials and opening the documentation.

This is a fairly open ended question. The candidate could talk about their favorite tutorial website or a book they would refer to. They should also mention checking the docs for things like the class member variables, methods and constructors.

Question 6: Tell me about a time you received feedback on a task
Requirement: Following instructions and receiving feedback

Answer: This is an open-ended question. The candidate should demonstrate they can accept, understand and act on feedback.

Question 7: Describe how you would approach solving (some problem) on a high level?
Requirement: Thinking like a programmer

Answer: In this question, the problem should be directly related to the work the candidate will actually be doing. You aren't looking for a perfect answer. Instead, listen to how the candidate approaches solving the problem, their ability to break a problem down into parts, and if they can anticipate problems well.

Mid-Level Developer Interview Questions

Skip to the Senior developer section for the Java advanced interview questions.

Next we have interview questions and answers for experienced Java developers. Mid-level developers are the workhorses of the software development world. Whatever Java app or website you're using, chances are that most of the code was written, checked and tested by a mid-level developer.

Typically with 2-10 years experience, mid-level developers demand a higher salary than juniors. In return, you want developers that can handle most technical tasks on their own and produce excellent code, especially for routine tasks. However, you generally can`t expect high-level design skills, like designing a new type of application from scratch.

Common Skill Requirements for Mid-Level Java Developers

- Deep knowledge of Java
- Relevant Java frameworks and toolkits
- Object-Oriented Programming
- Data structures
- Efficient programming and clean code
- Debugging
- Testing
- Breaking a problem into parts

Example Java Interview Questions and Answers

Note: Important keywords are underlined in the answers. Look out for them in interviews!

Question 8: What are the differences between vectors and arrays?
Requirement: Deep Knowledge of Java, Data structures

Answer: In Java, the main difference is that vectors are dynamically allocated. They contain dynamic lists of references to other objects and can hold data from many different data types. This means their size can easily change as needed. Arrays, on the other hand, are static. They group data with a fixed size and a fixed primitive type.

Question 9: Since Java has a garbage collection feature, does this mean your program can never run out of memory?
Requirement: Deep Knowledge of Java

Answer: Not necessarily. Although Java does have an automatic garbage collection feature, this won't do you any good if you're creating objects faster than your garbage collector can work. So while garbage collection does mean it's harder for your program to run out of memory, don`t push your luck – it can still happen.

Question 10: What is an abstract class? Why are they useful and when would you use one?
Requirement: Object-Oriented Programming

Answer: Abstract classes contain one or more abstract methods (that is, methods that are declared, but contain no implementation). You can't instantiate your abstract classes, so you need subclasses if you want implementations for your abstract methods.

Question 11: What is 'method overriding' in Java?
Requirement: Object-Oriented Programming, Knowledge of Java

Answer: Method overriding is a kind of run-time polymorphism. In Java (and any object-oriented language) method overriding allows a subclass (or derived class/extended class/child class) to provide a different implementation of a method from one of its superclasses (or base class/parent class). To override a method in Java, a subclass method must have the same name, signature and return type as the method in its superclass.

Method overriding always requires inheritance. Here's an example:

```
class Car {
void run(){
System.out.println("car is running");
}
Class Audi extends Car{
void run()
{
System.out.prinltn(“Audi is running safely with
100km”);
}
public static void main( String args[])
{
Car b=new Audi();
b.run();
}
}
```

Question 12: What is the finalize()?
Requirement: Efficient programming and clean code, Knowledge of Java

Answer: The finalize() method is called by the Java garbage collector when it determines there are no references to an object and it can be discarded. The finalize method of the Object class doesn’t do anything, but you can override it to do clean up tasks like dispose of system resources or breaking I/O connections.

Question 13: Write a sample unit testing method for testing exception named as IndexOutOfBoundsException when working with ArrayList
Requirement: Testing, Java JUnit Package

Answer:

```
@Test(expected=IndexOutOfBoundsException.class)
public void outOfBounds() {
new ArrayList<Object>().get(1);
}
```

Question 14: Describe a time you fixed a bug in a complex system? How did you approach the problem? What debugging tools did you use? What did you learn from this experience?
Requirement: Debugging, Breaking down a problem into parts

Debugging one of the key skills for any software developer. However, the real skill is in breaking the problem down in a practical way rather than finding small errors in code snippets. Debugging often takes hours or even days, so you don’t have time in an interview setting. Asking these questions will give you an idea of how your candidate approaches errors and bugs.

Answer: In the candidate's response you should look out for things like:

- A measured, scientific approach
- Breaking down the problem into parts
- Finding out how to reproduce the error
- Expressing and then testing assumptions
- Looking at stack traces
- Getting someone else to help/take a look
- Searching the internet for others that have had the same problem
- Writing tests to check if the bug returns
- Checking the rest of the code for similar errors
- Turn problems into learning experiences

Senior Java Developer Interview Questions

Senior developers are the most experienced and skilled and they are experts in the Java programming language. But with the highest salary range, you`re looking for much more than that.

A great senior Java developer has mastered the entire software lifecycle, including planning, designing, building, testing, deploying and maintaining Java applications. You should also be looking for leadership skills and an interest in the business and legal context of whatever they are developing. To do this, you'll need to go further than just asking Java programming interview questions.

Common Skill Requirements for Senior Java Developers

- Expert Java knowledge
- Object-Oriented Design Patterns
- High-level design skills
- Designing for specific requirements (e.g. security, scalability, optimization)
- DevOps
- Distributed/parallel programming
- Maintaining and upgrading applications
- Leadership skills
- Clear communication skills
- Mentoring junior and mid-level developers
- Understanding business and legal context

Example Java Advanced Interview Questions and Answers

Note: Important keywords are underlined in the answers. Listen out for them!

Question 15. What's the difference between unit, integration and functional testing?

Requirement: Expert Java knowledge

Answer: A unit test works on an isolated piece of code, usually a method. An integration test tests how your code integrates with other systems, for example, a database or a third-party application. A functional test tests the actual functionality of an application and uses automated tools to carry out and simulate user interactions.

Question 16: What is the difference between factory and abstract factory pattern?
Requirement: Object-Oriented Design Patterns

Answer: Abstract Factory provides an extra layer of abstraction. Consider different factories each extended from an Abstract Factory and responsible for the creation of different hierarchies of objects based on the type of factory. E.g. AbstractFactory extended by AutomobileFactory, UserFactory, RoleFactory etc. Each individual factory would create the objects of that specific type.

Question 17: What's the difference between path and classpath variables in Java?
Requirement: Expert Java knowledge

Answer: Both path and classpath are environment variables. However, the operating system uses path to find the executables (which is why you always need to add the directory location in the path variable any time you install Java or want the OS to be able to find the executable). Classpath, on the other hand, is used by Java executables to locate class files.

Question 18: What's the most important thing to look for or check when reviewing another team member's code?
Requirement: Mentoring junior and mid-level developers, Leadership skills

Answer: Here you're checking for analysis skills, knowledge of mistakes that less experienced developers make, keeping in mind the larger project and attention to detail.

A good answer might mention code functionality, readability and style conventions, security flaws that could lead to system vulnerabilities, simplicity, regulatory requirements, or resource optimization.

Open-Ended Questions

Once you've established that your senior dev is a Java expert, you should ask some open-ended questions. They should spark a discussion and you should tailor them to fit your own job requirements. Don't be afraid to ask follow up questions!

Question 19: What tools & practices do you consider necessary for a Continuous Delivery solution?
Requirement: DevOps systems design, Maintaining and upgrading applications

Question 20: You want to synchronize 5 threads to start at the same time. Describe a solution.
Requirement: Parallel/concurrent programming

Question 21: In a 3-tier application running a Java application server you notice freezes for several seconds during high load. What are the most likely reasons? How would you troubleshoot them?
Requirement: Problem solving and troubleshooting, Communication skills, Analysis

Question 22: How would you implement a layered application?
Requirement: High-level design

In Summary

Hiring the right people for your development team is critical to the success of your project. Remember that you should not be aiming to hire the perfect Java developer, but rather the right person for the job at hand.

With the help of this information and sample interview questions on Java, you can ensure that the hiring process goes smoothly and successfully - allowing you to hire a great programming team to get your project completed on time and on budget.

Finally, you can simply print out this ready-to-use cheat sheet of questions and answers and bring it with you to the interview. your candidates.

Happy Hiring!

DevTeam.Space

Hear From DevTeam.Space Clients

DevTeam.Space team is proactive, drawing on operating experience to understand not only your vision but also its purpose; they are skilled, making the right judgment calls and iterating quickly; and they get customer service, providing honest counsel on cost-benefit and real-time process transparency. I highly recommend DTS and look forward to working with them again!

Investment Fund / Website development
NIC POULOS – Bowery Capital

DevTeam.Space has been a great support to us. We needed help with frontend specific projects for a big release. They came on board, with almost no time taken in ramping up with our code base and were able to deliver on time! For fast, effective service, contact DevTeam.Space.

Consumer Services / Frontend
RAHUL THATHOO – MyTime

We had 5 projects and 3 different development teams across multiple tech stacks. One dashboard to navigate all the projects, two project managers, daily updates directly from developers, blockers tracking, and daily stand-up calls have created a productive atmosphere and helped us to move much quicker. If you are looking for high-end software outsourcing services - look no further.

Property management / Backend Infrastructure / Data
JASON JOU – SenStay

Working with DevTeam.Space was a positive and professional experience from the start. They had all the tools in place to support Agile Scrum project management, communicated daily via the dashboard, delivered their Sprints on time, and stayed on top of project blockers. I look forward to working with them again!

Fintech / Backend / Frontend
TONY AMOS - Principis Capital

Successful Businesses and Entrepreneurs Rely on DevTeam.Space for Their Most Innovative Projects

GET STARTED >

Tell us about your challenge & get a free strategy session