

TIPS AND TRICKS TO CRACK THE INTERVIEW

- By Sivareddy

In this session, I will cover
RESUME PREPARATION tips and tricks
Interview Tips and Tricks to crack the interview.

RESUME PREPARATION:

Mention at top of your resume with below details if you have any.

Certifications/technical Blogs/Github URL/Stack overflow/YouTube Channel details
Number of years of experience

- **Java Developer**

Technologies: Core Java, J2EE, Struts, JSF, Spring, Springboot, Restful Webservices, Junit, Mockito, Rest Assured, TDD, BDD

Code Quality tools: Sonarqube, PMD, CheckStyle, FindBugs

Code coverage Tools: Cobertura, Jacoco

CI/CD Tools: Jenkins, Pipeline

SCM: Git, SVN, Perforce

Build Tools: Gradle, Maven and Ant

Application Servers: Tomcat/Web Sphere/Web Logic

Ticketing tool: JIRA, Service Now, Bugzilla

Databases: Oracle, MySQL, PostgreSQL, DB2

NoSql: MongoDB, Cassandra, Redis, CouchDB

Queuing technologies: ActiveMQ, Kafka, RabbitMQ

Good understanding in Algorithms and data structures

- **Full Stack Developer**

Java Developer technologies + Below technologies

Tech Stack: HTML5, CSS, JQuery, Node.js, Angular 5/6/7/8, React JS, Bootstrap, Less, vue js

Code Quality Tools: ESLint, JsLint, Sonarqube

Code Coverage Tools: cover, unittest, karma-code coverage

Unit Framework: MOCHA, JASMINE, KARMA

- **Microservices Developer**

Java Developer Technologies + Below technologies

Microservices Frameworks: Springboot, Micronaut, dropwizard

Container Tools: Docker, Kubernetes, Docker Swarm, OpenShift, rancher

Monitoring: Kibana, Grafana, Sensu, nagios

Cloud Technologies: AWS, Google Cloud, AZURE

Service discovery tools: Consul, Zookeeper, etcd, Eureka

Configuration Management tools: Puppet, Chef, Ansible, SaltStack

Technical Lead :

Java Developer roles + Microservices Developer

- ✓ Working with product managers to understand the requirements
- ✓ Translate requirements into high level design, considering the SOLID design principles
- ✓ Mentoring the junior members in the team
- ✓ Participating in the code review process

- ✓ Creating reusable components following Design patterns
- ✓ Troubleshoot the performance issues, refactoring the code, process improvements, performance tuning

Work Experience:

Mention experience from recent one to previous experience

Education qualifications:

Mention Highest qualification with completion of year

Awards/Achievements:

Technical Stack:

Provide technology stack in table format

Programming Languages	
Java Technologies	
Frame Work	
ORM	
IDE Tool	
Application Servers	
Databases	
Scripting Languages	
Ticketing Tools	
Testing Frameworks	
Code Quality tools	
Code Coverage Tools	
Continuous Integration tool	
DevOps/Build Tools	

Projects:

Project description

Project duration

Roles and responsibilities

Tech stack used for project

Tips and Tricks to crack the interview:

Once you are ready with resume preparation, please make sure you follow the below points to get your resume shortlisted based on the job description.

1. Upload your resume into job portals like Naukuri/LinkedIn/company career websites
2. If you get any job notification for company regarding the open positions, the first step is to go through company profile in GlassDoor/Indeed about company reviews
3. You can check the company reviews, frequently asked interview questions/code programs/benefits/salary etc.
4. Before you submit your profile to any job, go through the job description and may sure that you tweak the profile slightly based on the JD and rename your profile name by prefix or suffix with Company name so that you will remember which profile you submitted for the company
Note: Don't submit same resume for all the jobs
5. Once your resume gets shortlisted for the Job, Please follow the below suggestions.

If you don't have any hands-on experience working on real-time projects, You can go through GitHub(<https://github.com/>), you will find plenty of open source projects related to your technologies.

Java Developers + Full Stack Developers interview preparation Tips:

Prepare for data structures and algorithms or design UI components (mostly you have to crack hacker ranker/any online coding test/ expect you to write a program in the interview)

Below are some of the useful links that may help you with the coding exercise.

<https://www.hackerrank.com/>

<https://leetcode.com/>

<https://www.coderbyte.com/>

<https://www.codechef.com/>

<https://www.topcoder.com/>

2. Prepare well on the topics/technologies mentioned on the Job description
3. Prepare one project with real-time use cases and implement end to end flow.
Example(CRUD operations using Springboot, JPA, Angular 6/React JS)
4. Prepare one example of how you implemented CI/CD process to automate tasks.
Example(Git, Jenkins, Maven/Gradle, Sonarqube, Test cases execution, Deploy the war, etc)
5. Basics on Agile methodology and design patterns
6. TDD approach (JUnit, Mocking, Integration test cases)
7. Code quality tools like Sonarlint, FindBugs, Error-Prone, etc.
8. Basics on Micro Services, RDBMS, NoSql etc.
9. If you are looking for Job opportunity on microservices then you need to understand frameworks like spring-boot, AWS, deployment, Unit and Integration testing

In the case of Manager Round, The questions mostly on the real-time experience like your project architecture,
your daily roles and responsibilities,
What is the tough challenge you faced and how did you resolve it,
what is process for implementing a feature or resolve an issue
how do you troubleshoot performance issue or production issue

Technical Lead interview preparation tips:

Java Developers + Full Stack Developers + below points

1. You have to design an application for the given business use cases around security, scalability and high availability
2. Design Patterns
3. Micro-services design patterns
4. How do you develop/secure/deploy/inter communication between micro services,
5. How do you implement caching for your application
6. Do you have any experience in process automation or productivity improvements
7. What are your roles and responsibilities?
8. How did you secure your application?
9. What are checks you consider for code reviews
10. what are the SOLID design principles and how did you implement them in your project?
11. What are technologies used for Connection pooling
12. Do you have any experience in working with Queuing mechanism like Kafka /Active MQ etc
13. How do you limit the number of requests for each rest API based on threshold?
14. How do you deploy micro-services in Cloud

