

CI Preparation Guide

Object Oriented Principles

<https://beginnersbook.com/2013/04/runtime-compile-time-polymorphism/>

<https://www.geeksforgeeks.org/dynamic-method-dispatch-runtime-polymorphism-java/>

<https://beginnersbook.com/2013/05/aggregation/>

<https://www.geeksforgeeks.org/overloading-in-java/>

Core Java – Collections

<https://crunchify.com/hashmap-vs-concurrenthashmap-vs-synchronizedmap-how-a-hashmap-can-be-synchronized-in-java/>

<http://javahungry.blogspot.com/2014/02/hashmap-vs-concurrenthashmap-java-collections-interview-question.html>

<https://www.baeldung.com/java-exceptions-interview-questions>

<https://javarevisited.blogspot.com/2013/06/10-java-exception-and-error-interview-questions-answers-programming.html>

<http://tutorials.jenkov.com/java-concurrency/race-conditions-and-critical-sections.html>

<https://www.baeldung.com/java-volatile>

<https://jorosjavajams.wordpress.com/volatile-vs-synchronized/>

<https://java2blog.com/java-serialization-interview-questions-and-answers/>

<https://www.journaldev.com/349/java-classloader>

<https://betsol.com/2017/06/java-memory-management-for-java-virtual-machine-jvm/>

<https://stackoverflow.com/questions/1440134/when-should-a-class-be-comparable-and-or-comparator>

<https://www.geeksforgeeks.org/java-util-hashmap-in-java/>

<http://coding-geek.com/how-does-a-hashmap-work-in-java/>

<https://www.youtube.com/watch?v=Nr56SlbMed4>

<https://www.journaldev.com/1058/deadlock-in-java-example>

<http://mrbool.com/working-with-java-executor-framework-in-multithreaded-application/27560>

<https://www.journaldev.com/129/how-to-create-immutable-class-in-java>

<https://www.journaldev.com/2529/jdbc-interview-questions-and-answers>

<https://blogs.oracle.com/poonam/about-g1-garbage-collector,-permanent-generation-and-metaspace>

<https://blog.codecentric.de/en/2008/07/memory-analysis-part-1-obtaining-a-java-heapdump/>

<https://dzone.com/articles/memory-analysis-how-to-obtain-java-heat-dump>

<https://javarevisited.blogspot.com/2015/01/top-20-string-coding-interview-question-programming-interview.html>

Java 8 – Functional Interface, Lambdas, Stream, Optional and Method References etc.

<https://medium.freecodecamp.org/learn-these-4-things-and-working-with-lambda-expressions-b0ab36e0fffc>

<https://www.journaldev.com/2774/java-8-stream>

https://javabrainz.io/courses/java_lambdabasics/lessons/Introduction-To-Streams

<https://www.javatpoint.com/java-8-stream>

<https://www.journaldev.com/2752/java-8-interface-changes-static-method-default-method>

http://www.java2s.com/Tutorials/Java/Stream_How_to/IntStream/Use_lambda_to_check_if_a_integer_is_a_prime_number.htm

<https://www.journaldev.com/10081/javase8-interview-questions-part2>

Hibernate

<https://www.thoughts-on-java.org/complete-guide-inheritance-strategies-jpa-hibernate/>

<https://codingcompiler.com/hibernate-interview-questions-answers/>

<https://howtodoinjava.com/hibernate/complete-hibernate-query-language-hql-tutorial/>

<https://dzone.com/articles/all-about-hibernate-second>

<https://www.journaldev.com/2969/hibernate-caching-first-level-cache>

SQL

<http://a4academics.com/interview-questions/53-database-and-sql/397-top-100-database-sql-interview-questions-and-answers-examples-queries>

<https://www.c-sharpcorner.com/article/most-asked-sql-queries-in-interview-questions/>

<https://www.youtube.com/watch?v=2HVMiPPuPIM>

<https://community.modeanalytics.com/sql/tutorial/sql-performance-tuning/>

MongoDB

<https://www.linkedin.com/learning/learning-mongodb/install-mongodb-on-windows-and-import-data?u=2113185>

[https://www.ibm-bpnetwork.com/linux-blog/mongodb-architecture#:~:text=MongoDB%20Data%20Management%20\(Auto%20Sharding,multiple%20physical%20partitions%20called%20shards.](https://www.ibm-bpnetwork.com/linux-blog/mongodb-architecture#:~:text=MongoDB%20Data%20Management%20(Auto%20Sharding,multiple%20physical%20partitions%20called%20shards.)

<https://dzone.com/articles/why-mongodb>

<https://www.geeksforgeeks.org/how-mongodb-works/>

<https://intellipaat.com/blog/interview-question/mongodb-interview-questions/>

Postgresql

<https://www.linkedin.com/learning/postgresql-essential-training/what-is-postgresql?u=2113185>

<https://www.bestinterviewquestion.com/postgresql-interview-questions#what-is-write-ahead-logging-in-postgresql-alx3v28461a>

<https://www.onlineinterviewquestions.com/postgresql-interview-questions/>

Spring / Spring Boot and Rest/ Spring Data/ Spring integration

<https://www.baeldung.com/spring-interview-questions>
<https://intellipaat.com/interview-question/spring-interview-questions/>
<http://candidjava.com/tutorial/spring-boot-interview-questions/>
<https://www.techbeamers.com/rest-api-interview-questions-answers/>
<https://dzone.com/articles/spring-transactional-amp-exceptions>
<https://www.baeldung.com/exception-handling-for-rest-with-spring>
<https://spring.io/projects/spring-integration>
<https://www.youtube.com/watch?v=JLI94Os175k&list=ULSxukqBZmWsM&index=829>

Design Patterns

<https://www.journaldev.com/1827/java-design-patterns-example-tutorial#strategy-pattern>
<https://www.journaldev.com/1827/java-design-patterns-example-tutorial#template-methodpattern>
<https://www.journaldev.com/1827/java-design-patterns-example-tutorial#builder-pattern>
<https://www.journaldev.com/1827/java-design-patterns-example-tutorial#decorator-pattern>
<https://www.geeksforgeeks.org/singleton-design-pattern/>
<https://www.geeksforgeeks.org/abstract-factory-pattern/>
<https://dzone.com/articles/factory-method-vs-abstract>
<http://www.codespaghetti.com/java-design-pattern-interview-questions/>

Code Smells

<http://www.programmergirl.com/identifying-code-smells-java/>

Code Coverage

<https://www.atlassian.com/continuous-delivery/software-testing/code-coverage>

Exception Handling

<https://stackify.com/best-practices-exceptions-java/>
<https://www.baeldung.com/exception-handling-for-rest-with-spring>
<https://www.toptal.com/java/spring-boot-rest-api-error-handling>

SOLID

<https://www.geeksforgeeks.org/solid-principle-in-programming-understand-with-real-lifeexamples/>
<https://deviq.com/solid/>
<https://www.youtube.com/watch?v=t86v3N4OshQ>

SSL/TLS/Certificates

https://www.youtube.com/watch?v=T4Df5_cojAs

<https://www.digicert.com/ssl/>

https://www.internetsociety.org/deploy360/tls/basics/?gclid=Cj0KCQjwtZH7BRDzARIsAGjbK2b8ewTAQCqRBhoWK2z97_mmfAxz3RFRhu_TnSpZh5v5lZCUdsVMgnMaAmWOEALw_wcB

JWT

<https://auth0.com/learn/json-web-tokens/>

OAuth

<https://www.youtube.com/watch?v=996OiexHze0>

<https://www.varonis.com/blog/what-is-oauth/#:~:text=OAuth%20doesn't%20share%20password,without%20giving%20away%20your%20password.>

OWASP

<https://sectigostore.com/blog/what-is-owasp-what-are-the-owasp-top-10-vulnerabilities/>

Spring Cloud

<https://www.baeldung.com/intro-to-feign>

<https://nirajsonawane.github.io/2019/03/21/Simplifying-Microservices-CommunicationUsing-Spring-Cloud-OpenFeign/>

<https://www.youtube.com/watch?v=aO3W-lYnw-o>

<https://www.youtube.com/watch?v=x1BR0DbuQg&list=PLVz2XdJiJQxz3L2Onpxbel6r72IDdWrJh>

<https://www.youtube.com/watch?v=rIS9eH5tEnY>

Spring Security

<https://www.youtube.com/watch?v=X80nJ5T7YpE>

[https://www.javatpoint.com/spring-security-](https://www.javatpoint.com/spring-security-introduction)

[introduction](https://www.javatpoint.com/spring-security-introduction)

Microservices

<https://cloudacademy.com/blog/microservices-architecture-challenge-advantage-drawback/>

<https://microservices.io/patterns/refactoring/strangler-application.html>

<https://microservices.io/patterns/data/saga.html>

<https://microservices.io/patterns/microservice-chassis.html>

<https://livebook.manning.com/book/microservices-patterns/chapter-2/25>

DDD

<https://www.youtube.com/watch?v=NNFJREcalc0>

<https://martinfowler.com/tags/domain%20driven%20design.html#:~:text=Domain%2DDriven%20Design%20is%20an,through%20a%20catalog%20of%20patterns.>

Unit & Integration Testing

<https://www.toptal.com/java/unit-integration-junit-tests>

Test Driven Development (TDD)

<https://hackernoon.com/introduction-to-test-driven-development-tdd-61a13bc92d92>

<https://www.youtube.com/watch?v=y8TcPr73Bwo>

<https://www.youtube.com/watch?v=s9vt6UJiHg4>

<http://aspiringcraftsman.com/2012/04/01/tdd-best-practices-dont-mock-others/>

<https://github.com/testdouble/contributing-tests/wiki/Don't-mock-what-you-don't-own>

Mocking

<https://www.softwaretestinghelp.com/mockito-interview-questions/>

<https://examples.javacodegeeks.com/core-java/powermockito/powermockito-tutorialbeginners/>

<https://www.youtube.com/watch?v=8S8o46avgAw>

Git

<https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow-workflow>

<https://www.freecodecamp.org/news/an-introduction-to-git-merge-and-rebase-what-they-are-and-how-to-use-them-131b863785f/>

Jenkins

<https://www.youtube.com/watch?v=7KCS70sCoK0>

<https://www.edureka.co/blog/jenkins-master-and-slave-architecture-a-complete-guide/>

<https://www.edureka.co/blog/what-is-jenkins/>

<https://www.infoworld.com/article/3239666/what-is-jenkins-the-ci-server-explained.html>

CI/CD

https://www.youtube.com/watch?v=p7-U1_E_j3w

<https://www.xoriant.com/blog/mobility/introduction-sonarqube.html>

<https://blog.codinghorror.com/code-smells/>

<https://www.javaworld.com/article/2074515/core-java/unit-test-code-coverage-with-maven-and-jacoco.html>

Sonar

<https://www.youtube.com/watch?v=BuT1Ji0P9Ug>

<https://dzone.com/articles/why-sonarqube-1#:~:text=SonarQube%20collects%20and%20analyzes%20source,providing%20reports%20for%20your%20projects.&text=The%20Sonar%20platform%20analyzes%20source,down%20to%20the%20class%20level.>

<https://docs.sonarqube.org/latest/analysis/overview/>

<https://docs.sonarqube.org/latest/architecture/architecture-integration/>

Docker

<https://www.edureka.co/blog/docker-architecture/>

<https://k21academy.com/docker-kubernetes/docker-architecture-docker-engine-components-container-lifecycle/>

<https://geekflare.com/docker-architecture/>

<https://wiki.aquasec.com/display/containers/Docker+Architecture>

<https://www.edureka.co/blog/interview-questions/docker-interview-questions/>

<https://www.toptal.com/docker/interview-questions>

Kafka

<https://data-flair.training/blogs/kafka-architecture/>

Introduction to Amazon Elastic Compute Cloud (EC2)

<https://www.aws.training/Details/Video?id=16382>

<https://www.aws.training/Details/Video?id=16387>

<https://www.udemy.com/course/aws-certified-developer-associate/learn/lecture/4327160#overview>

<https://www.udemy.com/course/aws-certified-developer-associate/learn/lecture/4327160#overview>

AWS S3

<https://www.aws.training/Details/eLearning?id=55038>

https://www.aws.training/LearningLibrary?filters=language%3A1&filters=classification%3A29&search=storage&tab=view_all

<https://www.aws.training/Details/eLearning?id=54609>

<https://www.simplilearn.com/tutorials/aws-tutorial/aws-interview-questions>

<https://www.sanfoundry.com/cloud-computing-amazon-s3-interview-questions-answers/>

AWS Lambda

<https://www.aws.training/Details/eLearning?id=27197>

https://www.aws.training/LearningLibrary?filters=language%3A1&filters=digital%3A1&search=serverless&tab=view_all

<https://www.udemy.com/course/aws-certified-developer-associate/learn/lecture/2329054#overview>

Problem Solving

<https://www.geeksforgeeks.org/subarraysubstring-vs-subsequence-and-programs-to-generate-them/>

<https://www.geeksforgeeks.org/sum-of-all-subsequences-of-an-array/>

<https://www.geeksforgeeks.org/write-a-program-to-reverse-an-array-or-string/>

<https://www.geeksforgeeks.org/minimum-halls-required-for-class-scheduling/?ref=leftbar-rightbar>

<https://www.geeksforgeeks.org/count-of-elements-on-the-left-which-are-divisible-by-current-element/?ref=leftbar-rightbar>

<https://www.geeksforgeeks.org/sum-even-odd-elements-array/>

<https://www.geeksforgeeks.org/maximum-number-of-unique-values-in-the-array-after-performing-given-operations/?ref=leftbar-rightbar>

Other Important Topics

<https://howtodoinjava.com/interview-questions/spring-mvc-interview-questions-with-answers/>

<http://tutorials.jenkov.com/java-concurrency/java-memory-model.html>

<http://tutorials.jenkov.com/java-concurrency/index.html>

<http://tutorials.jenkov.com/java-concurrency/deadlock.html>

<http://tutorials.jenkov.com/java-concurrency/deadlock-prevention.html>
<https://www.baeldung.com/java-executor-service-tutorial>