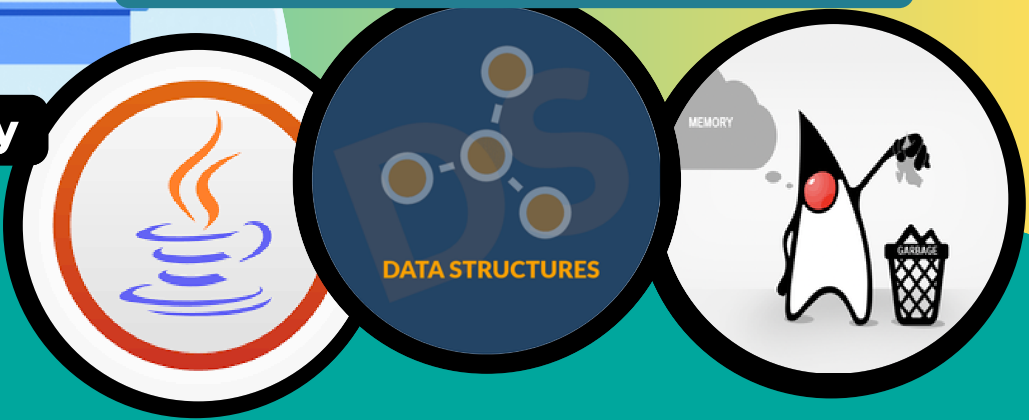
Java Master Course Interview preparation

"We try to learn and master certain areas of Computer Science extensively,then we try to comprehend that knowledge into a form that is understandable to Busy Coder, who doesn't have time for such study."

Busy Coder Academy



Java, OOP, DataStructure, Concurrency, GOF Patterns, Java 5 to 20 features, JVM Master Course

Training by Best Java Trainer with 18+ Experience

"Price, Quality, and Speed Now you get all"

Duration: 3-Months, Session : 8-10PM PM IST(Saturday/Sunday)
Start Date: 27 May 23
Fee for entire course : 8000/- per head

Rajeev Gupta
WhatsApp us
9958543978
rgupta.mtech@gmail.com
https://www.linkedin.com/in/rajeevguptajavatrainer

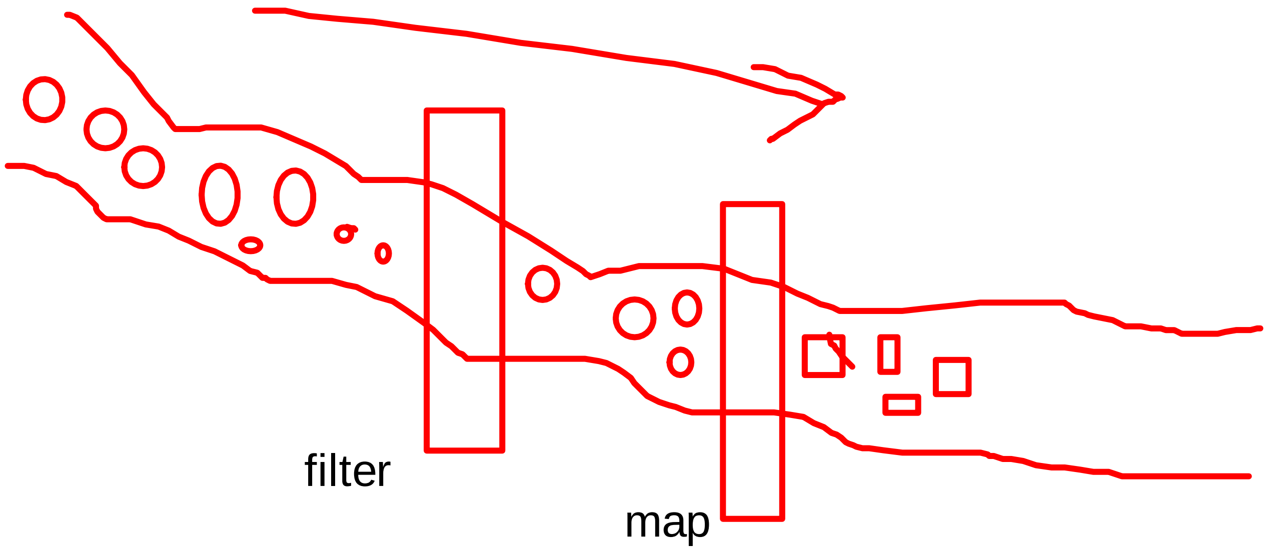
KEY HIGHLIGHTS

- ✓ Entire training programme is in English
- ✓ Interview prepration
- ✓ Provides Recording of each live session which you can access
- ✓ Provide complete notes / Hand-out, Handwritten for easy revision
- ✓ **no prerequisites**

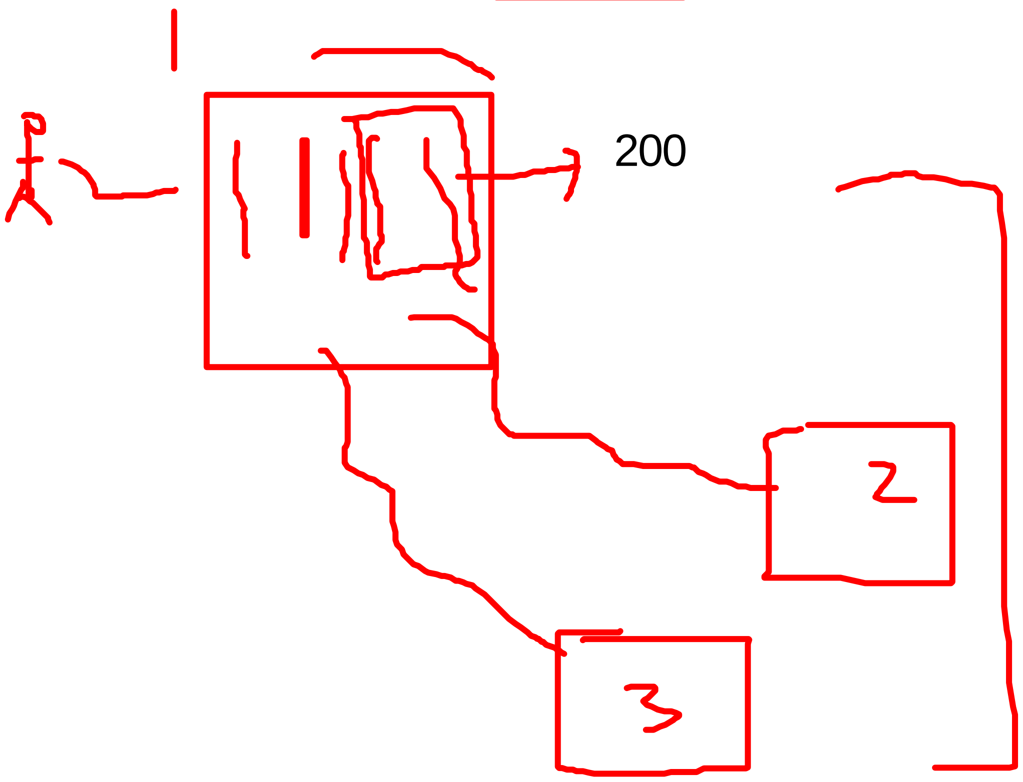
Core Java Master Course BUSYCODER ACADEMY	
Topic	**Topics Details**
✓ **Introduction to Java**	Introduction to Java, features of Java, IntelliJ community basicsJava Architecture, JDK, JRE, Array, Loops, dry run, examples, and practice
✓ **Object Orientation**	Introduction to Object-Oriented Principles, UML basics class, Objects declaration, Methods, Constructors and Destructors(Garbage Collections - Memory management), Passing parameters to methods(By value & By reference), Polymorphism - Method Overloading, Constructor Overloading, static keywords, var args, static import.
✓ **Inheritance, Overloading, Overriding**	The concept of inheritance, conversion, and casting in case of objects, override a method in a subclass, covariant returns, super keyword, Method overriding(Dynamic method dispatch), final keyword
Abstract class, Interface, Inner classes	Abstract keyword, Definition, implementation of an interface, using interface, interface and multiple inheritance, interface to share constants, importance of setting standards, marker interface Types of inner classes, Static and non-static inner classes, local and anonymous inner classes
String, Exception handling, IO	Object class, String class, StringBuffer, StringBuilder, Wrapper classes, primitive wrapper classes, immutability, autoboxing, Math class, The concept of exception handling, try blocks, Exception class methods, throw and rethrow, finally keyword, user-defined exceptions.Input/Output, Serialization

|| streams in java 8=> immutable, threads

Collections and Generics	Collections and Generics -- Collection framework, collection interfaces, and classes, For-each method for collection and iterators, classes implementing List interface Comparator, classes implementing Set interface, hashCode (), classes implementing Map interface, collections, and arrays classes, Queue interface, Creating our own LinkList, Stack, Queue, Heap, Tree Data Structure with Java
Multithreading	Multithreading, Introduction to Threads, creation for child threads, Thread methods, synchronization, inter-thread communication
Advance Multithreading	Create Worker threads using Callable and use an ExcecutorService to concurrently execute task. Using java.util.concurrent collection and classes including CyclicBarrier, CopyONWriteArrayList. Fork and Join framework. Parallel Streams inclusing reduction, decompostion, merging proesses, pipeline and performance. Concurrency Design patterns
Coding practice, GOF design patterns	Java coding conventions, coding practice, code refactoring, SOLID principle, Introduction to GOF design pattern some commonly used design pattern
Java 8	Introduction to Java 8, Why I should care for it, functional interface, Lambda expressions, Difference ann inner class vs lambda expression, Performance lambda, lambda under the hood, Passing code with behavior parameterization, introduction to stream processing, functional interfaces defined in Java 8, Working with streams details, Primitive stream specializations, Optionals Parallel data processing and performance, Collecting data with streams, Debugging and logging, Design pattern with java 8, Fork and join example, issues, Basics of Parallel streams, consideration and issues, gotchas parallel stream, Date/time API enhancement


filter
map

Where multithreading matter in real life?



Bulkhead dp

loan

dep

withraw

Threads : LWP

Mulithreading

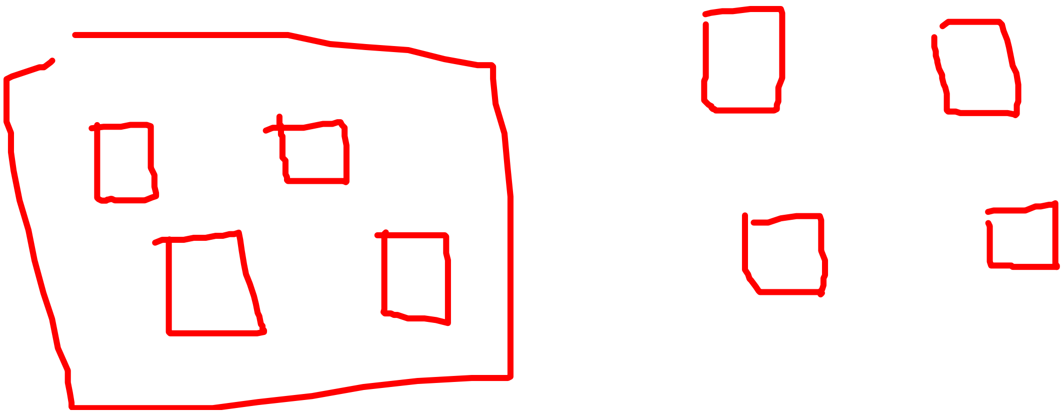
Concurrency || processing

java 4 classical threads

java 5 juc(java util concurrency)
Thread pool...

java 7 Fork and join :(

java 8 declartive Y ==> perellel streams



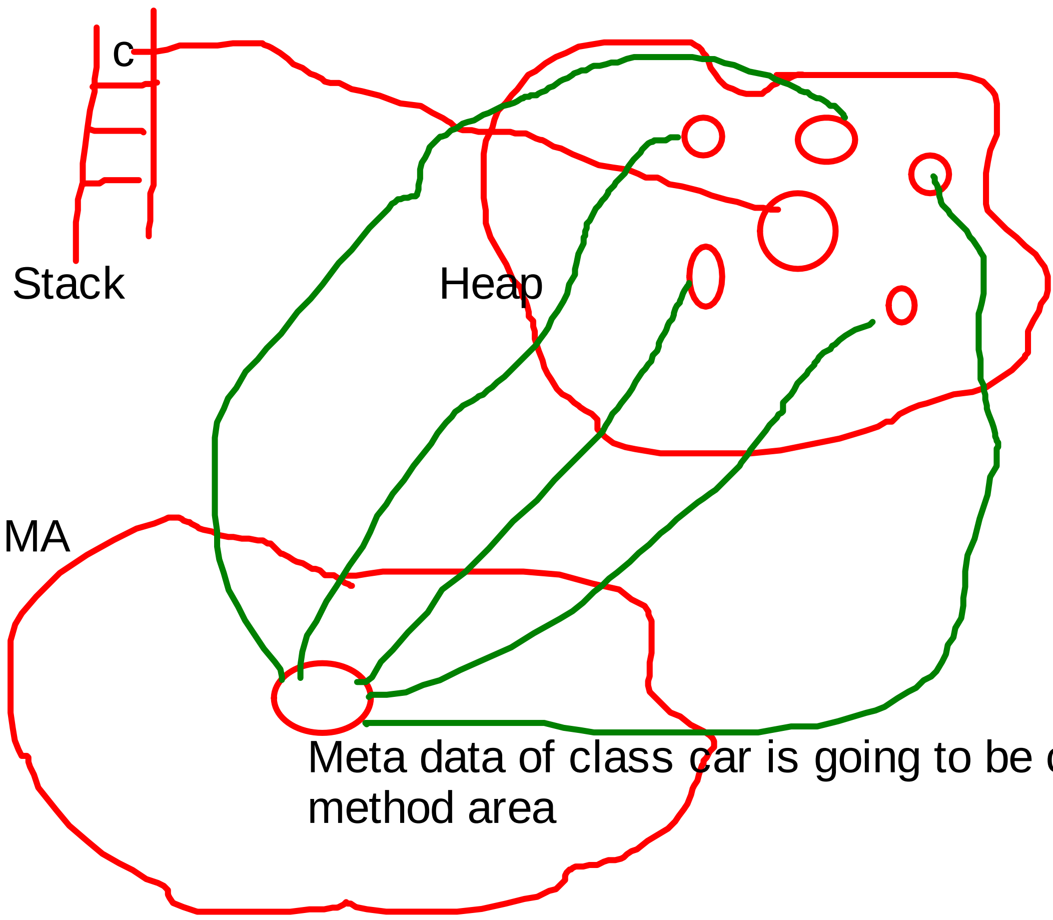
java 8

jdk
rt.jar70MB

Java 9 modularity? MS	Need of private Methods inside interface, Try with Resources Enhancements, In JDK 9, Usage of Diamond Operator extended to Anonymous classes also, SafeVarargs Annotation Enhancements, Factory Methods for creating unmodifiable Collections, Creation of unmodifiable Map (Immutable Map) with Java 9 Factory Methods, Java 9 Enhancements for Stream API, JShell, Java Platform Module System (JPMS), JLINK (Java Linker), Process API Updates (JEP-102), Java 9 HTTP/2 Client
Java 10	Time-Based Release Versioning,Local-Variable Type Inference,Experimental Java-Based JIT Compiler, Heap allocation on alternative memory devices, Application Class-Data Sharing, Parallel Full GC for G1
Java 11-20 15	Running Java File with a single command, New utility methods in String class, Local-Variable Syntax for Lambda Parameters, Nested Based Access Control, Switch Expressions, Pattern Matching for instanceOf, Text Blocks, New Methods in String Class for Text Blocks, Switch Expressions Enhancements, Helpful NullPointerExceptions, Records, Sealed Classes (Preview), Hidden Classes, Vector API, Foreign Function & Memory API, Virtual Threads, Structured Concurrency, ThreadGroup Degraded.
Java Application Performance Tuning and Memory Management	Introduction to JVM, Basic architecture, Memory management in JVM, Memory Leak, Metaspace and Internal JVM memory structure, JVM tuning, Heap monitoring, Analysing a heap dump, GC generational,GC Tuning, Profilers, Basics of JMH Benchmarking, GraalVM introduction

Car c=new Car();

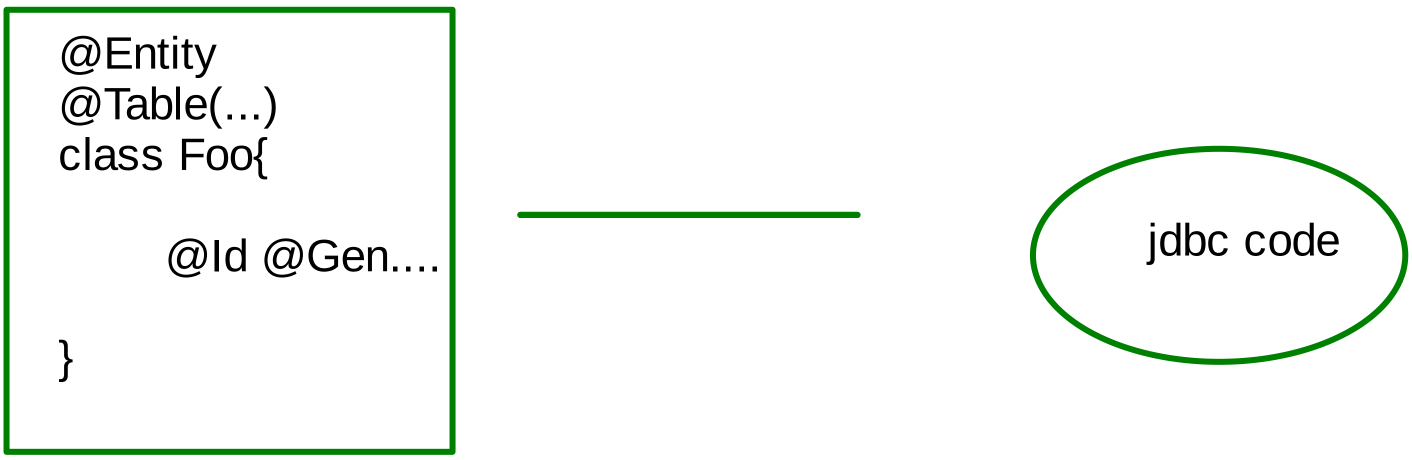
Java Reflection



Hib
Spring
AOP

Meta data of class car is going to be create on the method area

Any java based framewrork based on java refleaction



```
@Entity
@Table(...)
class Foo{

        @Id @Gen....

}
```

Byte code manipulation api

annotation processing

@Override
can u write ur own custion annotation :YES

1

Core
oops
ex handing
threads
collection

2

Adv threds

adv ds

solid, dp

3

java

enhancement

4

JVM, performace tuning
Banchmarking tools
JMX

5.
Networking prog
socket progmming
RMI

distrbuted computing

$$R = \underbrace{A + B \pm \Delta}$$

80/20

Corporate Client

- Bank Of America
- MakeMyTrip
- Edureka
- GreatLearning
- Hilti Asia Pacific
- Deloitte
- Kronos
- Yamaha Motors
- IBM
- Sapient
- Accenture
- Airtel
- Gemalto
- Cyient Ltd
- Fidelity Investment Ltd
- Blackrock
- Mahindra Comviva
- Iris Software
- SHL india
- Synechron pune
- harman
- Infosys
- Espire
- Steria
- Incedo
- Capgemini
- HCL
- CenturyLink
- Nucleus
- Ericsson
- Ivy Global
- Avaya
- NEC Technologies
- A.T. Kearney
- UST Global
- TCS
- North Shore Technologies
- Incedo
- Genpact
- Torry Harris
- Indian Air force
- Indian railways

Trainer's Profile

- Expert Java trainer MTech (Computer Science) with 18+ years experience in expertise in Java , OOAD, Design Patterns core, Spring Framework, Spring Boot, Spring Boot Microservice, Spring REST, Spring Data, Spring Security, Spring WS, Spring Mongo DB, DevOps with Java, Jenkin, Docker, Kubernetes, Messaging with Rabbit MQ, Kafka, Cloud AWS, GCP, Hibernate 5, EJB 3, Struts 1/2
- Helping technology organizations by training their fresh and senior engineers in key technologies and processes.
- Taught graduate and post-graduate academic courses to students with professional degrees.
- Conducted about 100 batches including fresher and lateral engineers.