

Java Fundamentals

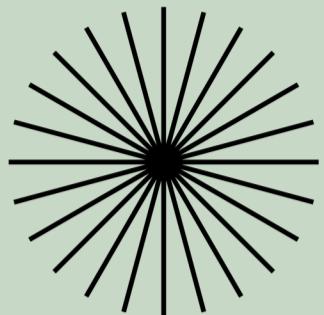


What Shouldn't You Ignore?

Swipe to see



Rohit Doshi
@rohitdoshi9

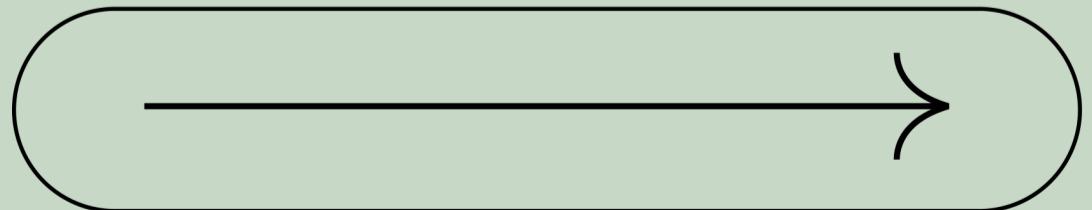


Whether you're a beginner or a seasoned coder, these timeless principles will pave the way for your Java journey.

Let's explore the key foundations you should never overlook.

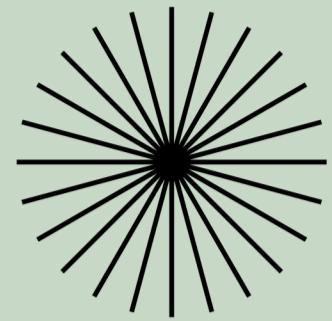


Rohit Doshi
@rohitdoshi9



01

Core Concepts



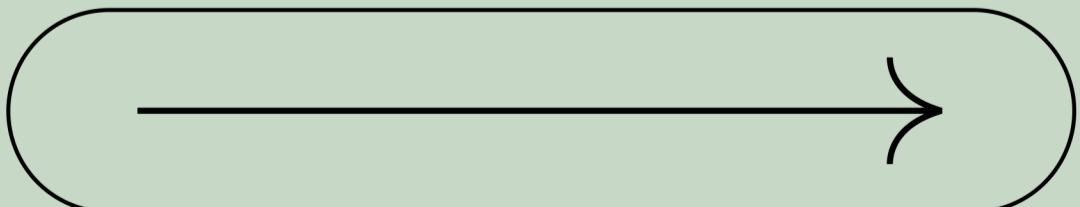
At the heart of Java are its core concepts.

Objects & Classes : Java is an object-oriented language. Master the art of creating classes and objects to model real-world entities.

Inheritance & Polymorphism : Understand how classes can inherit attributes and behaviors from others, fostering code reusability and flexibility.

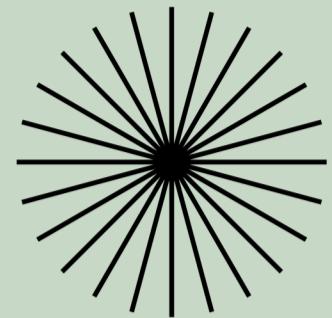


Rohit Doshi
@rohitdoshi9



02

Core Concepts



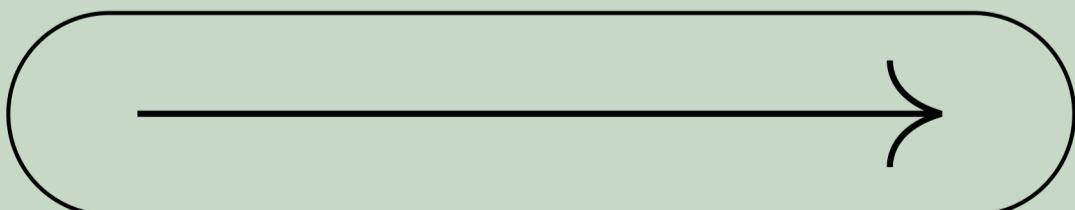
At the heart of Java are its core concepts.

Abstraction & Encapsulation : Hide complex implementations and expose only necessary details, promoting modular and maintainable code.

Interfaces & Implementations : Leverage interfaces to define contracts that classes must adhere to, enabling multiple inheritance-like behavior.

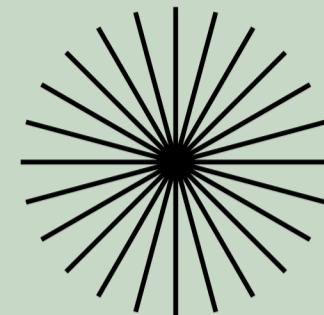


Rohit Doshi
@rohitdoshi9



03

Memory Management

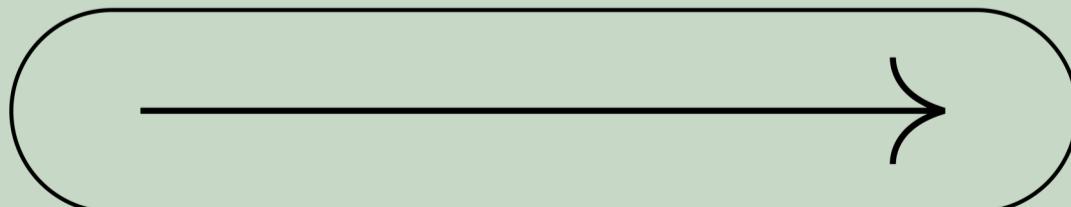


Garbage Collection : Java automates memory management through garbage collection. Learn how it deallocates memory from unused objects, ensuring efficient resource utilization.

Memory Leaks : Beware of unintentional memory retention, causing memory leaks. Properly release resources to prevent long-term performance issues.

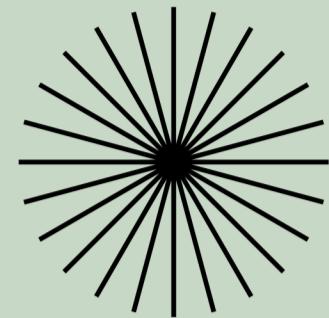


Rohit Doshi
@rohitdoshi9



04

Exception Handling

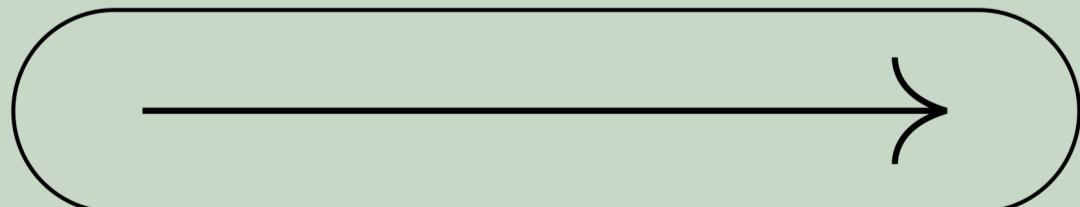


Try-Catch Blocks : Grasp the art of handling exceptions gracefully. Employ try-catch blocks to catch and handle errors without crashing your program.

Checked vs. Unchecked : Differentiate between checked (compile-time) and unchecked (runtime) exceptions. Handle or propagate them appropriately.

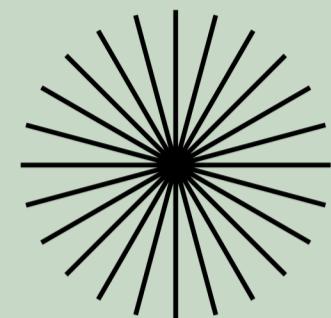


Rohit Doshi
@rohitdoshi9



05

Collections Framework

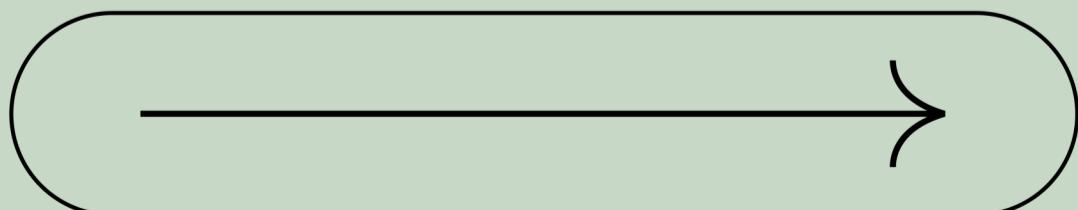


Lists, Sets, Maps : Explore the diverse collection interfaces and implementations for storing, organizing, and manipulating data.

Iterators : Learn to traverse collections safely and efficiently using iterators, preventing data corruption.

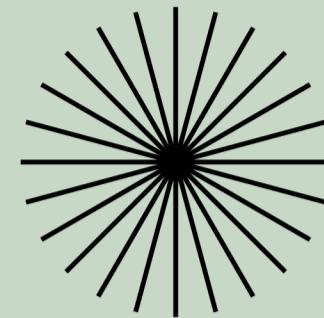


Rohit Doshi
@rohitdoshi9



06

Multithreading

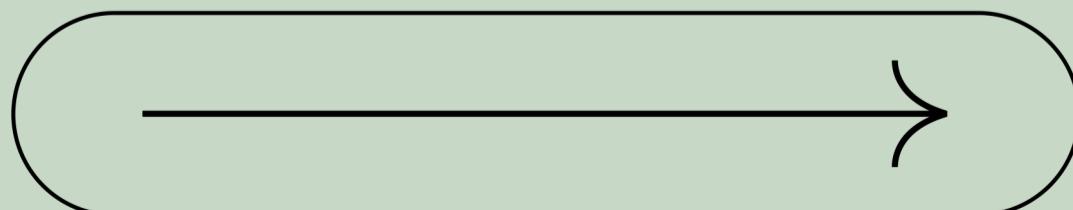


Concurrency : Java's multithreading capabilities enable efficient execution of tasks simultaneously. Understand synchronization to avoid race conditions.

Thread Pools : Implement thread pools to manage resources effectively and enhance performance in multi-threaded applications.

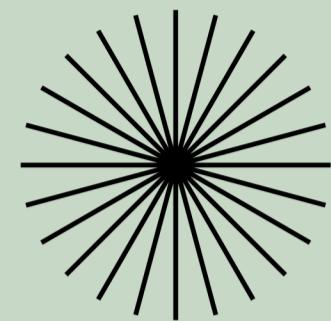


Rohit Doshi
@rohitdoshi9



07

Input/Output Streams

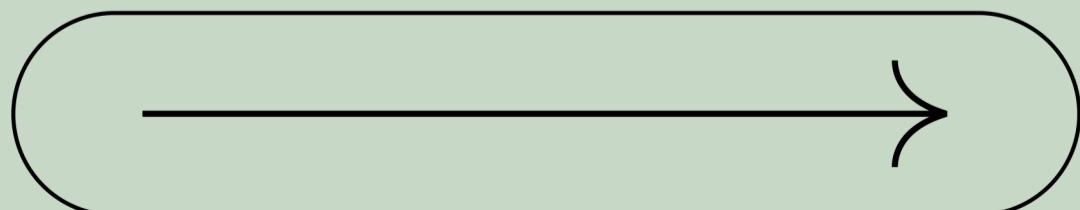


File Handling : Master reading from and writing to files using input and output streams, ensuring data persistence.

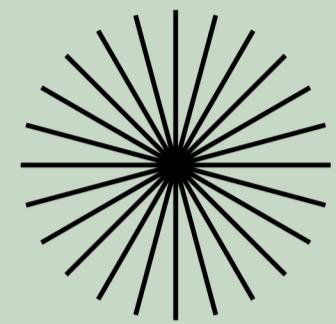
Serialization : Explore object serialization to convert objects into byte streams, facilitating data storage and transfer.



Rohit Doshi
@rohitdoshi9



These fundamental principles lay the groundwork for your Java excellence.

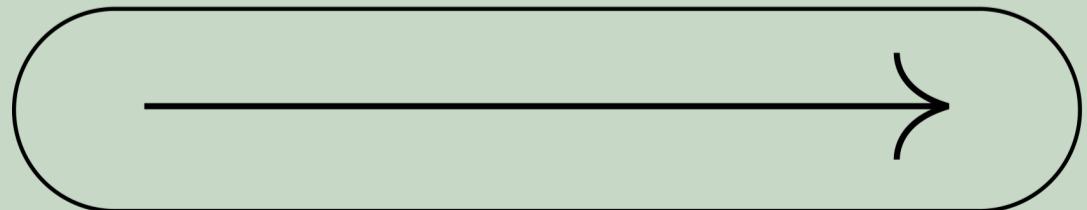


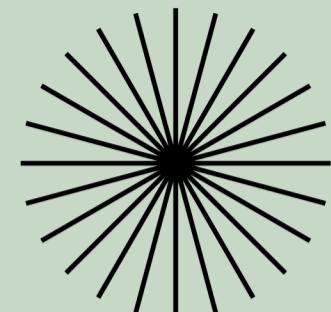
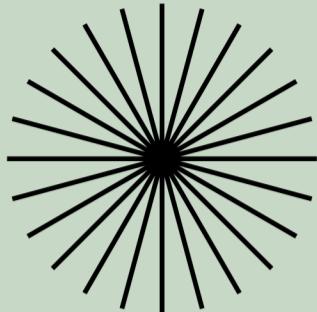
Remember, a strong understanding of these concepts will not only make you a better coder but also open doors to more complex Java functionalities.

Keep coding, keep learning!



Rohit Doshi
@rohitdoshi9



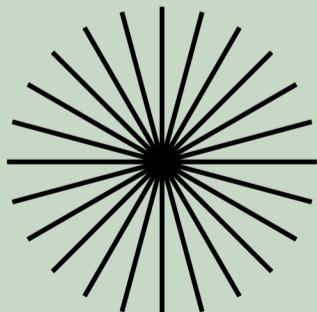


Did you find the post
helpful?

Checkout my LinkedIn
profile for more such
insightful reads!

Like! Comment ! Repost!

It truly helps!



Rohit Doshi
@rohitdoshi9

