# Java Developer Notes

## 1. Introduction

This document serves as a structured reference guide for Java developers. It includes key concepts, code examples, and references for easy navigation.

## 2. Key Concepts

Below are some of the essential concepts every Java developer should be familiar with:

|  |  |
| --- | --- |
| Concept | Description |
| OOP Principles | Encapsulation, Abstraction, Inheritance, Polymorphism |
| Collections Framework | List, Set, Map, Queue, Stack |
| Concurrency | Threads, Executors, Synchronization, Locks |
| Spring Framework | Dependency Injection, AOP, Transactions |
| Microservices | REST APIs, Circuit Breakers, Service Discovery |
| Java 8 Features | Lambdas, Streams, Optional, Functional Interfaces |

## 3. Code Examples

Here are some useful Java code snippets for reference:

### Example: Lambda Expression

@FunctionalInterface  
interface MathOperation {  
 int operation(int a, int b);  
}  
  
public class LambdaExample {  
 public static void main(String[] args) {  
 MathOperation addition = (a, b) -> a + b;  
 System.out.println(addition.operation(5, 10));  
 }  
}

## 4. References

This section contains references for further reading and official documentation:

|  |  |
| --- | --- |
| Reference | Link |
| Java Documentation | https://docs.oracle.com/en/java/ |
| Spring Framework | https://spring.io/ |
| Effective Java (Book) | https://www.oreilly.com/library/view/effective-java-3rd/9780134686097/ |
| Microservices with Spring | https://spring.io/microservices |
| Concurrency in Java | https://www.baeldung.com/java-concurrency |