

©2016 Capgemini. All rights reserved.

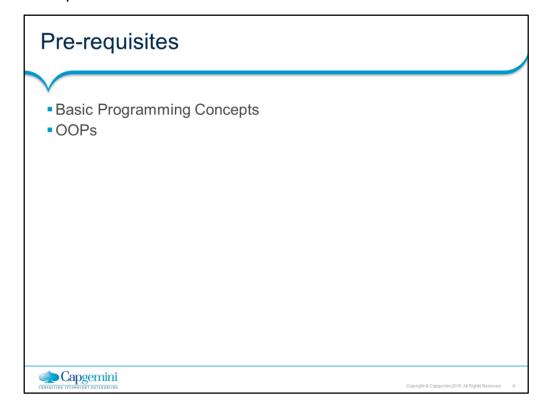
The information contained in this document is proprietary and confidential. For Capgemini only.

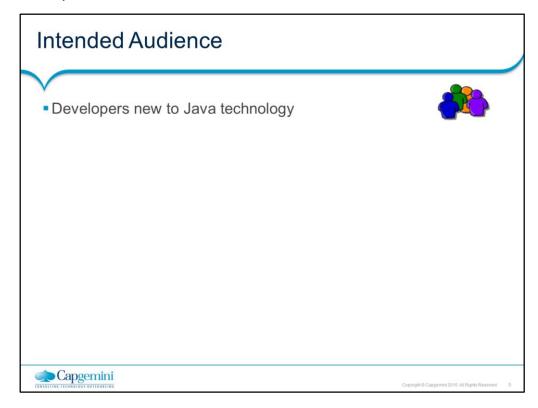
### Course Goals and Non Goals

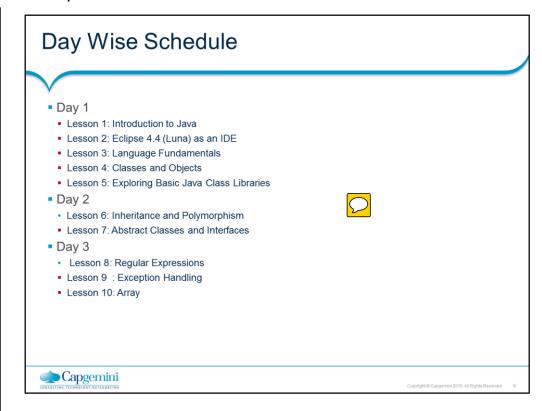
- ➤ Course Goals
  - Implementing OOPs features in Java
  - Developing Java Desktop Applications
  - Use of Core JDK 1.8 API
  - Testing using Junit 4
- ➤ Course Non Goals
  - Developing GUI applications











# Day Wise Schedule Day 4 Lesson 11: Collection Lesson 12: File IO Day 5 Lesson 13: Introduction to JUnit 4 /Test NG Lesion 14: Advanced Testing Concepts

- Lesson 1: Introduction to Java
  - 1.1: Introduction to Java
  - 1.2: Features of Java
  - 1.3: Simple Program in Java
  - 1.4: Developing software in Java
- Lesson 2: Eclipse 4.4 (Luna) as an IDE
  - 2.1: Installation and Setting up Eclipse
  - 2.2: Introduction to Eclipse IDE
  - 2.3: Creating and Managing Java Projects
  - 2.4: Miscellaneous Options



- Lesson 3: Language Fundamentals
  - 3.1: Keywords
  - 3.2: Primitive Data Types
  - 3.3: Operators and Assignments
  - 3.4: Variables and Literals
  - 3.5: Flow Control: Java's Control Statements
  - 3.6: Best Practices
- Lesson 4: Classes and Objects
  - 4.1: Classes and Objects
  - 4.2: Packages
  - 4.3: Access Specifies
  - 4.4: Constructors Default and Parameterized
  - 4.5: this reference
  - 4.6 using static keyword
  - 4.7: Best Practices



- Lesson 5: Exploring Basic Java Class Libraries
  - 5.1: The Object Class
  - 5.2: Wrapper Classes
  - 5.3: Type casting
  - 5.4: Using Scanner Class
  - 5.5: String Handling
  - 5.6: Date and Time API
  - 5.7: Best Practices
- Lesson 6: Inheritance and Polymorphism
  - 6.1: Inheritance
  - 6.2: Using super keyword
  - 6.3: Instance Of Operator
  - 6.4: Method & Constructor overloading
  - 6.5: Method overriding
  - 6.6: @override annotation
  - 6.7: Using final keyword
  - 6.8 Best Practices



- Lesson 7: Abstract Classes and Interfaces
  - 7.1: Abstract class
  - 7.2: Interfaces
  - 7.3: default methods
  - 7.4: static methods on Interface
  - 7.5 : Runtime Polymorphism
- Lesson 8: Regular Expressions
  - 8.1: Regular Expressions
  - 8.2: Validating data
  - 8.3: Best Practices



- Lesson 9: Exception Handling
  - 9.1: Introduction
  - 9.2: Exception Types
  - 9.3: Exception Hierarchy
  - 9.4: Try-catch-finally
  - 9.5: Try-with-resources
  - 9.6: Multi catch blocks
  - 9.7: Throwing exceptions using throw
  - 9.8: Declaring exceptions using throws
  - 9.9: User defined Exceptions
  - 9.10: Best Practices



Copyright © Capgemini 2015. All Rights Reserved

Page 00-12

- Lesson 10: Array
- 10.1: One dimensional array
- 10.2: Multidimensional array
- 10.3: Using varargs
- 10.4: Using Arrays class
- Lesson 11: Collection
  - 11.1: Collections Framework
  - 11.2: Collection Interfaces
  - 11.3: Implementing Classes
  - 11.4 Iterating Collections
  - 11.5: Best Practices



### Lesson 12: File IO

- 12.1: Overview of I/O Streams
- 12.2: Types of Streams
- 12.3: The Byte-stream I/O hierarchy
- 12.4: Character Stream Hierarchy
- 12.5: Buffered Stream
- 12.6: The File class
- 12.7: The Path Interface
- 12.8: Object Stream
- 12.9: Best Practices
- Lesson 13: Introduction to Junit 4
  - 13.1: Introduction
  - 13.2: JUnit
  - 13.3: Installing and Running JUnit
  - 13.4: Testing with JUnit
  - 13.5: Testing Exceptions
  - 13.6: Test Fixtures
  - 13.7: Best Practices



opyright © Capgemini 2015. All Rights Reserved

Page 00-14

# ■ Lesson 14: Advanced Testing Concepts ■ 14.1: Advanced Testing concepts ■ 14.2: Test Suites ■ 14.3: Parameterized Tests ■ 14.4: Mocking Concepts ■ 14.4: Mocking Concepts

### References

- Books:
  - Java, The Complete Reference; by Herbert Schildt
  - Thinking in Java; by Bruce Eckel
  - Beginning Java 8 Fundamentals by Kishori Sharan
- Websites:
  - Java home page: http://java.sun.com/
  - JDK 1.8 documentation: http://docs.oracle.com/javase/8/docs/





