Codeelse ;
LEARNING CENTRE

CORE & ADVANCED JAVA

Course Outline

Chapter 1: Introduction to Java Programming with Jshell using Multiplication Table

00 - Getting Started with Programming
01 - Introduction to Multiplication Table challenge
02 - Launch JShell
03 - Break Down Multiplication Table Challenge
04 - Java Expression - An Introduction
05 - Java Expression - Exercises
06 - Java Expression - Puzzles
07 - Printing output to console with Java
08 - Printing output to console with Java - Exercise Statements
09 - Printing output to console with Java - Exercise Solutions
10 - Printing output to console with Java - Puzzles
11 - Advanced Printing output to console with Java
12 - Advanced Printing output to console with Java - Exercises and Puzzles
13 - Introduction to Variables in Java
14 - Introduction to Variables in Java - Exercises and Puzzles
15 - 4 Important Things to Know about Variables in Java
16 - How are variables stored in memory?
17 - How to name a variable?
18 - Understanding Primitive Variable Types in Java
19 - Understanding Primitive Variable Types in Java - Choosing a Type
20 - Java Assignment Operator
21 - Java Assignment Operator - Puzzles on Increment, Decrement and Compound Assignment
23 - Java Conditionals and If Statement - Introduction
24 - Java Conditionals and If Statement - Exercise Statements
25 - Java Conditionals and If Statement - Exercise Solutions
26 - Java Conditionals and If Statement - Puzzles
27 - Java For Loop to Print Multiplication Table - Introduction
28 - Java For Loop to Print Multiplication Table - Exercise Statements
29 - Java For Loop to Print Multiplication Table - Exercise Solutions
30 - Java For Loop to Print Multiplication Table - Puzzles

Chapter 2: Introduction to Method with Multiplication Table

Chapter 3: Introduction to Java Platform

Chapter 4: Introduction to Eclipse - First Java Project

Chapter 5: Introduction to Object-Oriented Programming

00 - Introduction to Object-Oriented Programming - Section Overview
01 - Introduction to Object-Oriented Programming - Basics
02 - Introduction to Object-Oriented Programming - Terminology - Class, Object, State and Behavior
03 - Introduction to Object-Oriented Programming - Exercise - Online Shopping System and Person
04 - Create a Motor Bike Java Class and a couple of objects
05 - Exercise Solutions - Book class and Three instances
06 - Introducing State of an object with speed variable
07 - Understanding the basics of Encapsulation with Setter methods
08 - Exercises and Tips - Getters and Generating Getters and Setters with Eclipse
09 - Puzzles on this and initialization of member variables
10 - First Advantage of Encapsulation
11 - Introduction to Encapsulation - Level 2
12 - Encapsulation Exercises - Better Validation and Book class
13 - Introduction to Abstraction
14 - Introduction to Java Constructors
15 - Introduction to Java Constructors - Exercises and Puzzles
16 - Introduction to Object-Oriented Programming - Conclusion

Chapter 6: Primitive Data Types And Alternatives

00 - Primitive Data Types in Depth - Section Overview
01 - Basics about Java Integer Data Types - Casting, Operators and More
02 - Java Integer Data Types - Puzzles - Octal, Hexadecimal, Post and Pre increment
03 - Java Integer Data Types - Exercises - BiNumber - add, multiply and double
04 - Java Floating-Point Data Types - Casting, Conversion and Accuracy
05 - Introduction to BigDecimal Java Class
06 - BigDecimal Puzzles - Adding Integers
07 - BigDecimal Exercises - Simple Interest Calculation
08 - Java Boolean Data Type - Relational and Logical Operators
09 - Java Boolean Data Type - Puzzles - Short Circuit Operators
10 - Java Character Data Type char - Representation and Conversion
11 - Java char Data Type - Exercises 1 - isVowel
12 - Java char Data Type - Exercises 2 - isDigit
13 - Java char Data Type - Exercises 3 - isConsonant, List Upper Case and Lower Case Characters
14 - Primitive Data Types in Depth - Conclusion

Chapter 7: Conditionals

00 - Conditionals with Java - Section Overview

01 - Introduction to If Else Statement

02 - Introduction to Nested If Else

03 - If Else Statement - Puzzles

04 - If Else Problem - How to get User Input in Java?

05 - If Else Problem - How to get number 2 and choice from the user?

06 - If Else Problem - Implementing with Nested If Else

07 - Java Switch Statement - An introduction

08 - Java Switch Statement - Puzzles - Default, Break and Fall Through

09 - Java Switch Statement - Exercises - isWeekDay, nameOfMonth, nameOfDay

10 - Java Ternary Operation - An Introduction

11 - Conditionals with Java - Conclusion

Chapter 8: Loops

00 - Java Loops - Section Introduction

01 - Java For Loop - Syntax and Puzzles

02 - Java For Loop - Exercises Overview and First Exercise Prime Numbers

03 - Java For Loop - Exercise - Sum Upto N Numbers and Sum of Divisors

04 - Java For Loop - Exercise - Print a Number Triangle

05 - While Loop in Java - An Introduction

06 - While Loop - Exercises - Cubes and Squares up to limit

07 - Do While Loop in Java - An Introduction

08 - Do While Loop in Java - An Example - Cube while the user enters positive numbers

09 - Introduction to Break and Continue

10 - Selecting Loop in Java - For vs While vs Do While

Chapter 9: Reference Types

00 - Java Reference Types - Section Introduction

01 - Reference Types - How are they stored in Memory?

02 - Java Reference Types - Puzzles

03 - String class - Introduction and Exercise - Print each word and char on a new line

04 - String class - Exercise Solution and Some More Important Methods

05 - Understanding String is Immutable and String Concat, Upper Case, Lower Case, Trim methods

06 - String Concatenation and Join, Replace Methods

07 - Java String Alternatives - StringBuffer and StringBuilder

08 - Java Wrapper Classes - An Introduction - Why and What?

09 - Java Wrapper Classes - Creation - Constructor and valueOf

10 - Java Wrapper Classes - Auto Boxing and a Few Wrapper Constants - SIZE, BYTES, MAX_VALUE and MIN_VALUE

11 - Java Dates - Introduction to LocalDate, LocalTime and LocalDateTime
12 - Java Dates - Exploring LocalDate - Creation and Methods to play with Date
13 - Java Dates - Exploring LocalDate - Comparing Dates and Creating Specific Dates
14 - Java Reference Types - Conclusion

Chapter 10: Arrays and ArrayLists

00 - Introduction to Array and ArrayList - Section Introduction with a Challenge
01 - Understanding the need and Basics about an Array
02 - Java Arrays - Creating and Accessing Values - Introduction
03 - Java Arrays - Puzzles - Arrays of Objects, Primitive Data Types, toString and Exceptions
04 - Java Arrays - Compare, Sort and Fill
05 - Java Arrays - Exercise - Create Student Class - Part 1 - Total and Average Marks
06 - Java Arrays - Exercise - Create Student Class - Part 2 - Maximum and Minimum Mark
07 - Introduction to Variable Arguments - Need
08 - Introduction to Variable Arguments - Basics
09 - Introduction to Variable Arguments - Enhancing Student Class
10 - Java Arrays - Using Person Objects and String Elements with Exercises
11 - Java String Arrays - Exercise Solutions - Print Day of Week with the Most number of letters and more
12 - Adding and Removing Marks - Problem with Arrays
13 - First Look at ArrayList - An Introduction
14 - First Look at ArrayList - Refactoring Student Class to use ArrayList
15 - First Look at ArrayList - Enhancing Student Class with Add and Remove Marks
16 - Introduction to Array and ArrayList - Conclusion

Chapter 11: Object-Oriented Programming Again

00 - Object-Oriented Programming - Level 2 - Section Introduction
01 - Basics of Designing a Class - Class, Object, State and Behavior
02 - OOPS Example - Fan Class - Deciding State and Constructors
03 - OOPS Example - Fan Class - Deciding Behavior with Methods
04 - OOPS Exercise - Rectangle Class
05 - Understanding Object Composition with Customer Address Example
06 - Understanding Object Composition - An Exercise - Books and Reviews
07 - Understanding Inheritance - Why do we need it?
08 - Object is at top of Inheritance Hierarchy
09 - Inheritance and Overriding - with toString() method
10 - Java Inheritance - Exercise - Student and Employee Classes
11 - Java Inheritance - Default Constructors and super() method call
12 - Java Inheritance - Puzzles - Multiple Inheritance, Reference Variables and instanceof
13 - Java Abstract Class - Introduction
14 - Java Abstract Class - First Example - Creating Recipes with Template Method

15 - Java Abstract Class - Puzzles
16 - Java Interface - Example 1 - Gaming Console - How to think about Interfaces?
17 - Java Interface - Example 2 - Complex Algorithm - API defined by an external team
18 - Java Interface - Puzzles - Unimplemented methods, Abstract Classes, Variables, Default Methods and more
19 - Java Interface vs Abstract Class - A Comparison
20 - Java Interface Flyable and Abstract Class Animal - An Exercise
21 - Polymorphism - An introduction

Chapter 12: Collections

01 - Java Collections - Section Overview with Need For Collections
02 - List Interface - Introduction - Position is King
03 - List Interface - Immutability and Introduction of Implementations - ArrayList, LinkedList and Vector
04 - List Interface Implementations - ArrayList vs LinkedList
05 - List Interface Implementations - ArrayList vs Vector
06 - List Interface - Methods to add, remove and change elements and lists
07 - List and ArrayList - Iterating around elements
08 - List and ArrayList - Choosing iteration approach for printing and deleting elements
09 - List and ArrayList - Puzzles - Type Safety and Removing Integers
10 - List and ArrayList - Sorting - Introduction to Collections sort static method
11 - List and ArrayList - Sorting - Implementing Comparable Interface in Student Class
12 - List and ArrayList - Sorting - Providing Flexibility by implementing Comparator interface
13 - List and ArrayList - A Summary
14 - Set Interface - Introduction - No Duplication
15 - Understanding Data Structures - Array, LinkedList and Hashing
16 - Understanding Data Structures - Tree - Sorted Order
17 - Set Interface - Hands-on - HashSet, LinkedHashSet and TreeSet
18 - Set Interface - Exercise - Find Unique Characters in a List
19 - TreeSet - Methods from NavigableSet - floor,lower,upper, subSet, head and tailSet
20 - Queue Interface - Process Elements in Order
21 - Introduction to PriorityQueue - Basic Methods and Customized Priority
22 - Map Interface - An Introduction - Key and Value
23 - Map Interface - Implementations - HashMap, HashTable, LinkedHashMap and TreeMap
24 - Map Interface - Basic Operations
25 - Map Interface - Comparison - HashMap vs LinkedHashMap vs TreeMap
26 - Map Interface - Exercise - Count occurrences of characters and words in a piece of text
27 - TreeMap - Methods from NavigableMap - floorKey, higherKey, firstEntry, subMap and more
28 - Java Collections - Conclusion with Three Tips

Chapter 13: Generics

01 - Introduction to Generics - Why do we need Generics?

02 - Implementing Generics for the Custom List

03 - Extending Custom List with a Generic Return Method

04 - Generics Puzzles - Restrictions with extends and Generic Methods

05 - Generics and WildCards - Upper Bound and Lower Bound

Chapter 14: Introduction to Functional Programming

01 - Introduction to Functional Programming - Functions are First-Class Citizens

02 - Functional Programming - First Example with Function as Parameter

03 - Functional Programming - Exercise - Loop a List of Numbers

04 - Functional Programming - Filtering - Exercises to print odd and even numbers from List

05 - Functional Programming - Collect - Sum of Numbers in a List

06 - Functional Programming vs Structural Programming - A Quick Comparison

07 - Functional Programming Terminology - Lambda Expression, Stream and Operations on a Stream

08 - Stream Intermediate Operations - Sort, Distinct, Filter and Map

09 - Stream Intermediate Operations - Exercises - Squares of First 10, Map String List to LowerCase and Length of String

10 - Stream Terminal Operations - 1 - max operation with Comparator

11 - Stream Terminal Operations - 2 - min, collect to List,

12 - Optional class in Java - An Introduction

13 - Behind the Screens with Functional Interfaces - Implement Predicate Interface

14 - Behind the Screens with Functional Interfaces - Implement Consumer Interface

15 - Behind the Screens with Functional Interfaces - Implement Function Interface for Mapping

16 - Simplify Functional Programming code with Method References - static and instance methods

17 - Functions are First-Class Citizens

18 - Introduction to Functional Programming - Conclusion

Chapter 15: Introduction to Threads And Concurrency

01 - Introduction to Threads and MultiThreading - Need for Threads

02 - Creating a Thread for Task1 - Extending Thread Class

03 - Creating a Thread for Task2 - Implement Runnable Interface

04 - Theory - States of a Thread

05 - Placing Priority Requests for Threads

06 - Communication between Threads - join method

07 - Thread utility methods and synchronized keyword - sleep, yield

08 - Need for Controlling the Execution of Threads

09 - Introduction to Executor Service

10 - Executor Service - Customizing number of Threads

11 - Executor Service - Returning a Future from Thread using Callable

12 - Executor Service - Waiting for completion of multiple tasks using invokeAll

13 - Executor Service - Wait for only the fastest task using invokeAny

14 - Threads and MultiThreading - Conclusion

Chapter 16: Introduction to Exception Handling

01 - Introduction to Exception Handling - Your Thought Process during Exception Handling

02 - Basics of Exceptions - NullPointerException and StackTrace

03 - Basics of Handling Exceptions - try and catch

04 - Basics of Handling Exceptions - Exception Hierarchy, Matching and Catching Multiple Exceptions

05 - Basics of Handling Exceptions - Need for finally

06 - Basics of Handling Exceptions - Puzzles

07 - Checked Exceptions vs Unchecked Exceptions - An Example

08 - Hierarchy of Errors and Exceptions - Checked and Runtime

09 - Throwing an Exception - Currencies Do Not Match Runtime Exception

10 - Throwing a Checked Exception - Throws in method signature and handling

11 - Throwing a Custom Exception - CurrenciesDoNotMatchException

12 - write less code with Try with Resources - New Feature in Java 7

13 - Basics of Handling Exceptions - Puzzles 2

14 - Exception Handling - Conclusion with Best Practices

Chapter 17: Files and Directories

01 - List files and folders in Directory with the Files list method

02 - Recursively List and Filter all files and folders in Directory with the Files walk method and Search with find method

03 - Read content from a File - Files readAllLines and lines methods

04 - Writing Content to a File - Files write method

05 - Files - Conclusion

Chapter 18: More Concurrency with Concurrent Collections and Atomic Operations

01 - Getting started with Synchronized

02 - Problem with Synchronized - Less Concurrency

03 - Enter Locks with ReEntrantLock

04 - Introduction to Atomic Classes - AtomicInteger

05 - Need for ConcurrentMap

06 - Implementing an example with ConcurrentHashMap

07 - ConcurrentHashMap uses different locks for diferrent regions

08 - CopyOnWrite Concurrent Collections - When reads are more than writes

09 - Conclusion

Chapter 19: Java Tips

Java Tip 01 - Imports and Static Imports

Java Tip 02 - Blocks

Java Tip 03 - equals method

Java Tip 04 - hashcode method

Java Tip 05 - Class Access Modifiers - public and default

Java Tip 06 - Method Access Modifiers - public, protected, private and default

Java Tip 07 - Final classes and Final methods

Java Tip 08 - Final Variables and Final Arguments

Java Tip 09 - Why do we need static variables?

Java Tip 09 - Why do we need static methods?

Java Tip 10 - Static methods cannot use instance methods or variables

Java Tip 11 - public static final - Constants

Java Tip 12 - Nested Classes - Inner Class vs Static Nested Class

Java Tip 13 - Anonymous Classes

Java Tip 14 - Why Enum and Enum Basics - ordinal and values

Java Tip 15 - Enum - Constructor, variables and methods

Java Tip 16 - Quick look at inbuild Enums - Month, DayOfWeek