

Introduction to Java SE

The programming language for this course is Java SE (version 5.0).

The first part of the course (chapters 1-9) deals with basics of Java programming and the second part (chapters 10-18) introduces students to concepts of object-oriented programming and some features released in Java SE version 5.0.

Theory includes comprehensive examples of subject being studied. Chapters 4-8 and 11 have an example program at the end of the chapter where students can practice what they have learned about Java so far. Chapter 9, on the other hand, focuses completely on designing, implementing and testing a tic-tac-toe game.

Introduction to Java SE includes 47 programming exercises and 124 multiple-choice questions. Finishing this course takes about 100-160 hours.

Table of contents:

1 Foreword 1.1 Course introduction 1.2 Tips for studying 1.3 Computer and computer programs 1.4 From source code to a form the computer understands	2 Algorithm 2.1 From problem to algorithm 2.2 From algorithm to source code	3 Introduction to Java 3.1 Java-language in brief 3.2 Instructions on independent studying 3.3 Parts and structure of the source code
4 Types, variables, statements & blocks 4.1 Types and variables 4.2 Statements and expressions 4.3 Small Java-program on things learned so far	5 Printing, reading, operators 5.1 Class and method in brief 5.2 Printing data 5.3 Reading keyboard input with Scanner-class tools 5.4 Operators 5.5 Small Java-program on things learned so far	6 Control flow statements 6.1 Blocks 6.2 Conditional statements 6.3 Loop statements 6.4 Small Java-program on things learned so far
7 Methods 7.1 Declaring methods 7.2 Calling a method 7.3 Passing parameters 7.4 Overloading methods 7.5 Small Java-program on things learned so far	8 Arrays 8.1 Applying arrays 8.2 Array in use 8.3 Copying arrays 8.4 Multidimensional arrays 8.5 Small Java-program on things learned so far	9 Making programs 9.1 Tic-tac-toe: analyzing the problem 9.2 Tic-tac-toe: designing algorithm 9.3 Tic-tac-toe: coding 9.4 Tic-tac-toe: testing
10 Objects in problem solving 10.1 Procedural programming vs. object-oriented programming 10.2 Object 10.3 Class 10.4 Data hiding and encapsulation 10.5 Abstraction 10.6 Inheritance	11 Class, model of an object 11.1 Class and how it is done 11.2 Object from a class 11.3 Testing the class 11.4 Storing objects in arrays 11.5 String-class methods 11.6 Small Java-program on things learned so far	12 Class members and scope 12.1 Class members 12.2 Access control modifiers
13 Inheritance 13.1 Introduction to inheritance 13.2 Declaration of a subclass 13.3 Polymorphism 13.4 Abstract classes 13.5 Object	14 Interfaces 14.1 Introduction 14.2 Using an interface class	15 Packages 15.1 What is a package? 15.2 Using default packages 15.3 Declaring new packages 15.4 How does the compiler find the packages? 15.5 Package scopes
16 Exceptions and handling them 16.1 What is an exception? 16.2 Exception classes 16.3 Throwing an exception 16.4 Handling exceptions	17 Data streams and files 17.1 Concepts 17.2 Reading and writing information in text format 17.3 File handling with tools from java.io.File	18 Generic stacks 18.1 Introduction to stack example 18.2 Dynamic genericity with Object class 18.3 Glimpse at Java parameterized class models