Learn To Program Java

Synopsis:

This course follows a "fundamentals-first" approach to teaching Java. It teaches Java by first providing fundamental concepts of Java and then building on them as more difficult concepts are introduced in the later part of the course. The "fundamentals-first" approach is reflected in the title of the course as "Learn to Program". Participants will be taught programming fundamentals such as variables, data structures, operators, and program compilation and execution. The second component of the course covers the teaching of Java as a programming language: program structures; data types and declaration; expressions, statements, and operators; control flow and arrays; inputs and outputs; methods; object-oriented programming; encapsulation; and polymorphism. Besides, the course also covers the Java Application Programming Interface (API) – a rich Java class library. Case studies and exercise questions are included in the accompanying book each participant will receive.

Who should attend?

Anyone who wants to learn how to create computer programs using Java.

Objectives

1. Define computer programming
2. Learn about program compilation and execution
3. Learn how to use Java to write computer programs
4. Learn the concepts of object-oriented programming
5. Learn how to produce object-oriented programs using Java

Prerequisites

None. *It assumes participants have little or no knowledge on how to write software programs.*

Duration

4 days

Course Outline:

1. **BEGINNING PROGRAMMING**
   Define program compilation and interpretation
   Explain how to compile and run a Java program
   Explain common mistakes in Java programming
   Explain programming objectives
   Practise running pre-written Java programs

2. **BASIC BUILDING BLOCKS**
   Describe data type and variable
   Describe identifier and reserved words
   Describe comments and the basic program structure
   Practise writing Java program with basic building blocks

3. **EXPRESSION, STATEMENTS, OPERATORS**
   Define expressions and statements
   Explain various types of operators
   Discuss operator precedence and associativity
   Practise use of expressions, statements and operators

4. **CONTROL FLOW AND ARRAYS**
   Define the three types of control flow: Sequence, Iterations, Selection
   Explain sequence in statements
   Discuss the use of if..else and switch statements
   Discuss the use of while, do-while, and for statements

Explain labels and their use
Explain break and continue statements
Explain arrays and their use
Practise writing Java programs using the three types of control flow and using arrays

5.   **PROCEDURES AND FUNCTIONS (METHODS)**
Define the procedure concept
Explain the use of a procedure
Describe block structure and scope, local and global variables
Discuss parameters: actual, formal and value parameters
Define the function concept
Explain the return of value from a function
Explain the meaning of method in Java
Practise writing Java programs using methods (procedures and functions)

6.   **OBJECT-ORIENTED PROGRAMMING**
Define procedural and data abstraction
Explain stack as a data abstraction
Explain object-oriented programming, objects and class
Explain object creation using constructors
Explain instance variables and methods, and class variables and methods
Explain class hierarchy: superclass and subclass
Discuss inheritance
Practise writing object-oriented programs

7.   **THE JAVA APPLICATION PROGRAMMING INTERFACE (API)**
Define the concept of package
Describe the *package* and *import* keyword
Discuss the Java API
Practise using the Java API documentation

8.   **INPUTS AND OUTPUTS**
Describe input and output streams
Discuss files and file manipulation
Discuss exception handling
Practise the use of inputs and outputs using Java

9.   **ENCAPSULATION AND POLYMORPHISM**
Describe access modifier: public, private, protected
Describe bundling and information hiding
Explain encapsulation and how to enhance software maintainability
Describe abstract class and abstract method
Describe static and dynamic binding
Describe operation overriding and polymorphism
Practise on access modifier, encapsulation and polymorphism

10.   **SUMMARY**
Summarize important points taught in course
Review course objectives and how they are met

Course Leader

Dr. Danny Poo graduated with a BSc (Hons), MSc and PhD in Computer Science from the University of Manchester Institute of Science and Technology (UMIST), England. He is currently a tenured Associate Professor in the Department of Information Systems, National University of Singapore and teaches courses on Object-Oriented Software Engineering and Enterprise JavaBeans at the undergraduate level. He is presently the Vice-Chairman, Steering Committee for the Asia-Pacific Software Engineering Conference and is actively involved in teaching professionals on Object-Oriented Analysis, Design, and Programming. He is the founder and director of Cicada Cube Pte Ltd, an NUS spin-off company specializing in Enterprise-level Search and Retrieval Solutions. **Dr. Poo is the author of 5 books:** "**Object-Oriented Programming and Java**", 2nd edition, Springer-Verlag, 2007; "**Developing Systems Using J2EE**", Prentice-Hall, 2004, "**Learn To Program Java**", 3rd edition, Thomson Learning, 2006; "**Learn To Program Java User Interface**", Thomson Learning, 2006; and "**Learn To Program Enterprise JavaBeans 3.0**", 2nd edition, Thomson Learning, 2007.