Java
(12 Weeks)

Topic	Lecture No.

Introduction to Java Programming Language
• An Introduction to Java
   o Java as a Programming Platform, The Java "White Paper" Buzzwords, Java and the Internet, A Short History of Java, Common Misconceptions about Java.
• The Java Programming Environment
   o Installing the Java Development Kit, Choosing a Development Environment
	1

The Java Programming Environment • Using the Command-Line Tools, Using an Integrated Development Environment, Compiling and Running Programs from a Text Editor, Running a Graphical Application, Building and Running Applets.	2
Introduction to Object Oriented Programming • Thinking Object-Oriented. Why Is OOP Popular? A New Paradigm, A Way of Viewing the World. • Abstraction Layers of Abstraction, Other Forms of Abstraction.	3

Fundamental Programming Structures in Java
• A Simple Java Program, Comments
• Operators
   o **Arithmetic Operators**-The Basic Arithmetic Operators, The Modulus Operator, Arithmetic Assignment Operators, Increment and Decrement.
   o The Bitwise Operators, The Bitwise Logical Operators, The Left Shift, The Right Shift,The Unsigned Right Shift , Bitwise Operator Assignments.
   o Relational Operators
   o Boolean Logical Operators
   o Logical Operators.
   o The Assignment Operator.
   o The ? : Operator
• Operator Precedence.
   o Using Parentheses
• Strings, Input and Output
	4

Fundamental Programming Structures in Java
• Variables
   o Declaring a Variable
   o Dynamic Initialization
   o The Scope and Lifetime of Variables
   o Type Conversion and Casting
• Data Types
   o Integers: byte, short, int, long
   o Floating-Point Types: float, double
   o Characters
   o Booleans
• Control Flow
   o The if Statement
   o Switch statement
   o The for Loop
	5

• Control Flow
   o Do while loop
   o while loop
   o Nested loops
   o break and continue
• Big Numbers
• Arrays.
   o One-Dimensional Arrays
   o Multidimensional Arrays
   o Alternative Array Declaration Syntax
	6

Classes and Methods • Encapsulation, Class Definitions, Methods. **Objects and Classes** o Introduction to Object-Oriented Programming, Using Predefined Classes, Defining Your Own Classes.	7,8
Objects and Classes • Static Fields and Methods, Method Parameters o Introducing Methods, Adding a Method to the Box Class, Returning a Value. , Adding a Method That Takes Parameters, Constructors, Parameterized Constructors, this Keyword. **Messages, Instances, and Initialization** • Message-Passing Syntax, Statically and Dynamically Typed Languages • Accessing the Receiver from Within a Method, Object Creation, Pointers and Memory Allocation, Constructors {Constant Values}, Destructors and Finalizers.	9,10
Objects and Classes • Packages, Documentation Comments, Class Design Hints. • Object Construction o Declaring Objects, A Closer Look at new, Assigning Object Reference Variables**.**	11

Inheritance and Substitution • An Intuitive Description of Inheritance, Inheritance in Various Languages, Subclass, Subtype, and Substitution, Overriding and Virtual Methods, Interfaces and Abstract Classes. • Classes, Superclasses, and Subclasses, Object: The Cosmic Superclass, Generic ArrayLists	12
Inheritance and Substitution • Forms of Inheritance, The Benefits of Inheritance, The Costs of Inheritance. Examples (Language independent). • Object Wrappers and Autoboxing, Reflection **Static and Dynamic Behavior** • Static versus Dynamic Typing, Static and Dynamic Classes, Static versus Dynamic Method Binding.	13,14
Multiple Inheritance • Inheritance as Categorization, Problems Arising from Multiple Inheritance, Inner Classes. • Enumeration Classes, Design Hints for Inheritance. • Interfaces, Object Cloning, Interfaces and Callbacks, Inner Classes, Proxies. **Polymorphism and Software Reuse** • Polymorphism in Programming Languages, Mechanisms for Software Reuse, Efficiency and Polymorphism, Will Widespread Software Reuse Become Reality?	15-17
Overloading and Overriding • Type Signatures and Scopes, Overloading Based on Scopes, Overloading Based on Type Signatures, Redefinition, Notating Overriding, Replacement versus Refinement, Deferred Methods, Overriding versus Shadowing, Covariance and Contra variance.	18,19
Exceptions and Debugging • Dealing with Errors, Catching Exceptions, Tips for Using Exceptions, Logging, Using Assertions, Debugging Techniques, Using a Debugger.	20
Streams and Files • The Complete Stream Zoo, ZIP File Streams, Use of Streams, Object Streams, File Management, New I/O, Regular Expressions.	21-22
Introduction to GUI • AWT Architecture, Light-Weight vs Heavy-Weight, AWT Event Model, AWT Event Hierarchy & Event Handling, Using Top-Levels, components and containers	23

Introduction to GUI • Introduction to Layouts, Focus Architecture. **Graphics Programming** • Java2D Rendering Model, Strokes & Fills, Geometries, Fonts and Text Layout, Transformations.	24
Graphics Programming • Display and manipulation of Images and offscreen buffers, Using Color, Printing through Java	25
Graphics Programming • Doing More with Images using Image IO, Hardware Acceleration and Active Rendering techniques.	26
User Interface Components with Swing • The Model-View-Controller Design Pattern, Introduction to Layout Management	27
User Interface Components with Swing • Text Input, Choice Components, Menus, Sophisticated Layout Management	28
User Interface Components with Swing • Dialog Boxes **Deploying Applets and Applications** • Applet Basics ,The Applet HTML Tags and Attributes	29
Deploying Applets and Applications • Multimedia, The Applet Context, JAR Files, Application Packaging, Java Web Start, Storage of Application Preferences.	30
Database Programming • The Design of JDBC • The Structured Query Language o Introduction to relational database,Primary key, foreign key ,Examples of relational database,Joins (inner, outer, Cartesian) ,Select ,Where ,Order by ……,Group by …. Having.,Insert, update, delete . • JDBC Installation	31

Database Programming
• Basic JDBC Programming Concepts
    o JDBC-ODBC Bridge, JDBC Drivers, Creating DSN.
• Query Execution
    o DriverManager, Connection, Statement, ResultSet.
• Scrollable and Updatable Result Sets, Metadata, Row Sets, Transactions, Advanced Connection Management, Introduction to LDAP**.**
	32-34

Introduction to UML • Introduction, An outline Development Process and Use cases o What Is the UML?, How We Got Here, Notations and Meta-Models, Why Do Analysis and Design?, Overview of the Process, Inception,Elaboration, Planning the Construction	35

Phase, Construction, Transition, When to Use Iterative Development	
Introduction, An outline Development Process and Use cases • Use Case Diagrams, Business and System Use Cases, When to Use Cases. **Class Diagrams and Advance Concepts** • Perspectives, Associations, Attributes, Operations, Generalization, Constraint Rules, When to Use Class Diagrams, Stereotypes, Object Diagram	36
Class Diagrams and Advance Concepts • Class Scope Operations and Attributes, Multiple and Dynamic Classification, Aggregation and Composition, Derived Associations and Attributes, Interfaces and Abstract Classes, Reference Objects and Value Objects	37
Class Diagrams and Advance Concepts • Collections for Multivalued Association Ends, Frozen, Classification and Generalization, Qualified Associations, Association Class, Parameterized Class, Visibility.	38
Interaction Diagrams, Packages and Collaborations • Sequence Diagrams, Collaboration Diagrams, Comparing Sequence and Collaboration Diagrams, When to Use Interaction Diagrams, Packages, Collaborations, When to Use Package Diagrams and Collaborations.	39
State and Activity Diagrams • Concurrent State Diagrams, When to Use State Diagrams, Decomposing an Activity, Dynamic Concurrency, Swimlanes, When to Use Activity Diagrams. **Physical Diagrams** • Deployment Diagrams, Component Diagrams, Combining Component and Deployment Diagrams, When to Use Physical Diagrams.	40
Case Studies	41-43
Review	44