

# Unit 1 – Java Basics & First Program

## **Q1. Java ki main features (platform independence, OOP, security, multithreading, garbage collection) explain karo. Basic Java application ka structure batao.**

Java fully object-oriented, simple, robust, secure, architecture-neutral aur portable language hai. Source code compiler se bytecode me convert hota hai jo JVM par run hota hai, isliye “write once, run anywhere” possible hai. Automatic garbage collection memory leaks kam karta hai, built-in security model untrusted code (applets) ko sandbox me run karta hai. Multithreading support background me parallel tasks handle karna easy banata hai. Basic Java application me ek class declare ki jati hai jisme public static void main(String[] args) entry point hota hai. Curly braces class aur method blocks define karte hain, semicolon statements end karte hain. Source file ka naam public class ke naam jaisa hota hai (.java), compiler se .class bytecode banta hai jo java command se run hota hai. Applet programs browser/JVM plugin me run hote the, jabki applications standalone JVM process ke roop me execute hote hain.

## **Q2. Applet aur application me antar likho. Java program ko compile/run karne ke basic steps batao.**

Java application console/desktop par standalone program hai jiska entry point main method hota hai aur jo OS ke upar JVM process ke roop me run hota hai. Applet ek special Java program tha jo browser/applet viewer ke andar web page ke part ki tarah chalta tha; isme main nahi, life-cycle methods (init, start, stop, destroy) hote hain. Application generally command line ya GUI se launch hoti hai, applet HTML page me <applet> ya <object> tag se embed hoti thi. Compile/run steps: source file (ClassName.java) likho, javac ClassName.java se compile karo, agar compilation successful ho to ClassName.class generate hota hai, fir java ClassName se program execute hota hai. Source file me ek se zyada classes ho sakti hain, lekin public class ka naam file naam se match karna compulsory hai. Applets ko appletviewer ya browser plugin se load kiya jata tha; aaj kal security reasons ki wajah se modern browsers me applet support remove ho chuka hai.

## Unit 2 – Data Types, Variables, Strings

### Q3. Java ke primitive data types (integral, floating, char, boolean) aur variable naming rules/conventions explain karo.

Java me 8 primitive types hain: byte, short, int, long (integral); float, double (floating); char (16-bit Unicode); boolean (true/false). Inka size fixed hota hai, platform se independent, isliye Java programs predictable behave karte hain. Variable identifier letters, digits, underscore, dollar se ban sakta hai, lekin start letter ya \_/\$ se hi hona chahiye; space, special symbols allowed nahi; reserved keywords naam ke roop me use nahi kar sakte. Naming conventions me variables/methods camelCase (totalMarks), classes PascalCase (StudentInfo), constants ALL\_CAPS (MAX\_SIZE) me likhe jate hain. Primitives stack me direct value store karte hain, jabki reference types (String, arrays, objects) heap me object ka address store karte hain. Constants final keyword se banate hain (final int MAX = 100;), jise baad me change nahi kar sakte; ye readability aur safety improve karta hai.

### Q4. String class ko data type ki tarah kaise use karte hain? new ke sath/ bina new ke difference aur string immutability explain karo.

Java me String ek class hai, primitive nahi; lekin itna common hai ki language uske sath special syntax allow karti hai. String s1 = new String("ABC"); likhne se new object heap me banta hai; String s2 = "ABC"; likhne par string literal pool me entry create hoti hai ya existing reuse hoti hai. Dono case me variable s1/s2 reference hai jo object ka address hold karta hai; == references compare karta hai, equals() content compare karta hai. String immutable hai, matlab ek baar create hone ke baad content change nahi hota; concat jaisi operations naya object return karte hain. Isse thread-safety aur string pool optimization easy ho jata hai, lekin heavy modifications ke case me StringBuilder/StringBuffer prefer kiye jate hain. main method me String[] args command line arguments ko represent karta hai jo runtime par program ko input provide karte hain.

## Unit 3 – OOP, Methods, Control Statements

### **Q5. Class, object, encapsulation, inheritance aur composition (containment) concepts explain karo.**

Class user-defined type hai jo data members (fields) aur methods ka blueprint hota hai; object class ka runtime instance hai jisko new se create karte hain. Encapsulation ka matlab data + methods ko ek unit me band karna aur access modifiers (private, public, protected) se controlled access dena; isse data hiding, maintainability improve hoti hai. Inheritance me ek class (subclass) doosri class (superclass) ke members inherit karti hai, jisse code reuse, polymorphism possible hota hai; extends keyword use hota hai. Abstract classes partial implementation deti hain; kuch abstract methods subclass me implement hone hote hain, jisse common behavior share hote hue customization possible ho. Composition/containment me ek class dusre class ka object as field rakhkar "has-a" relationship show karti hai (Car has-a Engine); contained object ki lifetime containing object se tied ho sakti hai. Association looser relationship hai jahan objects logically related hain lekin lifetime independent hota hai (Student–Course).

### **Q6. Methods, method overloading, constructors aur type casting/promotions explain karo.**

Method reusable block of code hai jo specific task perform karta hai; declaration me return type, name, parameter list, body include hota hai. Method overloading me same naam ke multiple methods alag parameter list (type/number/order) ke sath define kiye jate hain; compiler call time par best match choose karta hai, isse readability banti hai. Constructor special method hai jo object banne par automatically call hota hai, iska naam class jaisa hota hai, koi return type nahi hota; default, parameterized, copy-like constructors possible hain. this keyword current object refer karne aur constructor chaining ke liye use hota hai; super se parent class constructor/members access karte hain. Type promotion me smaller types automatically larger type me convert ho jate hain (int to long, float to double) expressions evaluate karte waqt; explicit casting (double to int) narrow conversions ke liye zaroori hai. Arithmetic, relational, logical, assignment, increment/decrement operators control flow constructs (if, switch, loops) ke sath basic program logic banate hain.

## Unit 4 – GUI (AWT) & ActiveX / Scripting Overview

### **Q7. AWT (Abstract Window Toolkit) ka overview, Frame, Button, Panel aur Layout managers ka basic use explain karo.**

AWT Java ka original GUI toolkit hai jo native windowing system ke upar heavyweight components provide karta hai. java.awt package me core classes jaise Frame (top-level window), Button, Label, TextField, Panel, Layout managers (FlowLayout, BorderLayout, GridLayout) available hain. Frame application ka main window represent karta hai; isme components add() method se add hote hain. Panel ek container hai jo components ke group ko hold karta hai, jisse complex layouts (nested panels) create kiye ja sakte hain. Layout managers automatic component positioning/sizing handle karte hain; FlowLayout left-to-right flow, BorderLayout regions (North, South, East, West, Center), GridLayout rows×columns grid banata hai. Event handling delegation event model par based hai jahan listeners (ActionListener, MouseListener) register hote hain aur user actions par callback methods invoke hoti hain. Later Swing (javax.swing) ne lightweight, pluggable look-and-feel components provide kiye, lekin exam me basic AWT hierarchy, Frame creation, button click handling important hai.

### **Q8. ActiveX architecture, client-side/ server-side ActiveX controls ka basic concept likho. HTTP request/response aur ASP ka short note do.**

ActiveX Microsoft ki COM/DCOM based technology hai jo Windows platforms par reusable binary components (controls, code components, documents) banane aur host karne ke liye use hoti hai. ActiveX controls client-side browser ya applications me embed hote hain (OCX), jisse rich UI, multimedia, custom widgets banaye ja sakte hain; security risk ke kaaran modern browsers me ye heavily restricted/disabled ho chuke hain. Server-side ActiveX components IIS/ASP environment me business logic, database access, custom processing ke liye COM objects ke roop me use hote the. ASP (Active Server Pages) server-side scripting framework hai jahan HTML ke andar server scripts (VBScript/JScript) likh kar dynamic content generate kiya jata hai. HTTP client-server protocol hai jahan browser request line + headers + optional body send karta hai, server status line + headers + body (HTML, JSON, etc.) return karta hai. ASP engine request read karke, script execute karke, HTML response generate karta hai, cookies/Session se user information store ki jati hai.

### **Q9. Client-side scripting vs server-side scripting kya hai? Internet applications me inka role batao.**

Client-side scripting browser ke andar run hone wala code hai (JavaScript jaisa) jo HTML/CSS ke sath interact karke UI behavior, validation, DOM manipulation, animations handle karta hai. Ye code user machine par execute hota hai, isliye server load kam rehta hai aur response fast mil sakta hai; lekin security/sensitivity ke kaam ke liye is par trust nahi kiya jata. Server-side scripting (ASP, PHP, JSP, etc.) web server par execute hota hai; ye database se data fetch, business logic apply karke dynamic HTML/JSON generate kar ke browser ko bhejta hai. Client se aane wale HTTP request me URL, query parameters, headers, body hote hain; server response status code, headers (content-type, cookies) aur body ke sath wapas jata hai. Client-side + server-side scripting milkar complete web application banate hain: client UI + basic checks, server secure data processing + storage. Modern frameworks (AJAX, SPA) asynchronous requests se in dono ko tightly integrate karte hain.