Web Programming

III/I (R22)

- R22AI513PE

PART-A

1. (a) Mention any one JavaScript data type.

Ans: JavaScript features eight primary data types, categorized into primitive and non-primitive types.

Eg: String : Represents textual data, like "hello".

(b) What is form validation?

Ans: Form validation is the process of checking user input in web forms to ensure it meets specific requirements before submission, preventing errors like empty fields or invalid data formats.

(c) Define method overloading.

Ans: Method overloading is a programming concept where multiple methods share the same name but differ in parameters, allowing flexible function calls based on arguments.

(d) Mention any feature of Java.

Ans: One prominent feature of Java is platform independence, enabled by the Java Virtual Machine (JVM), which allows code compiled once to run on any device or operating system with a JVM installed, following the "write once, run anywhere" principle.

(e) What is InetAddress?

Ans: InetAddress is a Java class in the java.net package that represents an IP address, supporting both IPV4 (32-bit) and IPv6 (128-bit) formats and often includes an associated hostname.

(f) What is Serialization in RMI?

Ans: Serialization in Java RMI is the process of converting objects - such as method parameters, return values and exceptions - into a byte stream for transmission across the network between client and server JVMs.

(g) What is Event Handling?

Ans: Event handling is a mechanism that detects and responds to user actions or system-generated events, such as button clicks, mouse movements or key presses, primarily in GUI applications. ~~using AWT or Swing~~.

(h) What is Session Tracking?

Ans: Session tracking is a technique in web applications to maintain state and user-specific data across multiple stateless HTTP requests, as HTTP itself does not retain session information between interactions.

(i) What is Namespace?

Ans: A namespace in computing is a context or scope that groups unique identifiers (like variables, functions or classes) to prevent naming conflicts across different parts of code or systems.

(j) Define Web Driver. (SOAP)

Ans: Web Driver is a protocol and set of tools, most famously part of Selenium, used to automate web browsers. by simulating user actions to test web applications, scrape data or perform other tasks across different browsers and operating systems. efficiently.

PART-B

(2) Discuss HTML5 and CSS3 features in modern web designing.

Ans: HTML5 and CSS3 form the foundation of modern web design, enabling semantic structure, multimedia integration, responsive layouts and dynamic visuals without plugins.

HTML5 Key features

HTML5 introduces semantic elements like <header>, <footer>, <article> and <section> for better document structure, improving accessibility and SEO. Native <video> and <audio> tags support multimedia embedding directly in browsers, while the <canvas> element allows JavaScript-driven graphics and animations for interactive elements like charts or games. Additional capabilities include form enhancements with types like email and date, local Storage for offline data, geolocation API and drag-and-drop functionality.

CSS3 Key Features

CSS3 advances styling with Flexbox and Grid for flexible, responsive layouts that adapt to any screen size using viewport units like vw and vh. It supports

2D/3D transformations (transform: rotate()), transitions and animations for smooth effects, plus enhanced selectors, gradients, shadows and rounded corners for richer visuals.

Media queries enable device-specific designs, ensuring fluid grids and optimal performance across desktops, tablets and mobiles.

(3)(a) · Illustrate event handling in Javascript with suitable examples.

Ans: An event ~~listener~~ handling is a Javascript function that waits for an event (like a click, hover, key press, etc.) to happen and then executes a specific block of code in response.

~~Instead of writing Javascript inline in HTML, event~~

tt

Javascript event handling allows web pages to respond dynamically to user interactions like clicks, key presses and mouse movements using event listeners.

Event Type	Trigger
click	User clicks an element
mouseover	Mouse moves over an element
Keydown	User presses a key
Submit	Form is submitted
load	Page or image finishes loading
change	Input field value changes.

3(b) Explain difference between client-side and server-side scripting.

Ans: Client Side Scripting:

1) Client side scripting is the process of scripting on the web Browser.
2) The source code in client side scripting can be visible.
3) The main aim is to get the output
4) There is no security for the data in the client side scripting.
5) The code can be accessed on the computer Browser of the client in the client side scripting.

6) The languages used for Scripting are html, css, JavaScript etc.

7) The client side scripting depends completely on the client

Server side Scripting:

1) Server side scripting is the process of scripting on the web server.

2) The source code in the server side scripting cannot be visible.

3) The primary aim is to manipulate and access the database.

4) There is security for the data in the server side scripting.

5) The code can be accessed in the server side scripting.

6) The languages used for scripting are python, php, Java, Ruby.

7) The server side scripting doesnot depends on the client.

(4) Explain different types of inheritance in Java.

Ans:→ The process of creating a new class with the properties and Behaviours of the Existing class is called as Inheritance.

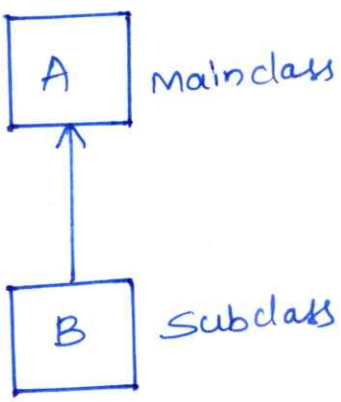
→ The Inheritance is classified into 5 types. They are:

(i) Single Inheritance.
(ii) Multi-level Inheritance.
(iii) Hierarchical Inheritance.
(iv) multiple Inheritance.
(v) Hybrid Inheritance.

(i) Single Inheritance:

→ Single Inheritance is the process of creating a subclass from the main class.

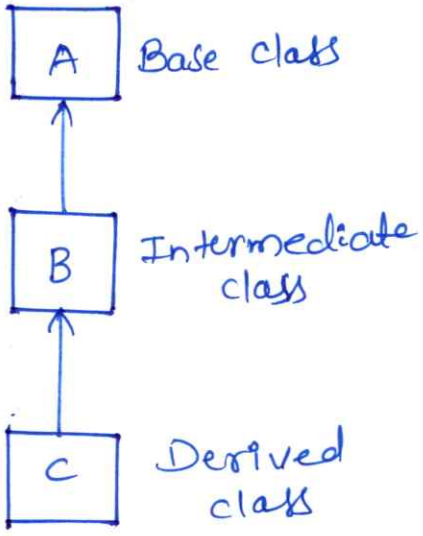
→ The subclass contains the properties and behaviour of the parent class.



(ii) Multi-level Inheritance;

→ Multi-level Inheritance is the process of deriving class from both Intermediate class and Base class.

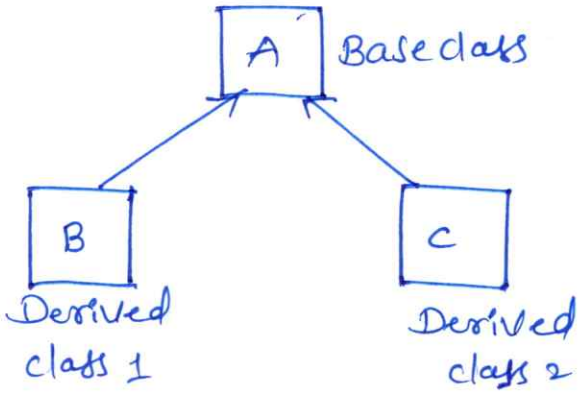
→ The Derived class contains the properties and behaviour of both Intermediate and Base class.



(iii) Hierarchical Inheritance:

→ Hierarchical Inheritance is the process of deriving two (or) more classes from the Base class.

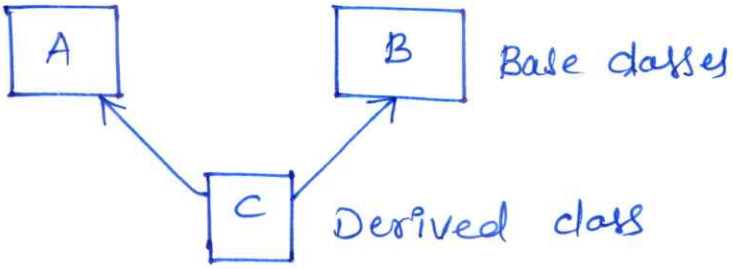
→ All the derived classes contains the properties and behaviour of Base class.



(iv) Multiple Inheritance:

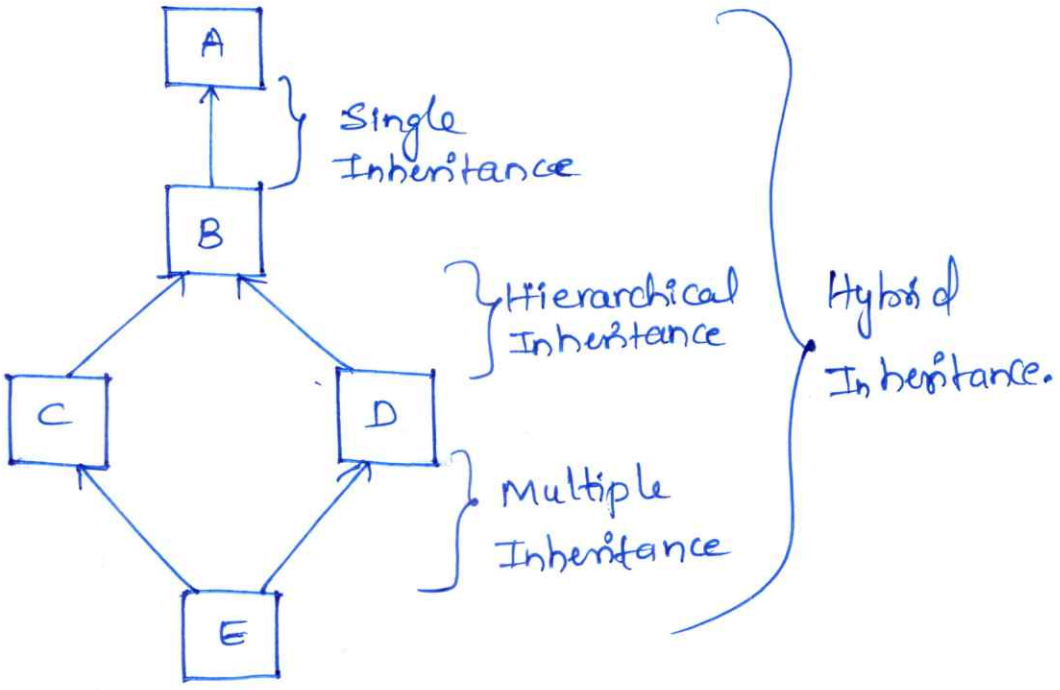
→ Multiple Inheritance is the process of obtaining one derived class from two Base classes.

→ In this, the derived class contains properties of both Base classes.



(v) Hybrid Inheritance:

→ Hybrid Inheritance is the process of combination of all the Inheritance.



(5) Compare method overloading and overriding with examples.

Ans: Method overloading and overriding are key OOP concepts primarily in languages like Java and C++, where overloading enables multiple methods with the same name but different parameters in the same class, while overriding redefines a superclass method in a subclass with the same signature.

Key Differences

Aspect	Overloading	Overriding
Location	Same class	Subclass
Signature	Same name, different parameters	Same name, parameters, return type.
purpose	Polymorphism at compile-time	Run time polymorphism
Binding	Static (compile-time)	Dynamic (runtime)
Access Modifier	can differ	cannot reduce visibility

(6) Explain JavaBeans Architecture with example.

Ans: JavaBeans architecture is a component model for Java that defines how reusable software components should be written so they can be easily discovered, configured and connected in tools and applications. A JavaBean is just a normal Java class that follows specific conventions so builder tools can automatically detect its properties, events and methods.

Architecture elements

JavaBeans architecture describes how beans expose features so tools and other components can interact with them.

→ Properties: represent configurable state such as width, height or color and can be simple, indexed (arrays) or bound/constrained (notify listeners when changed.

→ Events: beans can generate events and notify registered listeners so components can react to user actions or state changes.

→ Methods : public methods form the operational interface that other components or tools can call.

→ Persistence : through serialization, a bean's state can be stored and later reconstructed, supporting design-time and runtime configuration.

(7) Explain RMI Architecture - Stub, Skeleton & binding.

Ans: The RMI (Remote Method Invocation) is an API that provides a mechanism to create distributed application in java. The RMI allows an object to invoke methods on an object running in another JVM.

The RMI provides remote communication between the applications using two objects stub and skeleton.

Stub: The stub is an object, acts as a gateway for the client side. All the outgoing requests are routed through it. It resides at the client side and represents the remote object. When the caller invokes method on the stub object, it initiates a connection with remote Virtual Machine (JVM).

Skeleton: The skeleton is an object, acts as a gateway for the server side object. All the incoming requests are

routed through it. When the ~~skelton~~ skeleton receives the incoming request, it reads the parameter for the remote method. It invokes the method on the actual remote object and it writes and transmits the results to the caller.

(8) Explain Servlet lifecycle.

Ans: The Web container maintains the life cycle of a Servlet instance. Let's see the life cycle of the servlet:

1. Servlet class is loaded
2. Servlet instance is created
3. init method is invoked.
4. service method is invoked
5. destroy method is invoked.

1. Servlet class is loaded

The classloader is responsible to load the servlet class. The servlet class is loaded when the first request for the servlet is received by the web container.

2. Servlet instance is created.

The web container creates the instance of a servlet after loading the servlet class. The servlet instance is created only once in the servlet life cycle.

3) init method is invoked:

The web container calls the init method only once after creating the servlet instance. The init method is used to initialize the servlet. It is the life cycle method of the javax.servlet.Servlet interface. Syntax of the init method is given below:

public void init(ServletConfig config) throws ServletException

4) service method is invoked:

The web container calls the service method each time when request for the servlet is received. If servlet is not initialized, it follows the first three steps as described above then calls the service method.

5. destroy method is invoked:

The web container calls the destroy method before removing the servlet instance from the service. It gives the servlet an opportunity to clean up any resource for example memory, thread etc.

(9)

(9) Explain layout managers with examples.

Ans: The Layout Managers are used to arrange components in a particular manner. The Java LayoutManagers facilitates us to control the positioning and size of the components in GUI forms. LayoutManager is an interface that is implemented by all the classes of layout managers. These are the following classes that represent the layout managers:

java.awt.BorderLayout
java.awt.FlowLayout
java.awt.GridLayout
java.awt.CardLayout
java.awt.GridBagLayout
java.swing.BoxLayout
java.swing.GroupLayout
java.swing.ScrollPanelLayout
java-swing.SpringLayout etc.

Java.BorderLayout

The BorderLayout is used to arrange the components in five regions: north, south, east, west and center. Each region may contain one component only. It is the default layout of a frame or window. The BorderLayout provides five constants for each region:

1. public static final int NORTH
2. public static final int SOUTH
3. public static final int EAST
4. public static final int WEST
5. public static final int CENTER

(10) Explain Web services Architecture.

Ans: Web services are the types of internet software that uses standardized messaging protocol over the distributed environment. It integrates the web-based application using the REST, SOAP, WSDL. and UDDI over the network. For example, Java web service can communicate with .Net application.

~~Features~~ components of Web Services

→ The web service must be accessible over the internet.

→ The web service is discoverable through a common mechanism like UDDI.

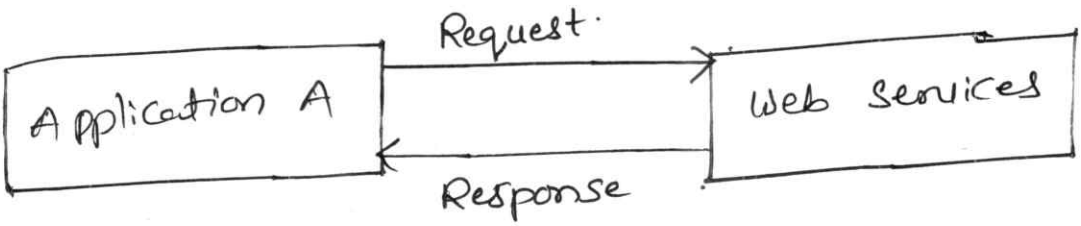
→ It must be interoperable over any programming language or Operating system.

Uses of Web services.

→ Web services are used for reusing the code and connecting the existing program.

→ Web services can be used to link data between two different platforms.

→ It provides interoperability between disparate applications.



(11) Explain SOAP message structure with example.

Ans: A SOAP message is an XML document with a fixed overall shape: an Envelope at the root, an optional Header, a mandatory Body and an optional Fault for errors.

Main parts of a SOAP message.

→ Envelope: Root element that marks the document as a SOAP message and defines the SOAP namespace, for example.

Http://www.w3.org/2003/05/soap-envelope

→ Header (optional): contains metadata such as security, authentication or routing information that intermediaries or the final service use while processing the message.

→ Body (mandatory): Holds the actual request or response data (the operation being called and its parameters or the result.

→ Fault (optional): Nested inside the Body and used to describe errors that occured while processing the message (error code, description etc).