Course Name:  
**Diploma in J2EE & JSP Technology**  
Course Code: D4

Address:  
Info Tech Corporation of Goa Ltd.,  
IT HUB, 3rd Floor,  
Altinho, Panaji-Goa.  
Pin: 403001  
e-mail Id: md-itg.goa@nic.in  
Tel: (0832) 2226024/22241926
Diploma in J2EE & JSP technology

1. Introduction:

Java is a programming language that is a core component of Sun Microsystems' Java platform. The language derives much of its syntax from C and C++ but has a simpler object model and fewer low-level facilities. Java applications are typically compiled to bytecode (class file) that can run on any Java Virtual Machine (JVM) regardless of computer architecture. Java is general-purpose, concurrent, class-based, and object-oriented, and is specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere". Java is considered by many as one of the most influential programming languages of the 20th century, and is widely used from application software to web applications.

2. Eligibility:

Graduate in any discipline

3. Duration:

260 Hours. 2 Hours each day
- Theory 60 hours
- Practical 200 hours

4. Course Details:

<table>
<thead>
<tr>
<th>Type of course</th>
<th>Professional IT Course</th>
</tr>
</thead>
<tbody>
<tr>
<td>Syllabus</td>
<td>Programming Concepts</td>
</tr>
<tr>
<td></td>
<td>o Flowcharts</td>
</tr>
<tr>
<td></td>
<td>o Algorithms</td>
</tr>
<tr>
<td></td>
<td>o Data Flow Diagrams</td>
</tr>
<tr>
<td></td>
<td>o Software Life Cycle</td>
</tr>
<tr>
<td></td>
<td>HTML</td>
</tr>
<tr>
<td></td>
<td>INTRODUCTION</td>
</tr>
<tr>
<td></td>
<td>The World Wide Web (WWW)</td>
</tr>
<tr>
<td></td>
<td>HTML History</td>
</tr>
<tr>
<td></td>
<td>Hypertext and Hypertext Markup Language</td>
</tr>
<tr>
<td></td>
<td>Microsoft Front Page</td>
</tr>
<tr>
<td></td>
<td>HTML DOCUMENTS</td>
</tr>
<tr>
<td></td>
<td>Dividing the document into 2 parts.</td>
</tr>
<tr>
<td></td>
<td>Headers</td>
</tr>
<tr>
<td></td>
<td>Body</td>
</tr>
<tr>
<td></td>
<td>Tags</td>
</tr>
<tr>
<td></td>
<td>Representing 2 types of tag (odd and even)</td>
</tr>
</tbody>
</table>
## Elements of an HTML Document
- Text Elements
- Tag Elements
- Special Character elements

## STRUCTURAL ELEMENTS OF HTML DOCUMENTS
- Header tags
- Body tags:
  - Paragraphs
  - Titles
  - Lists
    - Numbered list
    - Non-Numbered lists
    - Definition lists

## FORMATTING HTML DOCUMENTS
- Logical styles (source code, text enhancements, variables)
- Physical Styles (Bold, Italic, underlined, crossed)

## MANAGING IMAGES IN HTML
- Image format (quality, size, type, …)
- Importing images (scanners)
- Tags used to insert images
- Frames

## CREATING TABLES IN HTML DOCUMENTS
- Tags used in table definition
- Tags used for border thickness
- Tags used for cell spacing
- Tags used for table size
- Dividing table with lines
- Dividing lines with cells
- Cell types
  - Titles cells
  - Data cells

## HYPERTEXT AND LINK IN HTML DOCUMENTS
- URL/FTP/HTTP
- Types of links:
  - Internal Links
  - External Links
- Link Tags
- Links with images and buttons
- Links that send email messages
### SPECIAL EFFECTS IN HTML DOCUMENTS

- Text fonts
- Sensitive Images
- Tip tables
- Page background
  - Variable
  - Fixed
- Rotating messages (Marquee)
- Counters

### MULTIMEDIA

Audio files and acceptable formats (AIFF, AU, MIDI, WAVE)
- Inserting audio files

Video files and acceptable formats (MPEG, Quick Time, Video for Windows)
- Inserting video files
  - Screen control attributes (WIDTH, HEIGHT, ALIGN).
  - Start control attributes (START, FILEOPEN, LOOP, LOOPDELAY, MOUSEOVER).

### MANAGING FORMS

- Interactive forms
- Creating data entry forms

### J2EE

#### Understanding Java and the J2EE Platform

- Reviewing a brief history of Java
- Understanding J2SE
- Examining the origin of (J2EE)
  - Application components
  - Roles
- Working with the model-view-controller
  - The model
  - The view
  - The control
- Understanding J2EE APIs
  - J2EE standard services
  - Application component APIs
- Understanding the Java Community Process (JCP)
Reviewing XML Fundamentals

Well-formed XML
Valid XML
Understanding XML Document structure
Prologue
Elements
Attributes
Examining XML parsers
DOM parsers
SAX parsers
DOM versus SAX
Implementing XML DTD’S
Understanding XML Namespaces
Exploring XML schema
Working with extensible style sheet
Language transformations (XSLT)
Producing a simple HTML with XSLT
Producing a wireless markup language (WML) document with XML
Introducing J2EE XML-based API’S

Introducing application servers
Implementing the J2EE platform
Understanding the features of an application server
Scalability
Client agnosticism
Server management
Development
JBoss, Apache, Tomcat

The presentation tier

Studying serve let programming
Creating a magazine publisher application using servlets
The server side
The client side
Creating an HTML login screen
Using the servlet context
Performing URL Redirection
Using requestDispatcher
Using sendRedirect()
The lost password screen example
Session tracking with servlets
Cookies
URL rewriting
Hidden fields
The session-tracking API with HttpSession object
Example of a LoginServlet with an access counter
Listener
Filters
Deploying servlets
The web-application archive
Examining the web.xml deployment descriptor
Mandatory servlet elements
Servlet listener elements
Servlet filter elements
Applet-servlet communication

Going over JSP Basics
Introducing JSP
Examining MVC and JSP
JSP scripting elements and directives
  Declarations
  Expressions
  Directives
  Scriptlets
  Comments
  Actions
  Implicit JSP objects
  Working with variable scopes
  Error pages
Using JavaBeans
  Using JavaBeans in JSP
  The scope of JavaBeans
  Creating a login jsp using a java bean
  Deploying the login jsp example using tomcat

Using jsp tag extension
Explaining custom-tag concepts
  Working with the jsp standard tag library
  Importing a tag library
  The tag library descriptor
The tag-library descriptor locations
  Explanation taglib mapping
Understanding tag handlers
  Classic tag handlers
  Simple tag handlers
Exploring dynamic attributes

The enterprise information system tier
Introducing java transactions

atomic transactions
Examining transactional objects and participants
Reviewing atomicity and the two-phase commit protocol
  Optimizations
  Heuristics and removing the two-phase block
Understanding Local and Distributed transactions
  Local transactions
  Distributed transactions
  Interposition
Understanding consistency
Introducing isolation (Serializability)
  - Optimistic versus pessimistic concurrency control
  - Degrees of isolation
Understanding the role of durability
Performing failure recovery
Using transaction-Processing Monitors

Transaction Models
  - Nested transactions
  - Nested top-level transactions
  - Extended transaction models and the J2EE activity service

Understanding transaction standards
  - X/Open distributed transaction processing
  - The object transaction service

Understanding the java transaction API
  - The JTA’S relationship to the JTS
  - The user transaction interface
  - The transaction manager interface
  - Suspending and resuming a transaction
  - The transaction interface
  - The XAResource interface
  - Enrolling participants with the transaction
  - Transaction synchronization
  - Transaction equality
  - The XID interface

Examining JNDI AND Directory services

  Explaining naming services and directory services
  Providing an overview of X.500 and LDAP
     - LDAP implementation
     - Configuring OpenLDAP
     - LDAP schema
  Reviewing the JNDI structure
     - Directories and entries
     - Names and attributes
     - Biding and references
     - Contexts and subcontexts
     - File systems
     - DNS naming conventions
     - LDAP mapping
  Using JNDI and LDAP
     - Connecting to the server
     - Specifying environment properties
     - Implementing authentication
     - Performing simple LDAP lookups
     - Performing searches and comparing entries
     - Modifying the directory
     - Adding objects to a directory
  Connecting to DNS
     - DNS environment properties
     - DNS lookups
     - Reverse DNS lookups
<table>
<thead>
<tr>
<th>Considering other JNDI service providers</th>
</tr>
</thead>
<tbody>
<tr>
<td>File systems</td>
</tr>
<tr>
<td>COS naming for CORBA</td>
</tr>
<tr>
<td>Network information system</td>
</tr>
<tr>
<td>Directory services markup language</td>
</tr>
<tr>
<td>Application server providers</td>
</tr>
<tr>
<td>Exploring the enterprise javabeans environment</td>
</tr>
</tbody>
</table>

### Understanding Java Authentication and Authorization Services

Examining the importance of Java security
- Typical Java security weaknesses
- Providing an overview of JAAS

Understanding security realms
- Single login across security domains
- Setting up for JAAS
- Callback handlers
- Pluggable/stackable authentications

Examining the Java subject class
- Authenticating users
- Authorizing users
- JAAS policy files
- Compiling the example

Debugging the simple JAAS module
- Hiding JAAS
- Predefined JAAS login callbacks and their handlers
- Custom login modules
- Writing your own login handler
- Writing your own callback handler
- Authenticating a web user against a Windows NT domain
- Brief security analysis
- Security limitations
- Implementations
- Alternative methods

### The Service Tier

Understanding EJB Architecture and Design

Explaining the EJB components model
- Reviewing roles, relationships, and responsibilities
- The deployment descriptor
- The bean provider
- The server/container provider
- The application assembler
- The EJB deployer
- The system administrator

The enterprise javabeans
- Entity beans
- Session beans
- Entity beans versus session beans
- Message-driven beans (MDB)
- Understanding EJB container functionality
Restrictions on the bean provider
Achieving scalability by pooling resources
The life of an entity bean
The life of a session bean
Transactions and EJBs
Container-managed transactions
Examining a transactional EJB example
Naming objects
The security infrastructure
The timer service
Persistence in BMP and CMP
Distribution support
Integrating with CORBA
  Why CORBA important to J2EE
  When J2EE met CORBA
Performance and scalability issues
  Application-server availability strategies
  Transaction concerns
  Threading model
  Tools

**Explaining session beans and business logic**

Writing a session EJB
  The home interface
  The component interface
  The session bean class
  The deployment descriptor
  The stateless session bean
  The ejb-jar.xml deployment descriptor
Deployment
  Writing an EJB client
Stateful-session-bean model
  The lifecycle of the stateful session bean
Passivation and activation
  Implementing the session synchronization interface
Storing a handle
Choosing between stateless and stateful beans
  The stateless model
  The stateful model
Summary

**Working with entity beans**

Understanding entity beans
  Remote and local client views
Entity-bean components
  The entity-container contract
Container-managed persistence (CMP)
Bean-managed persistence (BMP)
Using message-driven beans
Understanding the need for MDB
Reviewing MDB lifecycle methods
Examining MDB deployment descriptors
   Deployment descriptor as per EJB 2.0
   Changes in MDB 2.1 deployment descriptors
   Internal messaging within EJB applications
Understanding clients and MDB
Working with EJBs asynchronously

The data tier

Reviewing java database connectivity

Introduction JDBC driver types
Creating your first JDBC program
   Retrieving data
   Database-error processing
   Processing result sets
   The resultsetmetadata class
   Scrollable result sets
   The preparedstatement class
   The callable statement class
Performing batch updates
Using savepoints
Configuring the JDBC-ODBC Bridge
Explaining database connection pools and data sources
   Configuring connection pools
   Creating data source objects
Revisiting DBProcessor
Using the rowset interface
   Working with cachedrowset
   The webrookset class

Web services

Introducing web services
Defining web services
   Universal resource identifiers
   XML-based technologies
Why do we need web services?
   Remote method invocation
   DCOM
   CORBA
   Web-service architecture
   Advantages of web services
Examining Some Web-Service Scenarios
   Enterprise-application integration (EAI)
Understanding the Technologies behind Web Services
   SOAP
   WSDL
   UDDI
   Web services in a service-oriented architecture
Understanding J2EE Web Services

- Integrating J2EE and Web Services
  - Using Java servlets in a Web-services architecture
  - Exposing EJBs as Web services
  - Using JMS as a transport layer
  - Exploring Products and Tools for Web Services

- JSR 109—J2EE Web Services
  - The client-side programming model
  - The server-side programming model
  - Web-service deployment descriptors

**Advanced topics**

Exploring Frameworks and Application Architecture

- Frameworks versus class libraries
- The pains of J2EE
- Understanding Framework Principles
- Inversion of control
- Separation of concerns
- Loose coupling
- Extensibility
- Configurability
- Alignment
- Design patterns
- Examining the Struts framework example

Understanding Framework Objectives and Benefits

- Design
- Development and testing
- Production and maintenance
- Application portfolios

Reviewing Application Architecture beyond Frameworks

- Overview of architectures
- Traditional application architecture
- Services-oriented architecture
- Application architecture versus frameworks

Building Your Own Framework

- Building versus buying
- Open source
- Software vendor
- System Integrators (SIs)

Predicting the Future of Frameworks

Alternatives to Frameworks

- All-in-one proprietary environments
- Model-driven architecture
- Minimal J2EE
- Advanced Integrated Development Environments
**JSP**

**Introduction to Web Application Technologies**

Web Application Technologies
- HTTP Client-Server Architecture
- Web Site Structure

Web Sites and Web Applications
- Execution of CGI Programs
- Advantages and Disadvantages of CGI Programs
- Execution of Java Servlets
- Advantages and Disadvantages of Java Servlets
- Using Separate Processes or Using Threads

Java Servlets
JavaServer

Web Application — Three—Tier Architecture
Model-View—Controller (MVC) Architecture in a Web Application
- Model 2 Architecture
- Model 2 Frameworks
- Java EE Containers
- Java EE Architecture Example
- Job Roles
- Web Application Migration

**Developing a View Component**

Designing a View Component
- Types of View Components
- Soccer League Case Study
- List Leagues Analysis Model
- List Leagues Page Flow

HTTP Revisited
- Hypertext Transfer Protocol
- HTTP GET Method
- HTTP Request
- HTTP Request Headers
- HTTP Response
- HTTP Response Headers

Web Container Architecture
- Request and Response Process
- Sequence Diagram of an HTTP GET Request

Developing a Simple HTTP Servlet
- List Leagues Architecture Model
- The HTTPServlet API
- The ListLeaguesServlet Class

Configuring and Deploying a Servlet
- Soccer League Web Application Structure
- Configuring a Servlet Definition
- Configuring a Servlet Mapping
- Complete Deployment Descriptor
<table>
<thead>
<tr>
<th>Course Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web Application Context Root</td>
</tr>
<tr>
<td>Sun Java™ System Application Server Deployment</td>
</tr>
<tr>
<td>WAR Files for Deployment</td>
</tr>
<tr>
<td>Activating the Servlet in a Web Browser</td>
</tr>
</tbody>
</table>

**Developing a Controller Component**
- Designing Controller Components
  - Types of Controller Components
  - Add a New League Analysis Model
  - Add a New League Page Flow
  - Form Verification
  - Soccer League Web Structure
- Creating an HTML Form
  - Add a New League Web Form
  - The form Tag
  - Textfield Component
  - Drop-Down List Component
  - Submit Button
  - Complete HTML Form
- How Form Data Is Sent in an HTTP Request
  - Form Data in the HTTP Request
  - HTTP GET Method Request
  - HTTP POST Method Request
  - HTTP GET and POST Methods
- Developing a Controller Servlet
  - Add a New League Use Case Revisited
  - Servlet API to Retrieve Form Parameters
  - Developing the AddLeagueServlet Servlet
- Dispatching From the Controller to a View
  - Add a New League Use Case
  - Request Scope
  - Using a Request Dispatcher
  - Developing the AddLeagueservlet Code
  - Developing the SuccessServlet Code

**Developing Dynamic Forms**
- Servlet Life Cycle Overview
  - Servlet Class Loading
  - One Instance Per Servlet Definition
  - The init Life Cycle Method
  - The service Life Cycle Method
  - The destroy Life Cycle Method
- Using Initialization Parameter to Customize the Add a New League Form
- Handling Errors in a Web Form
- Repopulating Web Forms

**Sharing Application Resources Using the Servlet Context**
- Purpose of the Servlet Context
  - Soccer League Demonstration
  - Servlet Context
- Developing a Servlet Context Listener
Designing the Business Tier
Describing the Analysis Model
Domain Entities
Designing a Service Component
   Entity Service
   Facade Service

Developing Web Applications Using Struts
Struts Framework
   Model-View-Controller Pattern
   Front Controller Pattern
   Struts MVC Framework
Developing a Struts Action Class
   Struts Action Class API
   Creating an Action Class
Configuring the Struts Action Mappings
   Configuring the Infrastructure Controller
   Configuring Action Mappings
   Installing the Struts Library Files

Developing Web Applications Using Session Management
HTTP and Session Management
Designing Web Applications to Use Session Management
Using Session Management in a Web Application
Using Cookies for Session Management
Using URL-Rewriting for Session Management

Using Filters in Web Applications
Web Container Request Cycle
   Web Container Request Processing
   Applying Filters to an Incoming Request
   Applying Filters to a Dispatched Request
Filter API
Developing a Filter Class
   The PerformanceFilter: Class
   The init Method
   The doFilter Method
   The destroy Method
Configuring the Filter
   Declaring a Filter in the web .xml File
   Declaring a Filter Mapping in the web .xml File

Integrating Web Applications With Databases
Mapping Sample Data Structures into Database Entities
Designing a Web Application
   Domain Objects
   Database Tables
   Data Access Object (DAO) Pattern
   DAO Pattern Advantages
   JDBC™ API
Developing a Web Application Using a Database
   Traditional Approaches to Database Connections
   Using a DataSource and the JNDI API
   Configuring a DataSource and the JNDI API

Developing JSP Pages
   JavaServer Pages Technology
      How a JSP Page Is Processed
      Developing and Deploying JSP Pages
   Writing JSP Scripting Elements
      Comments
      Directive Tag
      Declaration Tag
      Scriptlet Tag
      Expression Tag
      Implicit Variables
   Using the page Directive
   Using Standard Tags
   Using Expression Language (EL) Elements
   Configuring the JSP Enviromnent

Developing JSP Pages Using Custom Tags
   The Java EE Job Roles Involved in Web Application Development
   Designing JSP Pages with Custom Tag Libraries
      Contrasting Custom Tags and Scriptlet Code
      Developing JSP Pages Using Custom Tags
      Custom Tag Library Overview
      Custom Tag Syntax Rules
      JSTL Sample Tags
   Using a Custom Tag Library in JSP Pages
      Using an Empty Custom Tag
      Using a Conditional Custom Tag
      Using an Iterative Custom Tag
      JSTL Tags

Developing Web Applications Using Struts Action Forms
   Struts Application Components
      Model Elements Review
      Control Elements Review
      View Elements Review
   Developing an ActionForm Class
      The Add a New League Form
      The AddLeagueForm Class
      Data Conversion
      Writing the validate Method
      Struts ActionError Class
      How the Controller Uses the Form Bean
   Developing the JSP Code for a View Form
   Configuring the View Forms
      Configuring the Form Beans
      Configuring the View Aspects of the Actions

Building Reusable Web Presentation Components
   Complex Page Layouts
Presentation Segment Overview
Organizing Presentation Segments
Including JSP Page Segments
  Using the include Directive
  Using the jsp : include Standard Action
  Using the jsp : param Standard Action
Developing Layouts Using Struts Tiles
  Layout JSP File
  Tiles Layout
  Content Body

Introducing JavaServer™ Faces Technology
Introducing JavaServer Faces Technology
Components of a JavaServer Faces Technology-Based Web Application
Comparison of JavaServer Faces Technology With Struts Framework
Key JavaServer Faces Concepts
UI Component Model
UI Component Classes

Quick Reference for HTML
HTML and Markup Languages
  Creating an HTML Document
  Tag Syntax
  Comments
  Spaces, Tabs, and New Lines Within Text
  Character and Entity References
Creating Links and Media Tags
  The A Element and the HREF Attribute
  The IMG Element and the SRC Attribute
  The APPLET Element
  The OBJECT Element
Text Structure and Highlighting
HTML Forms
  The FORM Tag
    HTML Form Components
    Input Tags
    Text Fields
    Submit Buttons
    Reset Button
    Checkboxes
    Radio Buttons
    Password
    Hidden Fields
    The SELECT Tag
    The TEXTAREA Tag
Creating HTML Tables
Advanced HTML
  JavaScript™ Programming Language
  CSS
  Frames

Quick Reference for HTTP
HTTP Overview
Request Structure
<table>
<thead>
<tr>
<th>HTTP Methods</th>
<th>Request Headers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Response Structure</td>
<td>Response Headers</td>
</tr>
<tr>
<td>Status Codes</td>
<td></td>
</tr>
<tr>
<td>CGI Overview</td>
<td>Environment Variables Set</td>
</tr>
<tr>
<td></td>
<td>Data Formatting</td>
</tr>
</tbody>
</table>

**Quick Reference for the Sun Java System Application Server**
- Sun Java System Application Server Overview
- Installing the Sun Java System Application Server
- Starting and Stopping the Application Server
- Application Configuration
- Logging and Log Files

**Quick Reference for the Ant Tool**
- Ant Overview
- Build File Structure
  - Projects
  - Targets
  - Tasks
  - Properties
- Ant Special Features
  - Patterns
  - The `fileset` Element
  - Filtering
  - The `filter` Task
- Basic Built-in Ant Tasks
  - The `copy` Task
  - The `delete` Task
  - The `mkdir` Task
  - The `echo` Task
  - The `javac` Task
  - The `javadoc` Task
  - The `jar` Task
- Complete Ant Build File
- Executing Ant
- Installing Ant

**Quick Reference for UML**
- UML Fundamentals
- General Elements
  - Packages
  - Stereotypes
  - Annotation
  - Constraints
  - Tagged Values
- Use Case Diagrams
- Class Diagrams
  - Class Nodes
Inheritance
Interface Implementation
Association, Roles, and Multiplicity
Aggregation and Composition
Association Classes
Other Association Elements
Object Diagrams
Collaboration Diagrams
Sequence Diagrams
Statechart Diagrams
Transitions
Activity Diagrams
Component Diagrams
Deployment Diagrams

Project Work

5. **Fee:**
   Rs. 15000/- (exclusive of Service Tax)