

HOMTECH DIPLOMA PROGRAM SECURED PROGRAMMING IN JAVA COURSE CONTENT

MODULE: PROGRAMMING IN JAVA (CORE)

UNIT 1: INTRODUCTION

- THE BEGINNING OF JAVA
- THE FAMILY HISTORY OF JAVA
- JAVA AS A SUCCESSOR TO C++
- WHAT IS THE JAVA TECHNOLOGY?
- THE JAVA VIRTUAL MACHINE
- 1.6 WORKING OF JIT (JUST IN TIME) COMPILER
- FEATURES OF JAVA

UNIT 2: JAVA LANGUAGE FUNDAMENTAL

- JAVA TOKENS
- PRIMITIVE DATA TYPES
- EACH DATA TYPE WITH DETAILED EXAMPLE
- A SIMPLE PROGRAM IN JAVA
- VARIABLES, DECLARATIONS, AND ASSIGNMENTS
- PATH SETTINGS
- TOOLS OF JDK
- OPERATORS
- KEYWORDS
- COMMAND LINE ARGUMENT
- ARRAY

UNIT 3: OOPS CONCEPTS

- CLASSES AND OBJECT
- METHOD AND INSTANCE VARIABLE
- POLYMORPHISM, STATIC VS DYNAMIC BINDING
- INSTANCE VARIABLE HIDING
- LEARNING ABOUT STATIC
- ABSTRACT CLASSES AND INTERFACES
- ACCESS SPECIFIER IN JAVA
- CREATING AND USING PACKAGES
- THE STATIC IMPORT STATEMENT

UNIT 4: EXCEPTION HANDLING

- FUNDAMENTAL OF EXCEPTION HANDLING
- CHECKED AND UNCHECKED EXCEPTION
- CHECKED VS. UNCHECKED EXCEPTIONS

- HANDLING EXCEPTION
- TRY, CATCH AND FINALLY
- USING THROW AND THROWS

UNIT 5: MULTI-THREADING

- THE JAVA THREAD MODAL
- THREAD PRIORITIES
- CREATING THREAD
- STARTING THE THREAD
- BASIC CONTROL OF THREADS
- MULTITHREADING AND CONTEXT SWITCHING
- INTER-THREAD COMMUNICATION

UNIT 6: FILE HANDLING (I/O STREAMS)

- I/O FUNDAMENTALS
- FILE HANDLING
- STREAMS
- A FILE INPUT EXAMPLE IS
- FILE OUTPUT EXAMPLE
- CHARACTER STREAMS
- API OVERVIEW
- BINARY STREAMS
- CLASSES HIERARCHY IN BINARY STREAMS

UNIT 7: COLLECTION FRAMEWORK

- INTRODUCTION
- WHAT IS A COLLECTIONS FRAMEWORK?
- BENEFITS OF THE JAVA COLLECTIONS FRAMEWORK
- IN SHORT COLLECTIONS API
- COLLECTION APIS HIERARCHY
- ITERATION INSIDE COLLECTION
- ENHANCED FOR LOOP
- WORKING WITH MAP BASED COLLECTION
- USE OF COMPARATOR
- THE COLLECTION ALGORITHM
- LEGACY CLASSES AND INTERFACES
- IMPLEMENTATIONS
- CUSTOM IMPLEMENTATION
- DIFFERENT PACKAGES USED IN LIST , SET, MAP INTERFACES
- HOW THE COLLECTIONS FRAMEWORK INTEROPERATES WITH OLDER APIS

UNIT 8: NETWORKING

- SOCKETS
- SECURE SOCKETS CUSTOM SOCKETS
- UDP DATAGRAMS
- MULTICAST SOCKETS

- URL CLASSES
- READING DATA FROM THE SERVER
- WRITING DATA – CONFIGURING THE CONNECTION
- READING THE HEADER – TELNET APPLICATION
- JAVA MESSAGING SERVICES

UNIT 9: APPLET AND SWING

- APPLET BASICS
- APPLET LIFE CYCLE
- SIMPLE APPLET DESIGNING
- INTRODUCTION TO SWING
- LAYOUT SETTING

UNIT 10: EVENT HANDLING

- WHAT IS AN EVENT?
- DELEGATION MODEL
- A LISTENER EXAMPLE
- THE EVENTHANDLER CLASS
- LISTENERS SUPPORTED BY SWING COMPONENTS
- LISTENERS THAT ALL SWING COMPONENTS SUPPORT
- OTHER LISTENERS THAT SWING COMPONENTS SUPPORT
- HOW TO WRITE AN ACTION LISTENER

UNIT 11: JDBC (JAVA DATABASE CONNECTIVITY)

- ROLE OF JDBC
- JDBC ARCHITECTURE
- ESTABLISHING A CONNECTION
- MAKING THE CONNECTION
- JDBC-ODBC BRIDGE DRIVER
- RESULTSET AND METADATA

MODULE: J2EE ARCHITECTURE AND JSP

UNIT 1: J2EE ARCHITECTURE

- OBJECTIVE
- JAVA EE APPLICATION MODEL
- DISTRIBUTED MULTITIERED APPLICATIONS
- SECURITY
- JAVA EE COMPONENTS
- JAVA EE CLIENTS
- THE JAVABEANS™ COMPONENT ARCHITECTURE
- JAVA EE SERVER COMMUNICATIONS
- ENTERPRISE INFORMATION SYSTEM TIER
- WEB SERVICES SUPPORT
- XML
- SOAP TRANSPORT PROTOCOL

- WSDL STANDARD FORMAT
- JAVA EE APPLICATION ASSEMBLY AND DEPLOYMENT
- PACKAGING APPLICATIONS
- JAVA EE 5 APIS
- ENTERPRISE JAVABEANS TECHNOLOGY
- JAVA NAMING AND DIRECTORY INTERFACE
- JAVA AUTHENTICATION AND AUTHORIZATION SERVICE
- SIMPLIFIED SYSTEMS INTEGRATION
- GETTING STARTED WITHWEB APPLICATIONS
- WEB APPLICATION LIFE CYCLE
- CREATING A DATA SOURCE IN THE APPLICATION SERVER
- FURTHER INFORMATION ABOUT WEB APPLICATIONS

UNIT 2: UNDERSTANDING XML

- INTRODUCTION TO XML
- WHAT IS XML?
- TAGS AND ATTRIBUTES
- COMMENTS IN XML FILES
- THE XML PROLOG
- PROCESSING INSTRUCTIONS
- WHY IS XML IMPORTANT?
- HOW CAN YOU USE XML?

UNIT 3: GENERATING XML DATA

- WRITING A SIMPLE XML FILE
- WRITING THE DECLARATION
- ADDING A COMMENT
- DEFINING THE ROOT ELEMENT
- ADDING ATTRIBUTES TO AN ELEMENT
- ADDING NESTED ELEMENTS
- ADDING HTML-STYLE TEXT
- ADDING AN EMPTY ELEMENT
- THE FINISHED PRODUCT
- WRITING PROCESSING INSTRUCTIONS
- INTRODUCING AN ERROR
- SUBSTITUTING AND INSERTING TEXT
- PREDEFINED ENTITIES
- CHARACTER REFERENCES
- USING AN ENTITY REFERENCE IN AN XML DOCUMENT
- HANDLING TEXT WITH XML-STYLE SYNTAX
- CREATING A DOCUMENT TYPE DEFINITION
- BASIC DTD DEFINITIONS
- DEFINING TEXT AND NESTED ELEMENTS
- LIMITATIONS OF DTDS
- SPECIAL ELEMENT VALUES IN THE DTD
- DOCUMENTS AND DATA 3.23 DEFINING ENTITIES IN THE DTD

- REFERENCING EXTERNAL ENTITIES
- SUMMARIZING ENTITIES
- USING A MIME DATA TYPE
- THE ALTERNATIVE: USING ENTITY REFERENCES
- CONDITIONAL SECTIONS
- RESOLVING A NAMING CONFLICT
- USING NAMESPACES
- DEFINING A NAMESPACE IN A DTD
- DESIGNING AN XML DATA STRUCTURE
- FORCED CHOICES
- STYLISTIC CHOICES
- NORMALIZING DATA
- NORMALIZING DTDS

UNIT 4: EXAMINING XML PARSERS

- DOM PARSERS
- SAX PARSERS
- DOM VERSUS SAX
- IMPLEMENTING XML DTDS
- REVIEWING XML FUNDAMENTALS
- TRAVELINFO_INTERNALDTD.XML
- UNDERSTANDING XML NAMESPACES
- DELIVERING THE MESSAGE
- THE NEW XML DOCUMENT
- EXPLORING XML SCHEMA
- WORKING WITH EXTENSIBLE STYLESHEET LANGUAGE TRANSFORMATIONS (XSLT)
- PRODUCING SIMPLE HTML WITH XSLT
- RESULTING HTML
- PRODUCING A WIRELESS MARKUP LANGUAGE (WML) DOCUMENT WITH XML

UNIT 5: JAVA API FOR XML PROCESSING

- THE JAXP APIS
- AN OVERVIEW OF THE PACKAGES
- THE SIMPLE API FOR XML APIS
- SAX PARSER FACTORY
- THE DOCUMENT OBJECT MODEL APIS
- THE DOM PACKAGES
- THE EXTENSIBLE STYLESHEET LANGUAGE TRANSFORMATIONS APIS
- THE XSLT PACKAGES

UNIT 6: JAXB BINDING BETWEEN XMLSCHEMA AND JAVA CLASSES

- JAXB ARCHITECTURE
- THE JAXB BINDING PROCESS
- REPRESENTING XML CONTENT
- BINDING XML SCHEMAS
- JAVA-TO-SCHEMA

- CUSTOMIZING JAXB BINDINGS
- EXAMPLES
- GENERAL USAGE INSTRUCTIONS
- BASIC JAXB EXAMPLES
- CONFIGURING AND RUNNING THE SAMPLES
- JAXB COMPILER OPTIONS
- SCHEMA-DERIVED JAXB CLASSES
- BASIC EXAMPLES
- UNMARSHAL VALIDATE EXAMPLE
- CUSTOMIZING JAXB BINDINGS
- WHY CUSTOMIZE?
- CUSTOMIZATION OVERVIEW
- PASSING CUSTOMIZATION FILES TO THE JAXB BINDING COMPILER
- SCOPE, INHERITANCE, AND PRECEDENCE
- DATATYPE CONVERTER EXAMPLE
- CLASS DECLARATIONS

UNIT 7: JAVA SERVLETS

- WHAT IS A SERVLET?
- SERVLET LIFE CYCLE
- HANDLING SERVLET LIFE-CYCLE EVENTS
- CONSTRUCTING RESPONSES
- MAINTAINING CLIENT STATE
- ASSOCIATING OBJECTS WITH A SESSION
- SESSION MANAGEMENT
- SESSIONTRACKING
- FINALIZING A SERVLET
- TRACKING SERVICE REQUESTS
- NOTIFYING METHODS TO SHUT DOWN

UNIT 8: SECURITY

- INTRODUCTION
- HTTP AUTHENTICATION
- RETRIEVING AUTHENTICATION INFORMATION
- FORM-BASED CUSTOM AUTHORIZATION
- DIGITAL CERTIFICATES
- SECURE SOCKETS LAYER (SSL)
- SSL CLIENT AUTHENTICATION
- RETRIEVING SSL AUTHENTICATION INFORMATION
- RUNNING SERVLETS SECURELY
- THE SERVLET SANDBOX
- ACCESS CONTROLLERS

UNIT 9: INTERNATIONALIZATION

- WESTERN EUROPEAN LANGUAGES
- HTML CHARACTER ENTITIES

- UNICODE ESCAPES
- CONFORMING TO LOCAL CUSTOMS
- NON-WESTERN EUROPEAN LANGUAGES
- WRITING ENCODED OUTPUT
- READING AND WRITING ENCODED OUTPUT
- MULTIPLE LANGUAGES
- UCS-2 AND UTF-8
- DYNAMIC LANGUAGE NEGOTIATION
- RECEIVING MULTILINGUAL INPUT
- APPLLET-SERVLET COMMUNICATION
- SERVLET FILTERING
- AN OVERVIEW OF THE API
- A BASIC FILTER
- MAPPING A FILTER WITH A URL
- A LOGGING FILTER
- FILTER CONFIGURATION
- A FILTER THAT CHECKS USER INPUT

UNIT 10: WHAT'S NEW IN THE SERVLET 2.4 SPECIFICATION

- SERVLET API QUICK REFERENCE
- GENERIC SERVLET
- CLASS SUMMARY
- CONSTRUCTORS
- INSTANCE METHODS
- SERVLET
- INTERFACE DECLARATION
- METHODS
- SERVLET CONFIG
- SERVLETCONTEXT
- SERVLETEXCEPTION
- SERVLET INPUT STREAM
- CONSTRUCTORS 10.14 INSTANCE METHODS
- SERVLETOUTPUTSTREAM

UNIT 11: JSP (JAVA SERVER PAGES)

- WHAT IS A JSP PAGE?
- A SIMPLE JSP PAGE EXAMPLE
- THE LIFE CYCLE OF A JSP PAGE
- HANDLING JSP PAGE ERRORS
- CREATING STATIC CONTENT
- OPERATORS 11.7 CREATING AND USING A JAVABEANS COMPONENT
- SETTING JAVABEANS COMPONENT PROPERTIES
- RETRIEVING JAVABEANS COMPONENT PROPERTIES
- USING CUSTOMTAGS
- DECLARINGTAG LIBRARIES
- INCLUDING THETAG LIBRARY IMPLEMENTATION

- REUSING CONTENT IN JSP PAGES
- TRANSFERRING CONTROL TO ANOTHER WEB COMPONENT
- INCLUDING AN APPLLET
- CREATING A JSP DOCUMENT
- USING THE JSP: ROOT ELEMENT
- USING THE JSP: OUTPUT ELEMENT
- JAVASERVER PAGES STANDARD TAG LIBRARY
- USING JSTL
- TAG COLLABORATION
- E TAG LIBRARY
- VARIABLE SUPPORT TAGS
- FLOW CONTROL TAGS
- SQL TAG LIBRARY
- CUSTOMTAGS IN JSP PAGES
- WHAT IS A CUSTOMTAG?
- THE EXAMPLE JSP PAGES
- TYPES OF TAGS
- TAGS WITH ATTRIBUTES
- TAGS THAT DEFINE VARIABLES
- COMMUNICATION BETWEEN TAGS
- ENCAPSULATING REUSABLE CONTENT USING TAG FILES
- CUSTOM TAG EXAMPLES
- TAG LIBRARY DESCRIPTORS
- PROGRAMMING SIMPLE TAG HANDLERS
- INCLUDING TAG HANDLERS IN WEB APPLICATIONS
- HOW IS A SIMPLETAG HANDLER INVOKED
- TAG HANDLERS FOR BASIC TAGS
- TAG EXTRA INFO CLASS
- COOPERATING TAGS
- A TEMPLATE TAG LIBRARY

UNIT 12: HTML 5.0

- HISTORY OF HTML
- WHAT IS HTML?
- WHAT IS HTML 5.0?
- FEATURES OF HTML 5.0
- CANVAS IN HTML 5
- AUDIO AND VIDEO IN HTML 5
- HTML 5 APIs
- BETTER INTERACTIONS IN HTML 5

UNIT 13: CSS JAVA SCRIPT

- What is CSS?
- FEATURES OF CSS
- WHAT IS JAVASCRIPT?
- THINGS YOU CAN DO WITH THE JAVASCRIPT

- JAVASCRIPT FORM VALIDATION

MODULE: STRUCTURED QUERY LANGUAGE IN JAVA

UNIT 1: SQL SERVER 2005

- WHAT IS SQL SERVER 2005?
- OVERVIEW OF SQL SERVER 2005
- FEATURES OF SQL SERVER 2005
- NOTIFICATION SERVICES
- CONNECTING TO AND MANAGING YOUR SQL SERVERS
- ASYNCHRONOUS TREEVIEW AND OBJECT FILTERING
- TEMPLATE EXPLORER
- RESULTS PANE
- SUMMARY VIEWS

UNIT 2: INTRODUCTION TO SQL STATEMENT

- SQL DATABASE TABLES
- SQL DATA MANIPULATION LANGUAGE (DML)
- THE RESULT SET
- SEMICOLON AFTER SQL STATEMENTS?
- THE SELECT DISTINCT STATEMENT
- SQL WHERE CLAUSE
- USING QUOTES
- SQL INSERT INTO STATEMENT
- SQL UPDATE STATEMENT
- SQL DELETE STATEMENT
- SQL TRY IT
- SQL ORDER BY
- SQL AND & OR
- SQL IN
- SQL BETWEEN, BETWEEN ... AND
- SQL ALIAS
- SQL JOIN
- USING THE UNION ALL COMMAND
- SQLCREATE DATABASE, TABLE, AND INDEX
- SQL DROP INDEX, TABLE AND DATABASE
- TRUNCATE A TABLE
- SQL ALTER TABLE
- SQL FUNCTIONS
- AGGREGATE FUNCTIONS IN SQL SERVER
- SQL GROUP BY AND HAVING
- HAVING
- SELECT INTO STATEMENT
- SQL CREATE VIEW STATEMENT
- QUICK SQL REFERENCE

UNIT 3: SQL IS A STANDARD - BUT

- SYSTEM DATABASES IN SQL SERVER 2005
- CATEGORIZATION OF SQL STATEMENT
- INTERACTIVE SQL
- EXECUTING SQL STATEMENTS
- SQL'S DATA TYPES

UNIT 4: T-SQL ENHANCEMENTS FOR DEVELOPERS

- ENHANCEMENTS AFFECTING DML
- OLD-STYLE OUTER JOINS DEPRECATED
- COMMON TABLE EXPRESSIONS
- SIMPLIFYING COMPLEX QUERIES
- USING CTES FOR RECURSIVE QUERIES
- EXTENSIONS TO THE FROM CLAUSE
- NEW JOIN TYPES
- RANDOM DATA SAMPLING
- EXCEPT AND INTERSECT

UNIT 5: T-SQL ENHANCEMENTS FOR DBAS

- METADATA VIEWS
- DYNAMIC MANAGEMENT VIEWS AND FUNCTIONS
- DDL TRIGGERS
- CREATING AND ALTERING DDL TRIGGERS
- DROPPING DDL TRIGGERS
- ENABLING AND DISABLING DDL TRIGGERS
- ENUMERATING DDL TRIGGERS USING CATALOG VIEWS
- INDEXING AND PERFORMANCE ENHANCEMENTS
- TABLE 5.10 PARTITION FUNCTIONS
- PARTITION SCHEMES
- CREATING PARTITIONED TABLES AND INDEXES
- ADDING AND REMOVING PARTITIONS
- ENHANCEMENTS TO TABLES AND VIEWS
- ENHANCEMENTS TO INDEXED VIEWS
- PERSISTED COMPUTED COLUMNS
- VERIFYING A DATABASE'S PAGES AND INDEX PARTITIONING

MODULE: REMOTE METHOD INVOCATION AND ENTERPRISE JAVA BEANS

UNIT 1: REMOTE METHOD INVOCATION

- INTRODUCTION
- AN OVERVIEW OF RMI APPLICATIONS
- ADVANTAGES OF DYNAMIC CODE LOADING
- REMOTE INTERFACES, OBJECTS, AND METHODS
- CREATING DISTRIBUTED APPLICATIONS BY USING RMI
- DESIGNING AND IMPLEMENTING THE APPLICATION COMPONENTS
- BUILDING A GENERIC COMPUTE ENGINE

- WRITING AN RMI SERVER
- IMPLEMENTING A REMOTE INTERFACE
- DECLARING THE REMOTE INTERFACES BEING IMPLEMENTED
- CREATING AND INSTALLING A SECURITY MANAGER
- MAKING THE REMOTE OBJECT AVAILABLE TO CLIENTS
- COMPILING AND RUNNING THE EXAMPLE
- BUILDING A JAR FILE OF INTERFACE CLASSES
- STARTING THE SERVER
- STARTING THE CLIENT
- PARAMETERS IN RMI
- PASSING NON-REMOTE OBJECTS
- REFERENTIAL INTEGRITY
- PARAMETER TRANSMISSION
- PARAMETERS IN A SINGLE JVM
- PRIMITIVE PARAMETERS 37
- OBJECT PARAMETERS 37
- REMOTE OBJECT PARAMETERS 37
- DISTRIBUTING AND INSTALLING RMI SOFTWARE
- DISTRIBUTING RMI CLASSES
- AUTOMATIC DISTRIBUTION OF CLASSES
- FIREWALL ISSUES
- DISTRIBUTED GARBAGE COLLECTION
- MOBILE AGENT ARCHITECTURES

UNIT 2: EJB (ENTERPRISE JAVA BEANS)

- WHAT IS AN ENTERPRISE BEAN?
- ENTERPRISE JAVABEANS - AN INTRODUCTION
- WHEN TO USE ENTERPRISE BEANS
- WHY EJBS?
- PORTABLE COMPONENT-BASED ARCHITECTURE
- INDEPENDENCE FROM DATABASE SCHEMA
- MULTIPLE DATA SOURCES WITH TRANSACTIONAL CAPABILITIES
- MIDDLE-TIER ARCHITECTURE
- MULTIPLE CLIENT TYPES ACCESSING SHARED DATA
- CONCURRENT READ AND UPDATE ACCESS TO SHARED DATA
- METHOD-LEVEL OBJECT SECURITY
- MULTIPLE SERVERS TO HANDLE THROUGHPUT AND AVAILABILITY
- INTEGRATION WITH CORBA
- EJB SPECIFICATIONS
- ENTERPRISE JAVABEANS ARCHITECTURE
- LIMITATIONS
- NEW FEATURES SUPPORTED IN EJB 2.0
- CHANGES IN EJB 2.0
- LIMITATIONS
- GENERAL RESTRICTIONS OF EJB
- EJB 2.0 OVERVIEW

- CONTAINER-MANAGED PERSISTENCE (CMP)
- CONTAINER-MANAGED RELATIONSHIPS (CMR)
- EJB QUERY LANGUAGE
- EJB HOME METHODS
- MESSAGE-DRIVEN BEAN
- EJB ROLES
- EJB ARCHITECTURE
- TYPES OF EJBS
- MAIN EJB FRAMEWORK COMPONENTS
- EJB SERVER
- EJB CONTAINER
- PRIMARY SERVICES
- EJB COMPONENT (THE ACTUAL EJB)
- EJB TYPES
- EJB INTERFACES AND EJB BEAN
- LOCAL VERSUS REMOTE INTERFACES
- INHERITED COMPONENT INTERFACE METHODS
- CONSTRUCTING THE COMPONENT INTERFACE: BUSINESS METHODS
- EJB HOME INTERFACE: EJB HOME
- INHERITED HOME INTERFACE METHODS
- CONSTRUCTING THE HOME INTERFACE: LIFE-CYCLE METHODS
- EJB BEAN CLASS
- EJB DEPLOYMENT DESCRIPTOR
- EJB CLIENT VIEW
- CLIENT EJB INTERACTION
- USING THE HOME OBJECT TO CREATE OR FIND AN EJB OBJECT
- BEAN CLEANUP
- ENTERPRISE JAVABEANS CLIENTS INTRODUCTION
- TYPES OF ENTERPRISE BEANS
- DEFINING CLIENT ACCESS WITH INTERFACES
- DECIDING ON REMOTE OR LOCAL ACCESS
- THE CONTENTS OF AN ENTERPRISE BEAN
- NAMING CONVENTIONS FOR ENTERPRISE BEANS
- THE LIFE CYCLES OF ENTERPRISE BEANS
- THE LIFE CYCLE OF A STATEFUL SESSION BEAN
- THE LIFE CYCLE OF A STATELESS SESSION BEAN
- THE LIFE CYCLE OF A MESSAGE-DRIVEN BEAN
- GETTING STARTED WITH ENTERPRISE JAVA BEANS
- COMPILING AND PACKAGING THE CONVERTER EXAMPLE IN NETBEANS IDE
- CREATING THE CONVERTER APPLICATION CLIENT
- CODING THE CONVERTER APPLICATION CLIENT
- COMPILING THE CONVERTER APPLICATION CLIENT
- COMPILING THE CONVERTER WEB CLIENT
- GETTING STARTED WITH MESSAGE- DRIVEN ENTERPRISE BEANS
- THE BUSINESS INTERFACE
- SESSION BEAN CLASS

- LIFE-CYCLE CALLBACK METHODS
- HANDLING EXCEPTIONS
- A MESSAGE-DRIVEN BEAN EXAMPLE
- THE SIMPLE MESSAGE APPLICATION CLIENT
- THE MESSAGE-DRIVEN BEAN CLASS
- PACKAGING, DEPLOYING, AND RUNNING THE SIMPLE MESSAGE EXAMPLE
- ADDITIONAL CONCEPTS: TRANSACTIONS, EXCEPTIONS, SECURITY
- TRANSACTIONS
- TRANSACTION SUPPORT IN J2EE
- XA PROTOCOL
- TRANSACTION ATTRIBUTE METHOD RULES
- RESOURCE ACCESS INTENT
- HANDLING A ROLLBACK
- GUIDELINES FOR USING TRANSACTIONS
- EJB EXCEPTION HANDLING
- EXCEPTIONS AND EJBS
- GOALS OF EJB SYSTEM LEVEL EXCEPTION HANDLING
- REMOTE EXCEPTION VERSUS EJB EXCEPTION
- ADDITIONAL EXCEPTION HANDLING IN MDBS
- SECURITY
- GOALS OF EJB SECURITY
- JAVA AUTHENTICATION AND AUTHORIZATION SERVICES
- ASSIGNING METHOD-LEVEL PERMISSIONS
- BEST PRACTICE: AUTHORIZATION POLICY MATRIX

MODULE: PROGRAMMING WITH FRAMEWORKS

UNIT 1: JSF (JAVASERVER FACES)

- UNDERSTANDING FRONT-CONTROLLER DESIGN PATTERN
- CREATING USER INTERFACE USING JSF UI ELEMENTS
- DEFINING PAGE NAVIGATION USING FACES-CONFIG.XML
- WORKING WITH MANAGED/BACKING BEANS
- VALIDATING AND CONVERTING USER INPUT
- WORKING WITH EVENTS AND LISTENERS
- WORKING WITH INTERNATIONALIZATION
- AJAX AND JSF
- INTEGRATING BEAN VALIDATION AND JSF
- USING DATATABLE AND REPEAT
- TEMPLATING
- INTEGRATING HTML 5 WITH JSF
- INTEGRATING JQUERY WITH JSF
- UPLOAD FILES
- CONTROLLING FLOW WITH FACES FLOW

UNIT 2: WEB SERVICES

- WHAT IS A WEB SERVICE

- HOW WEB SERVICE USES SOAP, HTTP, XML AND WSDL
- UNDERSTANDING SOAP ENVELOPE AND MESSAGE
- CREATING A WEB SERVICE AND CONSUMING IT USING JAX-WS
- UNDERSTANDING XML MARSHALLING AND UNMARSHALLING
- UNDERSTANDING RESTFUL WEB SERVICES
- ACCESSING RESTFUL WEB SERVICES USING AJAX THROUGH JQUERY

UNIT 3: HIBERNATE FRAMEWORKS

- WHAT IS THE PROBLEM WITH JDBC - PARADIGM MISMATCH
- WHAT IS ORM?
- UNDERSTANDING DIFFERENT COMPONENTS OF HIBERNATE
- HOW TO PERSIST OBJECTS USING HIBERNATE
- HOW TO USE MAPPING FILES, CONFIGURATION FILES AND SESSION OBJECT
- INSTANCE STATES
- HOW TO GENERATE ID
- HOW TO IMPLEMENT INHERITANCE IN HIBERNATE
- WORKING WITH RELATIONSHIP BETWEEN ENTITIES
- TRANSACTIONS IN HIBERNATE
- QUERYING WITH HQL (HIBERNATE QUERY LANGUAGE)
- NAMED AND NATIVE QUERIES
- WORKING WITH CRITERIA INTERFACE
- QUERY BY EXAMPLE - QBE
- CACHING AND FETCHING
- USING HIBERNATE IN DIFFERENT TYPES OF APPLICATIONS

UNIT 4: STRUTS 2 TUTORIAL

- INTRODUCTION TO MVC ARCHITECTURE
- STRUTS2 ARCHITECTURE
- STRUTS2 COMPONENTS
- CONTROLLER
- ACTIONS
- INTERCEPTORS
- RESULT AND VIEW COMPONENTS
- VALUESTACK AND OGNL
- STRUTS TAG LIBRARY
- CONTROL TAGS
- DATA TAGS
- UI TAGS
- UNDERSTANDING ACTIONCONTEXT AND ACTIONINVOCATION
- IMPLEMENTING CUSTOM INTERCEPTORS
- IMPLEMENTING CUSTOM RESULT
- USE OF COMMON PREDEFINED INTERCEPTORS
- PARAMS
- EXCEPTION
- I18N
- EXECANDWAIT

- SERVLETCONFIG
- VALIDATION
- WORKFLOW
- MODELDRIVEN
- CUSTOM & PREDEFINED VALIDATIONS
- VALIDATABLE & VALIDATIONAWARE
- REQUIRED
- REQUIREDSTRING
- INT,DOUBLE & DATE
- EMAIL
- STRINGLENGTH
- REGEX
- INTERNATIONALIZATION
- GENERATING COMPOSITE VIEWS USING TILES
- TILES CONFIGURATION
- TILES TAG LIBRARY
- TILES RESULT
- TILES LAYOUT
- USING VELOCITY FOR GENERATING VIEWS
- USING ZERO CONFIGURATION PLUG IN
- STRUTS2 CONVENTIONS
- ANNOTATION BASED CONFIGURATION

UNIT 5: SPRING FRAMEWORK OVERVIEW

- INTRODUCTION TO SPRING 3.0
- STEPS TO USE SPRING FRAMEWORK IN APPLICATIONS
- UNDERSTANDING IOC AND DEPENDENCY INJECTION
- UNDERSTANDING THE BEAN LIFE-CYCLE – AUTO WIRING AND BEAN SCOPES
- ANNOTATION-BASED DEPENDENCY INJECTION
- ADDING BEHAVIOR TO AN APPLICATION USING ASPECTS – AOP
- CREATING AND APPLYING ASPECTS
- INTRODUCING DATA ACCESS WITH SPRING – JDBC THROUGH SPRING
- TRANSACTIONS IN A SPRING ENVIRONMENT
- GETTING STARTED WITH HIBERNATE IN A SPRING ENVIRONMENT
- WORKING WITH SPRING MVC
- SPRING MVC FORM HANDLING
- CREATING VIEWS IN SPRING MVC

MODULE: SECURE PROGRAMMING

UNIT 1: INTRODUCTION TO APPLICATION LEVEL ATTACKS

- UNDERSTANDING THE COMMON ATTACK VECTORS - PHISHING, DEFACEMENT, IDENTITY THEFT
- ATTACK SCENARIO: OS COMMAND INJECTION
- ATTACK SCENARIO: CROSS SITE SCRIPTING (XSS)
- ATTACK SCENARIO: PARAMETER TAMPERING
- ATTACK SCENARIO: AUTHENTICATION BREACH

- ATTACK SCENARIO: UNAUTHORIZED OPERATIONS AND ACCESS TO DATA

UNIT 2: JAVA SE RUNTIME SECURITY

- LANGUAGE AND JVM LEVEL SECURITY
- SECURITY MANAGER AND ACCESS CONTROLLER
- POLICIES AND PERMISSIONS
- PRIVILEGEDACTION
- GURADEDOBJECT

UNIT 3: SECURE CODING PRINCIPLES

- ATTACK SCENARIO: INTEGER OVERFLOWS
- SECURE STRING COMPARISON AND LEAST PRIVILEGE
- ATTACK SCENARIO: COMMAND INJECTION
- SECURE DEFAULTS AND INITIALIZATIONS
- INPUT VALIDATION
- ATTACK SCENARIO: SQL INJECTION
- AVOIDING HARD CODED SECRETS
- ATTACK SCENARIO: CANONICALIZATION
- STRONG TYPING AND SECURE COMPARISON
- CLIENT SIDE VALIDATION
- ATTACK SCENARIO: FORMAT STRINGS
- SECURE RESOURCE ACCESS AND ERROR HANDLING
- JAVA LOGGING TECHNOLOGIES (LOG4J, ESAPI)
- ATTACK SCENARIO: LOG INJECTION
- OPEN DESIGN PRINCIPLE, DEFAULT PRINCIPLE AND LEAST PRIVILEGED PRINCIPLE
- CLASS LEVEL SECURITY
- INTEGRATE SECURITY INTO THE DEVELOPMENT LIFECYCLE

UNIT 4: JAVA AUTHENTICATION & AUTHORIZATION (JAAS)

- INTRODUCTION TO JAAS SUBJECT AND PRINCIPAL
- PLUGGABLE AUTHENTICATION MODULE (PAM)
- LOGINCONFIGURATION , LOGINCONTEXT AND LOGINMODULE
- CALLBACKHANDLER AND CALLBACKS
- CODESOURCE DEPENDENT AND PRINCIPAL DEPENDENT AUTHORIZATION
- PROTECTIONDOMAIN AND ACCESSCONTROLCONTEXT
- PERMISSION AND POLICY
- APPLICATION INTERFACE
- SECURITY SYSTEM INTEGRATION
- LOGIN MODULES
- FORM AUTHENTICATION
- CORE CLASSES AND INTERFACES
- COMMON CLASSES
- SUBJECT
- AUTHENTICATION CLASSES AND INTERFACES
- LOGINCONTEXT
- LOGINMODULE

- CALLBACKHANDLER
- CALLBACK
- AUTHORIZATION CLASSES
- POLICY
- AUTHPERMISSION
- PRIVATECREDENTIALPERMISSION
- JAAS SETTINGS IN THE JAVA.SECURITY SECURITY PROPERTIES FILE
- LOGIN CONFIGURATION PROVIDER
- LOGIN CONFIGURATION URLS
- POLICY PROVIDER
- POLICY FILE URLS
- WHAT IS JAAS AUTHORIZATION?
- HOW IS JAAS AUTHORIZATION PERFORMED?
- HOW DO YOU MAKE PRINCIPAL-BASED POLICY FILE STATEMENTS?
- HOW DO YOU ASSOCIATE A SUBJECT WITH AN ACCESS CONTROL CONTEXT?
- THE AUTHORIZATION TUTORIAL CODE
- JAASAZN.JAVA
- SAMPLEACTION.JAVA
- THE LOGIN CONFIGURATION FILE
- THE POLICY FILE
- PERMISSIONS REQUIRED BY JAASAZN
- PERMISSIONS REQUIRED BY SAMPLEACTION
- THE FULL POLICY FILE
- RUNNING THE AUTHORIZATION TUTORIAL CODE

UNIT 5: INPUT VALIDATION

- WHAT IS CONSIDERED INPUT?
- THE NEED FOR INPUT VALIDATION
- BLACK LIST VS. WHITE LIST VALIDATION
- STRING MANIPULATION AND COMPARISON
- ATTACK SCENARIO: DIRECTORY TRAVERSAL
- DATA TYPE CONVERSION AND OUTPUT ENCODING
- REGULAR EXPRESSIONS
- VALIDATION IN STRUTS & JSF
- ATTACK SCENARIO: MALICIOUS FILE UPLOADS
- CLIENT SIDE VS. SERVER SIDE VALIDATIONS
- XML VALIDATION – DTD AND XML SCHEMA
- ATTACK SCENARIO: XPATH INJECTION
- SERVLET FILTERS

UNIT 6: ENCRYPTION IN JAVA

- SYMMETRIC ENCRYPTION IN JAVA
- SECURITY CONTEXT
- DESIGN PRINCIPLES
- ARCHITECTURE
- KEY MANAGEMENT

- JCA STRUCTURE
- SECURERANDOM
- MESSAGEDIGEST
- SIGNATURE
- DIGITAL SIGNATURE VS CRYPTOGRAPHY VS MESSAGEDIGEST
- SIGNED CERTIFICATE
- CIPHER CLASS
- STREAM VS. BLOCK CIPHER
- MODES OF OPERATION
- CREATING CIPHER OBJECT
- MESSAGE AUTHENTICATION CODES(MAC)
- RSA ENCRYPTION IN JAVA
- BASIC ALGORITHM AND TERMINOLOGY

UNIT 7: AUDITING & LOGGING

- WHICH EVENTS SHOULD BE LOGGED?
- ADDING A SECURITY LOG LEVEL IN LOG4J
- LOGGING WITH ESAPI
- LOG ENTRY STRUCTURE
- JAVA LOGGING TECHNOLOGIES
- ATTACK SCENARIO: LOG INJECTION
- EVENT MESSAGE STRUCTURE

UNIT 8: SECURING DATABASE CONNECTIONS

- WHAT IS DYNAMIC SQL?
- PROGRAMMING WITH DYNAMIC SQL
- EXECUTING DYNAMIC QUERIES
- PREVENTING SQL INJECTION IN JAVA
- EXAMPLE OF SQL INJECTION
- PARAMETERIZED QUERIES
- PREPARED STATEMENTS
- VARIABLE BINDING
- DYNAMIC QUERIES VIA STRING CONCATENATION

UNIT 9: PLATFORM SECURITY

- INTRODUCTION
- THE ORIGINAL SANDBOX MODEL
- EVOLVING THE SANDBOX MODEL
- CONNECTING TO SQL SERVER
- TO CONFIGURE IIS FOR WINDOWS INTEGRATED AUTHENTICATION
- TO CONFIGURE WEB.CONFIG TO IMPERSONATE THE IDENTITY SUPPLIED BY IIS
- TO CONFIGURE CONNECTION STRINGS FOR WINDOWS INTEGRATED SECURITY
- TO CONFIGURE SQL SERVER FOR WINDOWS INTEGRATED SECURITY

UNIT 10: SECURE SESSION & STATE MANAGEMENT

- USING SESSIONS

- THE SESSIONMANAGER
- SESSION TIMEOUT
- SESSION LISTENERS
- SESSION STORAGE
- WEB APPLICATIONS
- SESSION VALIDATION & SCHEDULING
- SESSION CLUSTERING
- ENTERPRISECACHESESSIONDAO
- EHCACHE + TERRACOTTA
- ZOOKEEPER
- SESSIONS AND SUBJECT STATE
- STATEFUL APPLICATIONS (SESSIONS ALLOWED)
- STATELESS APPLICATIONS (SESSIONLESS)
- A HYBRID APPROACH
- WEB APPLICATIONS

UNIT 11: CODE PROTECTION

- DECOMPILE JAVA CLASS FILE
- PROTECT YOUR JAVA CODES FROM DECOMPILERS
- SIGNING JAR FILES
- EXAMPLE
- SEALING PACKAGES WITHIN A JAR FILE
- AN EXAMPLE
- SEALING JAR FILES
- CLASS SIGNEDOBJECT
- CLASS SEALEDOBJECT

UNIT 12: SDL (SECURE DEVELOPMENT LIFECYCLE)

- THREAT RISK MODELING
- PERFORMING THREAT RISK MODELING USING THE MICROSOFT THREAT MODELING PROCESS
- IDENTIFY SECURITY OBJECTIVES
- APPLICATION OVERVIEW
- DECOMPOSE APPLICATION
- IDENTIFY THREATS
- STRIDE
- DREAD