

DIIPP

DUCAT INTEGRATED INDUSTRIAL PROFESSIONAL PROGRAM

DUCAT'S 100 DAYS - 500 HRS "INTEGRATED PROGRAM"



A DIIPP "Ducat Integrated Industrial Professional Program" is need of the hour for any fresher. The course has been designed to enable you to crack all the rounds (Mathematical & Technical Aptitude, Group Discussion, Programming, Technical interview and HR interview) involved in placement process.

PROGRAMMING & PROBLEM SOLVING

» Introduction

- Declaration & Initialization
- Floating Point Issues, Control Instructions,
- Structure, Union Enum, Functions,
- C Pre Processors, Pointers, Arrays, Strings,
- Expressions, Input/Output, Extern,
- Bitwise Operators, Typedef, Const, Static,
- Memory Allocation, Command Line Augments
- Variable No. of Arguments, Complicated Declarations
- C PROGRAMMING
- Introduction, main() function
- if statement, if and else
- if, else if and else, switch statments

» Loops

- while loop
- do-while loop
- for loop
- break
- continue

» Strings

- String Basics
- Program to find the Length of the String using builtin function
- Program to find the length of the String with out using built in function
- Program to find the length of the String using while loop
- Program to find the length of the String using do-while loop
- Program to find the length of the String using for loop
- Program to copy one String into another string
- Program to reverse a String using global variables
- Program to reverse a String using local variables
- Program to compare two String without using builtin functions
- Program to check whether string is palindrome or not
- Program to concatenate two strings
- Program to convert a string from lower case to upper case
- Program to print the ASCII equivalent of alphabets in a string
- Program to count the number of Vowels,Consonants and special symbols in a string
- Program to replace a Vowel with a special symbol

» Pointers

- Pointer basics, Reading different types of pointer
- Pointer creation, Pointer usage, Pointer to Strings
- Program to copy one string to another using pointer
- Parameter Passing Techniques in C
- Pass by value, Pass by reference
- Memory allocation techniques
- Static memory allocation
- Dynamic memory allocation

TECHNICAL APTITUDE

» DATA STRUCTURES

» INTRODUCTION

- Array Basics, Stack Basics
- Difference between Arrays and Stacks
- Program to Push, Pop and Display operation on Stacks
- Queue basics
- Program to Insert an element at rear of Queue
- Program to Delete an element at the beginning of Queue
- Program to display elements in Queue
- Circular Queue
- Program to Insert an element in a circular queue
- Program to delete an element in a circular queue
- Program to display elements in a circular queue
- Program to insert, delete and display elements in a circular queue using pointers
- Structures
- Double Ended Queue
- Polish Notation

» Singly Linked Lists

- Singly LinkedLists
- Basics of singly LinkedList
- Program to insert an element at front in singly linked list
- Program to insert an element at rear in singly linked list
- Program to delete an element from front in singly linked list
- Program to delete an element from rear in singly linked list
- Program to display elements in a singly linked list
- Program to insert an element at use is specific position
- Program to delete an element from user's specific position
- Sort linked list
- Reverse link list

» Doubly Linked Lists

- Basics of Doubly LinkedList
- Program to insert an element at front in doubly linked list
- Program to insert an element at rear in doubly linked list
- Program to delete an element from front in doubly linked list
- Program to delete an element from rear in doubly linked list
- Program to display elements in a doubly linked list
- Program to insert an element at use is specific position
- Program to delete an element from user's specific position

» Circular Linked Lists

- Circular Linked Lists, Basics of Circular LinkedList
- Program to insert an element at front in Circular linked list
- Program to insert an element at rear in Circular linked list
- Program to delete an element from front in Circular linked list

- Program to delete an element from rear in Circular linked list
- Program to display elements in a Circular linked list
- Program to insert an element at use is specific position
- Program to delete an element from user's specific position

» **Stacks and queues**

- Program to concatenate two linked lists
- Program to count the number of nodes in the linked list
- Program to add the data nodes of in the linked list
- Program to reverse a linked list
- Program to search an element in a linked list
- Program to delete a node in the linked list
- Program to insert an element towards the left of data node in linked list
- Program to insert an element towards the right of data node in linked list
- Program to find minimum element in the linked list
- Program to find maximum element in the linked list
- Program to count the number of even and odd number of elements in a linked list

» **Static and Dynamic memory Location**

- Program to compare two linked lists
- Program to delete duplicate element in a linked list

» **Tree Basics**

- Pre order traversal, Post order traversal, In order traversal
- Program to create a Binary Search Tree
- Program to find a minimum element in a Binary Search Tree
- Program to find a maximum element in a Binary Search Tree
- Binary Search Tree
- Threaded Binary Search Tree
- Heep Tree

SQL

» **Introduction to SQL**

- Categorize the different types of SQL statements
- Software Installation
- Database Creation
- Log on to the database.

» **Retrieve Data using the SQL SELECT Statement**

- Select Table Data With All Columns
- Select Table Data With Specific Columns
- Use Arithmetic Operators
- Use Concatenation Operators
- Use Column Alias
- DESCRIBE command to display the table structure
- Use of Distinct

» **Learn to Restrict and Sort Data**

- Use WHERE clause to limit the output retrieved
- Between , In() , Is Null and Like
- ORDER BY clause to sort Data

» **Use of DDL Statements to Create and Manage Tables**

- Create a simple table
- Alter and Truncate Table
- Drop Table and Concept of Recyclebin

» **Data Manipulation Statements**

- Describe each DML statement

- Insert rows into a table
- Change rows in a table by the UPDATE statement
- Delete rows from a table with the DELETE statement
- Save and discard changes with the COMMIT and ROLLBACK statements

» Constraints

- Stop Entry of Invalid Data Through CONSTRAINTS
- Add Primary Key , Foreign Key
- Not Null , Unique and Check Constraints

» Usage of Functions to Customize Output

- Differences between single row and multiple row functions
- Manipulate strings with character function in the SELECT and WHERE clauses
- Manipulate numbers with the ROUND, TRUNC, and MOD functions
- Manipulate dates with the DATE functions
- ROWNUM and ROWID

» Invoke Conversion Functions and Conditional Expressions

- Describe implicit and explicit data type conversion
- Use the TO_CHAR, TO_NUMBER, and TO_DATE conversion functions
- Nest multiple functions
- Use of Case Expression

» Aggregate Data Using the Group Function

- Use the aggregation functions to produce meaningful reports
- Divide the retrieved data in groups by using the GROUP BY clause
- Exclude groups of data by using the HAVING clause

» Display Data From Multiple Tables Using Joins

- Write SELECT statements to access data from more than one table
- View data that does not meet join condition by using outer joins
- Join a table to itself by using a self-join

» Use Sub-queries to Solve Queries

- Describe the types of problem that sub-queries can solve
- Define sub-queries
- Write single-row and multiple-row sub-queries
- Use of Exists

» Other Schema Objects

- Create a View
- Retrieve data from views
- Create, maintain, and use sequences
- Create and maintain indexes
- Create synonyms

» Control User Access

- Create Users
- System Privileges & Object Privileges
- Granting Privileges
- Manage Privileges Through Role
- Revoke Object Privileges

JAVA

» INTRODUCTION TO JAVA

- Why Java was Developed
- Application Areas of Java
- History of Java
- Platform Independency in Java
- USP of Java: Java Features
- Sun-Oracle Deal
- Different Java Platforms
- Difference between JDK,JRE,JVM
- Java Versions
- JVM Architecture
- Installing Java on Windows
- Understanding Path Variable: Why Set Path

» CREATING FIRST JAVA PROGRAM

- Understanding Text Editors to Write Programs
- How to compile java file
- Byte Code and class file
- How to run class file

» JAVA LANGUAGE FUNDAMENTALS

- Identifiers
- Keywords
- Variables
- Literals
- Data Types
- Operators
- Comments
- Looping Statements
- Condition Statements
- Type Casting

» OOP IMPLEMENTATION (PIE)

- Why OOP
- OOP Concepts with Real life examples
- Class& it's Syntax
- Object& it's Syntax
- Reference Variable
- Constructors
- Instance(Non-Static)& Static Variables
- Instance(Non-Static) & Static Methods
- this Keyword and it's usages
- Object & Static Initializers(Anonymous Blocks)
- Understanding '+' Operator
- Inheritance& it's Syntax
- Types of Inheritance
- Object Class as Root of Java Class Hierarchy
- Variable Hiding
- Method Hiding
- Method Overriding
- Method Overloading
- Super keyword and it's usages
- Final keyword and it's usages
- Constructor Chaining

- Upcasting and Downcasting
- Static & Dynamic Binding
- Run Time Polymorphism
- Abstract Keyword (Abstract classes and methods)
- Understanding Interfaces
- Implementation of Encapsulation
- Association with Implementation

» PACKAGES

- Understanding Packages
- Setting Class path
- Reading Input from Keyboard
- Access Modifiers

» NESTED TYPES

- Static Nested Class
- Non-static Nested Class
- Local Class
- Anonymous Class
- Nested Interface

» ARRAYS

- General Definition of Array
- Advantages from Array
- Arrays in Java
- 1-d Arrays
- 2-d Arrays
- Jagged Arrays
- Array of reference type
- Operations on Arrays

» COMMAND LINE ARGUMENTS AND WRAPPER CLASSES

- How to read command line arguments
- Wrapper Classes
- Parsing of Numeric Strings
- String representation of Primitives

» EXCEPTION HANDLING

- Types of Runtime Errors
- Understanding Exceptions
- Exception Class Hierarchy
- Try & Catch Blocks
- Patterns of Catch Block
- Nested Try statements
- Throw, throws and finally
- Creating Custom Exceptions
- Checked & Unchecked Exceptions
- Assertion

» WORKING WITH STRINGS

- What is String
- String Class
- Creating String Object
- Operations on String
- String Buffer Class and its Methods
- Difference between String and StringBuffer class
- String Builder Class and its Methods
- Difference between StringBuffer and StringBuilder

» SWING

- Introduction to AWT
- Introduction to Swing Components
- Look And Feel of Swing Components
- MVC Architecture of Swing Components
- Working with Image
- Advance Swing Components
 - JOptionPane,JTree,JTable,JTabbedPane
 - JFileChooser,JcolorChooser
- Menu Components
 - JMenu
 - JMenuItem
 - JMenuBar

» MULTITHREADED PROGRAMMING

- Multitasking: Why Concurrent Execution
- Multiprocessing v/s Multithreading
- Main Thread (Default Java Thread)
- Creating Child Threads and understanding context switching
- Thread States
- Thread Group
- Thread Synchronization: Methods and Blocks
- Inter-Thread communication
- Daemon Threads
- Deadlock

» I/O STREAMS

- What is I/O
- Why Need Streams
- Byte Streams and Character Streams
- Read/Write operations with file
- Scanner Class
- Object Serialization& Deserialization
- Transient keyword
- File Class and it's Methods

» SOCKET PROGRAMMING

- Understanding Fundamentals of a Network
- Socket and ServerSocket Classes
- InetAddress Class
- DatagramSocket and DatagramPacket Classes
- URL,URLConnection,HttpURLConnection Classes

» REFLECTION

- Understanding the Need Of Reflection
- Getting information about class's modifiers, fields, methods, constructors and super classes
- Finding out constant and method declaration belong to an interface
- Creating an instance of the class whose name is not known until runtime
- Getting and setting values of an object's field if field name is unknown until runtime
- Invoking a method on an object if the method is unknown until runtime
- Invoking Private Methods

» EXTENDED & UTILITY CONCEPTS

- Generics
- Lambda Expression
- Annotations

- Object Cloning
- Vargs
- Static-import
- Enum
- Static, Default and Private Methods of Interface
- Var Type
- Java Modules

» **COLLECTIONS FRAMEWORK**

- What is Collection?
- What is Framework?
- Collections Framework
- Core Interfaces
- Collection, List, Queue, Deque
- Set, NavigableSet, SortedSet
- Map, NavigableMap, SortedMap
- Core Classes
- ArrayList, LinkedList, PriorityQueue, ArrayDeque
- HashSet, LinkedHashSet, TreeSet,
- HashMap, IdentityHashMap, WeakHashMap, LinkedHashMap, TreeMap
- Accessing a Collection via an Iterator
- Accessing List via ListIterator
- Accessing a Collection via for each loop
- Working with User Defined Objects
- The Comparator and Comparable Interfaces
- The Legacy classes and Interfaces.
- Enumeration, Vector, Stack
- Hashtable, Properties

» **DATE & TIME API**

- java.util.Date
- java.util.Calendar
- java.sql.Date

» **JODA API**

- java.time.LocalDate
- java.time.LocalDateTime
- java.time.LocalTime

» **SYSTEM PROPERTIES & INTERNATIONALIZATION (I18N)**

- Understanding Locale
- Resource Bundle
- Usage of properties file
- Fetching text from Resource Bundle
- Displaying the text in HINDI
- Displaying date in Hindi

» **INTRODUCTION TO SQL(PROJECT BASED)**

» **DATABASE PROGRAMMING USING JDBC**

- Need Of JDBC
- JDBC Drivers
- Statement, PreparedStatement, CallableStatement
- Scrollable and Updatable ResultSet
- Batch Updates
- Transaction
- Metadata

» **JAVA EE(JAVA PLATFORM ENTERPRISE EDITION)**

- Understanding the Concept of Java EE : JEE Specification
- Java EE Architecture
 - Single Tier
 - Two Tier
 - Three Tier
 - N-Tier
- Java EE Components
- Web Components
- Distributed(Business) Components
- Java EE Containers& Servers
- Web Container& Web Server(Apache Tomcat)
- EJB Container& Application Server(Weblogic,Glassfish,Websphere)
- Java EE Services
 - JNDI Service
 - Java Transaction Service
 - JAAS
 - JMS

» **JAVA SERVLET**

- Introduction to web programming
- Role of Servlet in web programming
- Servlet Lifecycle
- Servlet with Annotations
 - @WebServlet
 - @WebInitParam
 - @WebListener
 - @WebFilter
 - @MultipartConfig
- Request Dispatching
- Parameters & Attributes and their differences
- ServletConfig and ServletContext
- File Uploading and Downloading
- Session Tracking&State Management
 - Cookie
 - Url Rewriting
 - Hidden Form Field
 - Session Object
- Events & Listeners
- Dependency Injection
- Refreshing Servlet
- Filters

» **JAVA SERVER PAGES (JSP) & JSTL**

- JSP Architecture
- JSP Elements
- JSP Objects
- Understanding JavaBeans
- Custom Tags
- Using tags of JSTL
- Expression Language

» **PROJECT CLASSES**

- Front End Coding
- FORM DESIGNING
 - HTML
 - CSS
 - JAVA SCRIPT
 - BOOTSTRAP

- Back End Coding
- DATABASE DESIGNING
- Connecting forms to database
- Writing Business Logic
- Project Hosting

» DESIGN PATTERN

- Why Design Patterns...?
- Front Controller
- Composite View
- Factory Pattern
- Singleton Pattern
- DAO Pattern

» JAVA MAIL API

- Email System and Protocols
- Sending & Receiving Mails
- Handling Attachments

» INTRODUCTION TO DISTRIBUTED PROGRAMMING

- RMI
- Web Services

» INTRODUCTION TO RESTFULL SERVICES

- @PathParam
- @Path
- @FormParam
- @QueryParam
- @DefaultValue

» OVERVIEW OF JPA FRAMEWORK

SPRING

» SPRING

- What is Spring?
- Spring modules
- Understanding dependency Injection
- Applying aspect-oriented programming

» BASIC BEAN WIRING

- Containing your Bean
- Creating bean
- Injecting into bean properties
- Auto wiring
- Controlling bean creation

» ADVANCED BEAN WIRING

- Declaring parent and Child Bean
- Applying method injection
- Injecting Non-spring Beans
- Registering Custom property editors

» ADVISING BEANS

- Introducing AOP
- Creating classic spring aspects
- Creating advice

- Defining Pointcuts and Advisors
- Using proxyFactory Bean
- Autoproxying

» HITTING THE DATABASE

- Learning spring's data Access Philosophy
- Configuring a data source
- Using JDBC with Spring
- Working with JDBC Templates
- Using Spring's DAO Support Classes for JDBC
- Integrating Hibernate with Spring
- Caching

» INTRODUCTION TO MVC

- Define MVC
- Hibernate Injection
- Spring Annotation
- Spring Controller

» MAVEN DEPLOYMENT

- Maven Configuration
- Converting Maven to Eclipse
- Various Maven Command

» SPRING REST API

- Creating Rest
- Consuming Rest
- Calling on Client

» BUILDING CONTRACT-FIRST WEB SERVICES IN SPRING

- Introducing Spring-WS
- Defining Contract (First!)
- Handling messages with service endpoints
- Wiring it all together
- Consuming Spring-WS Web services

» SPRING OBJECT/XML MAPPER

» SPRING BOOT

- Project Creation
- Boot Elements
- Boot Services
- Boot Annotation

HIBERNATE

» INTRODUCTION TO ORM

- Need of ORM
- Problems using JDBC Directly
- ORM Implementation

» INTRODUCTION TO HIBERNATE

- Hibernate Architecture
- Hibernate configuration
- Hibernate's Support for Other Technologies
- Installing Hibernate
- A "Hello world" stand alone application
- A Servlet-Based Hibernate application

» CREATING PERSISTING CLASSES

- Mapping a basic Java Class
- Mapping a Class with Binary Data
- Mapping a Serializable Class
- Mapping a class with Data/ calendar attributes
- Mapping a Read-only class
- Mapping a class using Versioning /Timestamps

» MAPPING INHERITENCE WITH JAVA CLASSES

- Table-Per –class Hierarchy Mapping
- Table-Per –subclass Hierarchy Mapping
- Table-Per –concrete-subclass Hierarchy Mapping
- Persistence interfaces

» WORKING WITH COLLECTIONS

- Associations
- Lazy initialization
- Mapping Maps/Sorted Maps
- Mapping Sets/Sorted Sets
- Mapping lists
- Mapping Arrays
- Mapping a Bidirectional Association

» SCALAR QUERIES AND HIBERNATE QUERY LANGUAGE

- Queries
- Named Queries
- SQL Queries
- Hibernate Queries language

» HIBERNATE TRANSACTIONS AND LOCKING

- Configuration
- Database support
- Using Transactions
- The Transactions API
- Transaction Example Using Oracle
- Locking

» HIBERNATE CACHING

- How caching improves performance
- First level lache
- Second level cache

WEB TECHNOLOGIES

- HTML, CSS, Angular

MANUAL TESTING

» Introduction

- What is Quality?
- What is testing?
- Why Testing?

» **Software Development Life Cycle**

- Initial (or) Requirements phase
- Analysis, Design, Coding & Testing phase
- Delivery and Maintenance phase

» **Testing Methodology**

- Black Box, White Box & Gray Box Testing

» **Levels of Testing**

- Unit Level, Module Level, Integration Level,
- System Level & User Acceptance Level Testing

» **Environments**

- One Tier, Two Tier, Three Tier Architecture
- N- Tier Architecture

» **Software Development Models**

- Waterfall, Prototype, Evolutionary, Spiral, Fish & V-Model

» **Types of Testing**

- Build Verification, Regression, Re-Testing,
- Alpha, Beta, Static, Dynamic, Installation,
- Compatibility, Usability, End-To-End, Exploratory,
- Security, Port, Reliability, Mutation, Adhoc Testing

» **Software Testing Life Cycle**

- Test Planning, Contents of Test Plan

» **Test Development**

- Use Case Reviews, Types of Test Cases
- Formats of Testing Documents
- Testing Process, Test case design
- Test design techniques, BVA, ECP

» **Test Execution**

- Execution Process
- End-to- End Scenarios execution
- Result Analysis

» **Bug Tracking and Reporting**

- Types of Bugs, Identifying the Bugs
- Isolation the Bugs, Bug Life Cycle
- Reporting the Bugs, Classical Bug Reporting
- Bug Tracking Tool

» **Test Closure Activity**

- Test Execution stop criteria
- Test summary reports

» **Real Time Process Awareness with Terminology**

- Quality Assurance, Quality Control
- NCR, Inspection, Audit, CAPA
- Software Configuration Management
- Build Release process, SRN, SDN, Slippage
- S/W Delivery process, Reviews, Peer-Review
- Traceability Matrix, Metrics
- Test Bed, Escalation Process
- Base Lining the Documents

- Publishing the documents
- Common Repository Management
- Patch, PPM, PPR, MRM, Defective Product
- Change Request, Impact Analysis, Walk Through
- Code Walk Through, Code Optimization
- Work Around, Defect Age, Latent Defect
- Defect Product, Test Suite, Prototype
- Review Report, Template

» Test Metrics

- Importance of test metrics
- Different types of metrics

» Ways of Testing

- Manual Testing, Automation Testing
- Drawbacks of Manual Testing
- Drawbacks of Automation Testing

AUTOMATION TESTING-SELENIUM

» Introduction:

- What is Automation Testing
- Use of automation Testing
- Tools for Automation Testing
- Why automation is important for you career?
- What is Selenium.
- Advantage of Selenium.
- History of Selenium.
- Component of Selenium.
- Architecture of Remote Control.
- Architecture of WebDriver.
- Architecture of GRID.
- Differences between API's Vs GUI's.

» Selenium IDE:

- Selenium IDE Introduction
- Record and Playback with Firefox and Chrome
- Debugging in Selenium IDE Script
- Introduction Katalone Studio
- Record and Playback through Katalone Studio

» Set up Eclipse:

- Download and install java
- Download and start Eclipse
- Download and configure WebDriver java client
- Set up a project
- Create packages
- Create a First Java test case
- Import WebDriver Source file

» WebDriver:

- WebDriver Interface
- WebElement Interface
- Launching Firefox browser

» **Browser & Navigation Commands:**

- How to Open a URL
- Verify Page title
- Strategy to get the Page Source
- Difference between Close & Quit
- Ways to Navigate Back & Forward
- How to Refresh Page
- Another way of Navigating to specific Page

» **Locators:**

- What are locators.
- HTML Basics.
- HTML language tags and attributes.
- ID, Name, Xpath, CSS etc.
- Difference between Absolute & Complete Xpath.
- Finding your first element.
- WebElement Commands.

» **Element Identification:**

- Element Inspector in Mozilla, Chrome and IE
- Element locator tool for Firefox Locator
- Firebug&Fire Path Add-Ons in Mozilla
- Various HTML locator strategies
- XPath Helper Plug-in for Chrome
- Selection of Effective Xpath
- Handling Dynamic objects/ids on the page

» **Tables, Checkboxes & Radio buttons:**

- Identify table rows and columns
- Extracting values from a cell
- Dynamically Identify Tables Data
- Select class in Selenium
- Drop Down Handle.
- Select multiple values from the list
- Select & Deselect operations by Index, Value & Visible Text.

» **Selenium Waits, Alert & Switch Windows:**

- Implicit and Explicit waits
- How to use Expected Conditions with Waits
- PageLoadTimeout & SetScriptTimeout property
- Simple use of Thread Sleep
- Concept of Fluent Wait in Selenium
- WebDriverWait and its uses
- Different WaitUntil Conditions
- Ways to handle Simple, Confirmation & Prompt Alert
- Concepts of Set Interface in Java
- Difference between Window Handle & Handles
- Switching & Closing Windows, Tabs & Popup's
- Concept of window ID
- Extracting window IDs with Selenium Object reference
- JavaScriptExecutor Interface
- Captured Screenshot
- Cookies Handles

» **Action Class:**

- What is Action Class & What can we do with Action Class
- Mouse Hover & Mouse Movement with Action
- Finding Coordinates of a Web Object
- Drag and Drop Action

» TestNG Framework:

- What is TestNG
- Benefits and Features of TestNG
- How to download TestNG
- Annotations in TestNG
- How to run Test Suite in TestNG
- Groups in TestNG
- Depend on in TestNG
- Test Case sequencing in TestNG
- TestNG Asserts
- TestNG Parameters
- Multi Browser testing in TestNG
- Parallel testing in TestNG
- Extent Report API

» Cucumber :

- Introduction Cucumber
- Using Cucumber
- Gherkins Language
- Creation of feature files and Step Definition
- Introduction TDD and BDD
- TDD Vs BDD
- Version Control (GitHub Introduction)

» Log4j Logging:

- Log4j Introduction
- Download Log4J
- Add Log4j Jar
- Test Case with Log4j
- Log4j Log Manager
- Log4j Appenders
- Log4j Loggers

» Database Connections:

- Database connection
- Database Testing in Selenium using MySQL Server

» Automation Framework:

- What is Automation Framework
- Features of Automation Framework
- Benefits of using Automation Framework
- Different types of Automation Framework
- What is Data Driven Framework
- What is Modular Driven Framework
- What is Keyword Driven Framework
- Apache POI API
- POI Setup and Configuration
- Read and Write Excel file with Apache POI

» Maven:

- Maven Introduction
- Install Maven in Eclipse IDE
- Install Maven on Windows
- Install Maven on Mac
- How to Create a New Maven Project
- How to Create a New Maven Project in Eclipse
- Configure Selenium Continuous Integration with Maven

» **Jenkins:**

- Jenkins Introduction.
- Selenium Integration with Jenkins.

REASONING & QUANTITATIVE APTITUDE

» **Quantitative Ability Test**

- Basic Mathematics
- Divisibility
- HCF and LCM
- Numbers, decimal fractions and power

» **Applied Mathematics**

- Profit and Loss
- Simple and Compound Interest
- Time, Speed and Distance
- Inverse

» **Engineering Mathematics**

- Logarithms
- Permutation and Combinations
- Probability

» **Logical Ability Test**

- Deductive Reasoning
- Coding deductive logic
- Data Sufficiency, Directional Sense, Logical Word Sequence
- Objective Reasoning
- Selection decision tables
- Puzzles

» **Inductive reasoning**

- Coding pattern and Number series pattern recognition
- Analogy and Classification pattern recognition

» **Abductive Reasoning**

- Logical word sequence
- Data sufficiency

SOFT SKILLS

- Group discussion
- Resume Writing
- HR Interview

PROJECTS

- Two Development Projects
- One Testing Projects

ENTIRE COURSE DURATION : 100 DAYS

CLASSES CONDUCTED : MONDAY - FRIDAY

TESTS & ASSESSMENT : SATURDAY

MOCK INTERVIEW : AS PER SCHEDULE

FEE : 30000+TAXES

NOTES

Partners :



Java



development | consultancy | training

E-mail: info@ducatindia.com
Visit us: www.ducatindia.com
www.facebook.com/ducateducation

NOIDA

A-43 & A-52, Sector-16,
Noida - 201301, (U.P.) INDIA
☎ 70-70-90-50-90
☎/☎ +91 99-9999-3213

GURGAON

1808/2, 2nd floor old DLF,
Near Honda Showroom,
Sec.-14, Gurgaon (Haryana)
☎ 70-70-90-50-90

GHAZIABAD

1, Anand Industrial Estate,
Near ITS College, Mohan Nagar,
Ghaziabad (U.P.)
☎ 70-70-90-50-90

PITAMPURA (DELHI)

Plot No. 366, 2nd Floor,
Kohat Enclave, Pitampura,
(Near- Kohat Metro Station)
Above Allahabad Bank,
New Delhi- 110034.
☎ 70-70-90-50-90

SOUTH EXTENSION (DELHI)

D-27, South Extension-1
New Delhi-110049
☎ 70-70-90-50-90
☎ +91 98-1161-2707