ðŸ'¥ Download Ashok IT Mobile App ðŸ'¥

ðŸ"¥ Android App Link (Ashok IT) : http://bit.ly/3Z2vXcO
ðŸ"¥ IOS App (MyInstitute) Link : https://apple.co/3IcEao3
ðŸ'‰ ORG Code For IOS : CNFXJ

=====================
Enrollment Process
=====================

Java Realtime Project (23-JRTP) - Payment Details

Plan-1 : 8,000 INR (Live Classes + Class notes)

Plan-2 : 12,000 INR (Live Classes + Class Notes + Backup Videos - 1 Year Access)

Bank Name : AXIS Bank
Holder Name : Bollepalli Ashok Kumar
Account Number : 913010027824643
IFSC Code : UTIB0000027
Branch : Taranaka
Hyderabad

Google Pay / Phone Pay : 9701787666 (Ashok)

Note: Complete your payment and send below details to ashokitpayments@gmail.com

1) Name
2) Contact Number
3) Whatsapp Number
4) Course Code (23-JRTP)
5) Class Mode (Online / Offline)

==================================
Java Realtime Project (23-JRTP)
==================================

Pre-Requisites :

1) Core Java
2) Adv Java (JDBC + Servlets)
3) Oracle (SQL)
4) Hibernate Basics
5) Spring Boot & Microservices (Paralelly can do this)

Q) Who is elgible to attend this course ?

1) Freshers

2) Experienced People also can attend

===============
Course content
===============

Part-1) Software Industry Introduction

1) Types of companies
2) Types of Projects
3) Technical Domain
4) Functional Domain
5) Types of Teams
6) Bridge Calls
7) Minutes of Meeting
8) Role Structures

Part-2) Realtime Tools (20+ tools)

1) Agile & JIRA
2) Maven & Gradle
3) Git Hub & BitBucket
4) Logging (Log4J / Log4J2 / LogBack / LogStash)
5) Log Monitoring (Putty / WinScp / Splunk)
6) Unit Testing (Junit + Mockito)
7) Code Coverage (Jacocco)
8) Code Review (SonarQube)
9) JMETER (Performance Testing)
10) JENKINS (CI CD)
11) Docker - Containerization
12) Kubernetes - Orchestration Platform
13) Apache Kafka - message broker
14) Redis Cache - to reduce db calls
15) POSTMAN - To rest REST APIs
16) Swagger - To generate REST API Documentation

Part-3 : Mini Projects (3 to 4)

- Requirement Analysis
- Development
- Testing
- Deployment

Part-4 : Major Project (Microservices Architecture)

Part-5 : AWS Cloud with Linux Commands

Part-6 : Angular Concepts

Part-7 : Interview Preparation

1) Resume Preparation
2) How to cover gap as experience
3) Interview Questions
4) Joining Formalities
5) Exit Formalities
6) Do's & Don'ts

==

Name : 23-JRTP

Duration : 3 to 4 Months

Class Timings : 9:30 AM to 11:30 AM (IST) (Mon-Sat)

Mode : Online & Classroom

Course Fee : 8,000 Live Classes+ Notes (Plan-1)

Course Fee : 12,000 Live Classes + Notes + Backup Videos - 1 year access)

Start Date: 15-Mar-2023

Note: Module-1 will start from 16-March-2023

============
Requirement
============

System with 8 GB RAM

======================================
Module-1 : Software Industry Details
======================================

=> We can see 3 types of Companies in IT

1) Product Based Companies : They will develop softwares & sell in market

Ex: Apple, Samsung, Dell, Oracle, Amazon, Microsoft etc...

2) Service Based Companies : Develop projects for clients

Ex: TCS, Infosys, Wipro, TechM, HCL, Deloitte, Capg, CTS etc...

3) Outsourcing Companies : Will supply employees to other companies

==================
Interview Process
==================

1) Product Based Companies

=> Data Structures
=> Algorithms
=> System Design
=> Problem Solving
=> Design Patterns
=> Coding Test

-> Interview : 5 to 6 rounds

-> Package : Years of exp * 6 to 7 lakhs

Note: On-Site Opportunity chances are very rare.

2) Service Based Company

=> Coding Round (Arrays + Strings + Java 8)
=> Core Java questions
=> Hibernate ORM
=> Spring Boot
=> Microservices
=> Tools (git, maven, docker, jenkins etc)
=> Frontend Questions
=> Cloud

Interview : 2 to 3 rounds

-> Package : years of exp * 3 to 4 lakhs

Note: On-Site Opportunities will be available

=======================
Types of Software Jobs
======================

1) Permanent Job (More Benefits for employee)

=> Health Insurance for family members
=> PF
=> Notice Period

2) Contract Job (less benefits)

=> Health Insurance (May Available / May Not)
=> PF Optional

=> Notice Period (15 to 30 days)

==================
Types of Projects
==================

1) Scartch Development ------> 10 %

2) Maintence / Support Project --> 80 %

=> Change Request (CR)
=> New Enhancement
=> Bug Fixing

3) Migration Project ---> 10 %

===============
Types of Teams
===============

1) Offshore Team (Devlopment + Testing + Operations)

2) Onshore Team (client location - Functional / Business Analyst team)

-> Bridge Calls (Zoom / WebEx / Skype Business / Microsoft Teams)

--> For every meeting we have to prepare MoM.

===================
Role Structure
===================

Fresher ===> Associate Analyst

1 - 3 exp : Business Analyst

3 - 6 exp : Consultant / Software Engineer

6 - 9 exp : Sr. Consultant / Sr. S/w engineer

9 -12 exp : Tech lead / Team Lead

12 - 16 : Manager

16 - 20 Years : Sr. Manager / Architect

20+ : Director

==========================
Module-2 : Realtime Tools

=========================

1) GIT HUB : Version Control Software (To store project source code)

Alternate : BitBucket

=> Integrate code
=> Monitor code changes

2) Maven : Build Tool (To automate build process of the application)

Alternate : Gradle

=> Can create project
=> Download dependencies
=> Can compile
=> Can execute Junits
=> can package (jar / war)

3) JIRA : Project Management & Bug Reporting

=> Task Assginment
=> Bug Report

4) Log4J : To implement Logging

=> Store application execution details to a file/console/db
=> It helps in understanding exceptions occured in the project

5) Splunk : To monitor log messages of our application

Alternates : Putty & WinScp

6) Junit & Mocking : Unit Testing framework

=> To identify code behaviour

7) Jacocco : Code Coverage

=> Identify how many lines of code is tested in unit testing
=> Industry standard is 80%

8) SonarQube : Code Review

=> Indentify developers mistakes in code
=> Check Naming Convention
=> identity duplicate code
=> Identify security issues in code

9) Jenkins : Continuos Integration & Continuous Deployment

=> To automate deployments

10) Postman : To test REST API functionality

11) Swagger : To generate Documentation for REST Api

15) Apache Kafa : Message Broker

16) Redis Cache : To reduce no.of db calls to fetch static data

17) Docker : Containerization

18) Kubernetes (K8S) / Openshift : Orchestration

========
Maven
=======

=> Maven is a build tool developed by Apache Organization

=> Maven is developed using Java Language

=> Maven is free software

=> Maven is used to automate build process of the project

========================
What Maven Can do ?
========================

1) Create Project folder structure
2) Download Required Dependencies (Ex: hibernate, spring etc...)
3) Add dependencies to project build path
4) Compile project source code
5) Execute Unit test cases of project (Junits)
6) Package our project as jar or war file

=================
Environment Setup
=================

1) Download and Install Java software

2) Set JAVA_HOME

JAVA_HOME = C:\Program Files\Java\jdk-11.0.17

3) Set Path for Java

Path = C:\Program Files\Java\jdk-11.0.17\bin

4) Download Maven software from Apache website as zip file

Link : https://dlcdn.apache.org/maven/maven-3/3.9.0/binaries/apache-maven-3.9.0-bin.zip

5) Extract that maven zip file

6) Set MAVEN_Home in environment variables

MAVEN_HOME = C:\apache-maven-3.9.0

7) Set Path for maven

8) Verify maven installation using cmd

```
$ mvn -v
```

Note: If you are not able to create Maven project in IDE then close your IDE and delete .m2 folder and open your IDE.

==================
Maven Terminology
==================

Archetype : It represents type of project we want to create

quickstart : standalone application
webapp : web application

groupId : It represents organization name

ex: com.tcs
com.ibm
in.ashokit
org.springframework

artifactId : It represents project name

ex: flipkart_app
amazon_app

version : It represents project version

SNAPSHOT : Under development
RELEASE : Develoment completed (production)

Packaging : How you want to package your app (jar / war)

quickstart : standalone application ==> jar
webapp : web application ==> war

Maven Goals : Goals are used to perform maven build process

clean : delete target folder
compile : compile source code
test : execute junits
package: create jar/war file
install : store artifact into artifact repo

Maven Plugins : Every Maven goal is having maven plugin to perform operation.

==
Creating Maven Standalone Project in CMD
===

```
mvn archetype:generate -DgroupId=in.ashokit -DartifactId=my-app -DarchetypeArtifactId=maven-archetype-quickstart -DarchetypeVersion=1.4 -DinteractiveMode=false
```

==
Creating Maven Web Project in CMD
===

```
mvn archetype:generate -DgroupId=in.ashokit -DartifactId=web-app -DarchetypeArtifactId=maven-archetype-webapp -DarchetypeVersion=1.4 -DinteractiveMode=false
```

Note: For every maven project pom.xml file will be created.

POM : Project Object Model

=> Maven dependencies we will configure in pom.xml file

Note: We need to execute maven goals from project directory

=> We can find maven dependencies in www.mvnrepository.com

```
<dependencies>
<dependency>
<groupId>org.springframework</groupId>
<artifactId>spring-context</artifactId>
<version>5.3.25</version>
</dependency>
</dependencies>
```

=> When we add above dependency maven downloading spring-core, spring-bean and spring-aop also. This is called as Transitive Dependency Management.

====================
Maven Repositories
====================

=> Maven repository means a place where all jars will be stored.

=> In Maven we have 3 types of Repositories

1) Central repository
2) Remote Repository
3) Local repository