Course name : Java Real-Time Project (Fullstack Development)

(Spring Boot, Microservices & Angular - 11)

Start Date : 22-Apr-2021

Class Timings: 10:00 AM IST - 11:30 AM IST (1.5 hour) (Mon-Sat)

Duration : 3 months (10 days +/-)

Trainer : Mr. Ashok (8+ yrs experience, Working Professional)

(5+ Years of experience in Training)

YouTube Channel : www.youtube.com/c/AshokIT

Facebook Group : www.facebook.com/groups/ashokitschool

----------------------------------------------------------------------------------

Pre-Requisites:

---------------

Core Java

Adv Java (JDBC, Servlets & JSP)

SQL

ORM Framework (Good To have)

Spring Boot & Microservices (You can do this course parallely - @7:30 AM IST)

Angular - 11 with Typescript (You can do parallely - @11:30 AM IST - FREE FOR ALL)

Course Content:

---------------

Part-1 : Software Industry Terminology

----------------------------------------

1) How many type of software companies

2) Interview Process In Software Companies

3) What is Software Project

4) How many types of software projects & work culture

5) How many types of teams & Roles of teams

6) Role Chart In Software Company

7) Bridge Calls & VC

8) Conclusion

Part-2 : Java Real-Time Tools (30+ Realtime Tools)

----------------------------------------------------

1) Maven

2) Gradle

3) SVN (Videos available in our Youtube)

4) GIT HUB

5) BitBucket

6) Source Tree

7) Log4J, LogBack & SLF4J

8) Log Monitoring Tools (Putty, WinScp & Splunk)

9) JUnit (For Unit Testing)

10) Mocking (Easy Mock, PowerMock, Mockito)

12) Code Coverage (Jacocco)

13) Sonar Qube (Code Review)

14) Agile with JIRA (Project Management)

15) JMETER (Performance Testing)

16) Swagger (Generating Documentation for REST APIs)

17) SOAP UI & POSTMAN (API Testing Tools)

18) JENKINS (CI & CD Software)

19) DOCKER (Containerization Tool)

20) Reports Using Apache POI & IText API

21) Java Mail API

22) Apache Kafka

23) Redis Cache

24) AWS Deployment

Part-3 : Three Mini Projects (Spring Boot & Microservices with Angular)

-----------------------------------------------------------------------

1) Project-1

2) Project-2

3) Project-3

Part-4 : Major Project (This keep project you can keep in Resume)

-----------------------------------------------------------------------------

- Health Insurance

- 7 Modules

- Project Architecture

- Project Flow

- Batch Processing

- Multi Threading

- Exception Handling

- Reports

Part-5 : Interview Guide

------------------------

- Resume Preparation

- Profile Creation In Job Portals

- How to apply for Jobs

- How to cover gap

- Joining Formalities

- First Day in Company

- Last Day in company

- Full and Final Settlement

- Do's and Don't in IT company

JAVA REALTIME PROJECT (FULLSTACK DEVELOPMENT) - ONLINE TRAINING

-----------------------------------------------------------------

Start date : 22-Apr-2021

------------------------------------------------------------------------

Session-1 (22-Apr) : Pre-requisites & Course-content

(https://www.youtube.com/watch?v=exSnWlLdT7E)

Session-2 (23-Apr) : Doubts Clarification for students

(https://www.youtube.com/watch?v=f\_kGbvI3CFg)

24-Apr & 25-Apr : No classes

-----------------------------------------------------------------------

Session-3 (26-Apr-2021) : Part-1 (Software Industry Details

-----------------------------------------------------------------------

-> Software companies can be categorized into 3 types

1) Product Based Companies

2) Service Based Companies

3) Outsourcing Companies

Product Based Companies

------------------------

-> The companies which will develop product/project and selling to the people in the market is called as Product Based Companies.

-> For Product Based Companies clients will not be available they will have only Customers.

Ex : Microsoft, HP, IBM, Google etc...

-> In Product Based Companies interviews, they will concentrate mainly on below concepts

1) Data Structures

2) Algorithms

3) Design Patterns

4) System Design

5) Problem Solving

6) Time Complexity and Space Complexity

Note: In Product Based Companies they will conduct Coding Test using Hacker Ranker or Hacker Earth.

-> Product Based Companies provide high packages

No.of.years of exp \* 6 Lakhs

3 \* 6 = 18 Lakhs

Service Based Companies

-----------------------

-> The companies which are providing services to others are called as Service Based Companies.

-> Service Based Companies will deveop the projects based on client given requirements.

Ex : TCS, Wipro, Infy, Capgemini, Accenture, Cognizent, Deloitte etc..

-> In Service Based Companies interviews, they will ask questions related to latest technologies

1) Spring Boot

2) REST Apis

3) Microservices

4) Fullstack development

-> Below package structure will be there in Service based companies

Years of exp \* 3 lakhs

-> In service based companies onsite opportunity will be there.

Outsourcing Companies

------------------------

-> Outsourcing companies will provide employees to other companies on contract basis.

Ex: Magna info Tech, SourceOne mgmt etc..

Course Details

------------------

15-JRTP Course Fee : 6000 INR (Angular-11 FREE)

Daily class notes and class recording will be shared

class notes life time access

class recording will be avaialble for only 2 days

Spring Boot & Microservices : 5000 INR

Yesterday's session : Types of companies in IT industry

-------------------------------------------------------------------------

-> We can see 3 types of companies in IT industry

a) Product Based Companies

b) Service Based Companies

c) Outsourcing Companies

------------------------------------------------------------------------

Today's session : How many types of projects in companies

-----------------------------------------------------------------------

-> In IT companies we can see 3 types of projects

1) Scratch development projects

2) Maintenence Projects

3) Migration Projects

-> If we develop a brand new project from the beginning then it is called as Scratch Development Project.

-> The projects which are already running in the market will be under maintenence.

-> In the maintenence projects we will have below types of works

1) Change Request

2) Enhancements

3) Bug Fixing

-> If we are changing our project from one technology to another technolor then it is called as Migration Project.

Part-2 of 15-JRTP : Java Realtime Tools

-------------------------------------------------------------------------

-> In part-2 we will learn some realtime tools which we will use in realtime project.

-> We will learn how to download and install those tools

-> We will learn how to work with those tools in project

-> We will learn what are common problems we will face while working with those tools and how to rectify them.

---------------------------------------------------------------------------------

-> After class is completed everybody has to practise the tools concept

-> Sometimes in class i will ask students randomely to perform operations with tools (By giving co-host)

-> Weekly assignments also will be there

Note: Now we will learn few tools then will start Mini Projects development. Once first mini project is completed we will learn remaining tools.

---------------------------------------------------------------------------------

Maven : Build tool

Git Hub & Git : Source Code Repository Tools

BitBucket : Source Code Repository Tool

Source Tree : Repository client software

Log4J : Logging

Putty, WinScp, Splunk : Log monitoring tools

Jenkins : CI & CD tool

Agile JIRA : Project management tool

----------------------------------------------------------------------------------

Mini Project Development

----------------------------------------------------------------------------------

Junit

Mocking (Easy Mock & PowerMock)

Jacoco (Code Coverage)

SonarQube ----------Code review

Swagger

PostMan

JMETER ---> Performance Testing

--------------------------------------------------------------------------------

2 Mini Project

--------------------------------------------------------------------------------

Docker

Apache Kafka

Redis Cache

Apache poi api

itext api

---------------------------------------------------------------------------------

3 mini project

--------------------------------------------------------------------------------

Major Project

--------------------------------------------------------------------------------

Interview Guide

----------------------------------------------------------------------------------

Note: Please check your email before joining 15-JRTP class daily

Today we will share new zoom link with all previous videos links.

Yesterday's session : Introduction for Part-2 in 15-JRTP

------------------------------------------------------------------------

Build Tools

-----------

-> Build tools are used to automate application build process

Ex : Ant, Maven and Gradle etc...

Note: Ant is outdated in market people are using Maven or Gradle as build tool for java projects development.

-> Apache Organization provided Maven software to simplify application build process easy.

Maven Main objectivies

------------------------

-> Make build process easy

-> Maintain uniform build process

-> Provide Quality information about project

-> Encouraging Better development practises

-> Download maven software using below url

https://maven.apache.org/download.cgi

(apache-maven-3.8.1-bin.zip)

-> Extract maven zip file then add below 2 things in Environment variables

1) MAVEN\_HOME

2) MAVEN PATH

Yesterday's session : Maven Introduction

-----------------------------------------

-> Maven software provided by apache org

-> Maven is a free & open-source software

-> Maven is used to simplify java projects build process

-> Maven follows uniform build process for all the applications

-> Maven provides quality information related to project

-> Maven is encouraging to follow better development practises

-> Maven setup also got completed

------------------------------------------------------------------------

-> We can create maven project using cmd and using IDE also

-> 'ArcheType' Selection is required to create maven project. It represents type of the project.

quick-start : Java Standalone app

web-app : Java Web application

Yesterday's session : Maven Project Folder Structure

------------------------------------------------------------------------

-> Maven project contains pom.xml file which acts as maven configuration file

-> All the dependencies which are required for our application we will add in pom.xml file only

Note: We can identify maven dependencies in mvnrepository.com website

---------------------------------Adding maven dependency----------------

<dependencies>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-core</artifactId>

<version>5.3.6</version>

</dependency>

</dependencies>

------------------------------------------------------------------------

-> We can add multiple dependencies under <dependencies/> tag in pom.xml file

-------------------------------------------------------------------------

-> To perform application build process maven provided several goals for us

1) clean

2) compile

3) test

4) package

5) install etc.

-> 'clean' goal is used to delete target folder

-> 'compile' goal is used to compile java classes and generate .class files in target folder

-> 'test' goal is used to execute junit classes available in application for unit testing

-> 'package' goal is used to generate 'jar' or 'war' file for our application

-> 'install' goal is used to represent our project as dependency for other projects

10-may

Last session : Maven Goals

------------------------------------------------------------------------

-> Maven provided several goals to perform application build process

-> Every maven goal having its plugin to perform the actual operation

1) clean

2) compile

3) test

4) package

5) install

-------------------------------------------------------------------------

Today's session : Maven install goal

--------------------------------------------------------

11-may

Yesterday's session : Maven install goal

------------------------------------------------------------------------

-> To create our own dependency we will use maven install goal

-> Once we create dependency using maven install goal, then we can use that project as a dependency in other projects by configuring dependency details in pom.xml file.

-------------------------------------------------------------------------

clean, compile, test, package and install

----------------------------------------------------------------------------------

Today's session : Maven Repositories

---------------------------------------------------------------------------------

-> Maven supports 3 types of repositories

1) Central Repository

2) Local Repository

3) Remote Repository

-> Central Repository will be maintained by apache org (all jars will be available)

-> When we configure maven it will create local repository in our system

(Local Repo Path : C:\Users\<username>\.m2 )

-> Every Software Company will maintain their own repository for jars that repository is called Remote Repository.

-> When we add dependency in project pom.xml file then maven will search for that dependency in local repo first. If dependency available in local it will add to project build path.

-> If dependency not availbale in local repo then it will download that dependency from Central Repository.

Note: If we configure Remote Repository in our IDE, if dependency not available in local then it will download from Remote Repository.

Note: In company we will connect to Remote REpository only

-> To configure Remote Repository in IDE we will follow below steps

IDE -> Window -> Preferences -> Maven -> User Settings -> Add settings.xml file

Note: Once we join in project, TL or Manager will provide settings.xml file for us

14-may

12-May-2021 & 13-May-2021 : No classes

-------------------------------------------------------------------------Last session : Maven Repositories (Central, Local and Remote)

-------------------------------------------------------------------------

-> Central Repo will be maintained by Apache Org

-> In our machine Maven Local Repo will be available

-> Every company will maintain their own Remote Repository

-------------------------------------------------------------------------

Today's session : Pom.xml tags

-------------------------------------------------------------------------

-> pom stands for project object model

<project>

<groupId>in.ashokit</groupId>

<artifactId>01-Maven-App</artifactId>

<version>0.0.1-SNAPSHOT</version>

<packaging>jar</packaging>

<properties>

<spring.version>5.3.7</spring.version>

</propertites>

<dependencies>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>${spring.version}</version>

<scope>runtime</scope>

<dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-webmvc</artifactId>

<version>${spring.version}</version>

<scope>runtime</scope>

<dependency>

<dependencies>

<build>

<finalName>First-App</finalName>

<plugins>

<plugin>

.....

</plugin>

</plugins>

</build>

</project>

-------------------------------------------------------------------------

-> Maven dependency 'scope' decides when to include that dependency in class path.

-> There are 6 scope elements available in maven

compile --------- This is by default

provided

runtime

test

system

import

-------------------------------------------------------------------------

Q) How to exclude maven dependency in pom.xml?

-------------------------------------------------------------------------

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>${spring.version}</version>

<exclusions>

<exclusion>

<groupId>org.springframework</groupId>

<artifactId>spring-aop</artifactId>

</exclusion>

</exclusions>

</dependency>

-----------------------------------------------------------------------------------------------

<project xmlns="http://maven.apache.org/POM/4.0.0"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>in.ashokit</groupId>

<artifactId>03-Maven-App</artifactId>

<version>0.0.1-SNAPSHOT</version>

<packaging>jar</packaging>

<properties>

<spring.version>5.3.6</spring.version>

</properties>

<dependencies>

<dependency>

<groupId>in.ashokit</groupId>

<artifactId>02-Pwd-Utils</artifactId>

<version>1.0</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>${spring.version}</version>

<exclusions>

<exclusion>

<groupId>org.springframework</groupId>

<artifactId>spring-aop</artifactId>

</exclusion>

</exclusions>

</dependency>

</dependencies>

<build>

<finalName>Maven-App</finalName>

</build>

</project>

-----------------------------------------------------------------------------------------------

1) What is Maven ?

2) What is Archetype selection in Maven?

3) What is maven-archetype-quickstart ?

4) What is the difference between src/main/java and src/test/java folders?

5) What is the purpose of src/main/resources folder?

6) What is pom.xml file ?

7) What are maven goals available?

8) What is the difference between clean and compile?

9) What is the purpose of maven test goal?

10) What is the purpose of maven package goal?

11) What is the use of install goal?

12) What is transitive dependency in maven?

13) How to exclude dependency in maven?

14) What is scope in maven?

15) What repositories available in maven?

16) What is settings.xml in maven?

17) What is <finalName> tag in pom.xml ?

18) What is Maven plugin?

19) What is force update in maven?

20) What is .m2 folder?

---------------------------------------------------------------------------------------------

Today's assignment : SDLC life cycle steps & Waterfall Methodology

---------------------------------------------------------------------------------------------

15-may

Last session : Maven Tool

-------------------------------------------------------------------------

Today' session : Agile Methodology

-------------------------------------------------------------------------

Q) What is SDLC ?

Q) What is waterfall methodology ?

-------------------------------------------------------------------------

-> SDLC stands for Software development life cycle.

-> It includes several phases in application development process

1) Requirements Gathering

2) Requirements Analysis

3) Design

4) Development

5) Testing

6) Deployment

7) Maintenence

-> Waterfall methodology follows linear process in SDLC.

-> In Waterfall methodology everything happens sequentially.

-> In Waterfall methodology requirements & Project budget is fixed

-> Waterfall methodology is suitable for small scale applications

---------------------------------------------------------------------------------------------

-> Now a days in industry Agile methology is using

---------------------------------------------------------------------------------------------

-> AGILE methodology is a practice that promotes continuous iteration of development and testing throughout the software development lifecycle of the project.

-> In the Agile model, both development and testing activities are concurrent, unlike the Waterfall model.

-> Agile methodology is best sutiable for large scale applications.

-> In Agile methodology requirements & budget is not fixed.

-> All the given requirements will be divided into multiple releases in Agile methodology.

----------------------------------------------------------------------------------------------

Agile Terminology

---------------------------------------------------------------------------------------------

Product Owner

Scrum Master

Tech Lead

Agile team members

-> Product Owner is responsible for client deliverables

-> Scrum Master is responsible to manage agile team

-> Tech Lead is responsible to resolve any technical problems facing by agile team members

-> Agile team contains both developers & testers (they will work paralelly)

1) Story

2) Story Points

3) Backlog Grooming

4) Backlog

5) Sprint Planning

6) Sprint

7) Scrum

8) Mid Iteration Review

9) Release

10) Retrospective

-> Story means a task in the project

ex : logging in project, exception handling, unit testing, code review etc...

-> Every story will have story points. Story points decides duration to complete the task

3 points --- 8 hours duration (1 day)

5 points --- 16 hours duration (2 days)

8 points ----24 hours duration (3 days)

Note: In the team anybody can create story and assign story points

-> Backlog grooming is the meeting conducted by scrum master to identify pending tasks in the project. All the team members will join in the backlog grooming meeting will identify pending tasks in the project.

-> For all the pending tasks stories will be created in JIRA that is called as Backlog bucket.

Note: The stories which are pending are called as Backlog Stories

-> Scrum Master will conduct a meeting with Agile team to identify the stories to complete from backlog that meeting is called as Sprint Planning. All the team members will join in Sprint Planning session.

-> The stories which are identified in Sprint Planning are called as one Sprint. Sprint will have start date and end date. Sprint duration will be 2 weeks (10 working days).

-> Once sprint started everyday scrum master will conduct scrum call for 15-20 mins of time.

In scrum call every team member will provide work status to scrum master (like which story we are working on and what is the status and when it will complete).

-> Mid Iteration review meeting will be conducted by Scrum Master to evoluate sprint stories once half of sprint duration got completed.

-> Retrospective meeting will be conducted by Scrum Master to check sprint work.

1) Lessons learnt in previous sprint

2) Improvement areas

3) Achievements

-> For every sprint we don't release project to client. After 3 to 4 sprints we will do one release to client.

16-may

Last session : Agile methodology & Agile Terminology

-------------------------------------------------------------------------

-> Agile is an iterative approach to develop software applications.

-> In Agile methodology developers & testers will work paralelly to complete the project.

-> In Agile methodology requirements are not fixed.

-------------------------------------------------------------------------Agile Terminology

-------------------------------------------------------------------------

-> Product Owner

-> Scrum Master

-> Tech Lead

-> Agile Team

-> Story

-> Story Points

-> Backlog Grooming

-> Backlog

-> Sprint Planning

-> Sprint

-> Scrum

-> Mid-Iteration Review

-> Retrospective

-> Release

-----------------------------------------------------------------------------------

-> Agile project work will be managed by using JIRA software

-> JIRA is an Atlasian company product.

-> JIRA is used to manage project work and jira is used for bug reporting also.

-> Software companies will purchase JIRA product from Atlasian company to manage projects work.

Note: We can use trail version of jira to understand jira functionality

17-may

Last session : Agile methodology & JIRA

-------------------------------------------------------------------------

Today's session : Version Control Softwares

-------------------------------------------------------------------------

-> In company multiple developers will be available to develop the project.

-> All the developers will not work from same location.

-> If developers are woring from multiple locations we will face below 2 problems

1) Code Integration

2) Monitoring Code Changes

-> To avoid above 2 problems we will use Version Control Software For Project.

-> Version Control Software will provide solution for code integration and providing monitored access for code.

-> There are several version control softwares are available in market. They are

1) Clear Case

2) CVS

3) SVN

4) Git Hub

5) BitBucket etc..

Working with GIT Hub

-------------------

-> Create account in Git Hub (Its free of cost)

-> Install Git Client software

18-may

Last session : Purpose source code repository / Version Control Software

-----------------------------------------------------------------------

-> In one project mulitple developers will be available

-> Developers will be working from different locations.

-> If developers are working from multiple location we will face below problems

1) Code Integration

2) Monitoring code changes

-> To resolve the above 2 problems we will use Version Control Softwares.

1) Clear Case

2) CVS

3) SVN

4) GIT Hub

5) Bit Bucket etc..

-----------------------------------------------------------------------

-> We have created Account in git hub and we have installed git client software.

Note: When we create account in git hub it provides some space for free of cost to host our applications.

-> Git Hub is a cloud platform which is used to store project code and for hosting applications.

-> Git is the version control software.

-> Git Hub cloud platform using Git as a version control software.

-------------------------------------------------------------------------

Note: Once we join in a project in IT company, they will provide git account credentials for us (We no need to create).

-------------------------------------------------------------------------

-> In Git account we will create repository to store project source code.

Note: For one project we will create one repository

Note: In IT company, GIT admin will create repository for the project.

-> In Git we can create 2 types of repositories.

1) Public Repository

2) Private Repository

-> Public Repository means anybody can see and we will choose who can commit

-> Private repository means we will choose who can see and commit to our repository.

-------------------------------------------------------------------------

-> Open Git bash to execute git commands

Introduce ourself to git using config command

----------------------------------------------

git config --global user.name "Ashok"

git config --global user.email "ashokitschool@gmail.com"

19-may

Last session : Git Repo Creation & Types of Repositories

-------------------------------------------------------------------------

-> Git hub is supporting for two types of repositories

1) public repo

2) private repo

-> Once Git Repository got created it will generate one unique url for that repo

https://github.com/Ashok-IT-School/15JRTP-01-App.git

-------------------------------------------------------------------------

Git Bash commands

-------------------------------------------------------------------------

git help

git help <command-name>

git config --global user.name "Ashok"

git config --global user.email "ashokitschool@gmail.com"

-------------------------------------------------------------------------

git init : To initialize our folder as git operations folder. If the .git folder is not available change file explorer options view can change display hidden files.

git add <filename> : To add one file to git staging area

Note: The files which are added to staging area are eligible for commit.

git status: To see files added and not added to staging area

git commit -m <msg> : To commit files from staging area to git local repo

git add –a : Add all the newly and modified files can added.

Git rm- -cached : all the file go under unstaging area.

$ git restore --staged B.java is unstaging file from staging area

git push : To move changes from local repo to central repository.

git pull : To take changes from git central repo to local repo

HP@DESKTOP-6E2LN7K MINGW64 /d/GIt Practice

$ git init

Initialized empty Git repository in D:/GIt Practice/.git/

HP@DESKTOP-6E2LN7K MINGW64 /d/GIt Practice (master)

$ git config --global user.name

narendra

HP@DESKTOP-6E2LN7K MINGW64 /d/GIt Practice (master)

$ git config --global user.email

dasaranarendra99@gmail.com

HP@DESKTOP-6E2LN7K MINGW64 /d/GIt Practice (master)

$ git status

On branch master

No commits yet

Untracked files:

(use "git add <file>..." to include in what will be committed)

demo1.java

nothing added to commit but untracked files present (use "git add" to track)

HP@DESKTOP-6E2LN7K MINGW64 /d/GIt Practice (master)

$ git add demo1.java

HP@DESKTOP-6E2LN7K MINGW64 /d/GIt Practice (master)

$ git status

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached <file>..." to unstage)

new file: demo1.java

HP@DESKTOP-6E2LN7K MINGW64 /d/GIt Practice (master)

$ git commit -m " my first commit"

[master (root-commit) 9c4ee0d] my first commit

1 file changed, 2 insertions(+)

create mode 100644 demo1.java

HP@DESKTOP-6E2LN7K MINGW64 /d/GIt Practice (master)

$ git branch -M main

HP@DESKTOP-6E2LN7K MINGW64 /d/GIt Practice (main)

$

HP@DESKTOP-6E2LN7K MINGW64 /d/GIt Practice (main)

$ git remote add origin https://github.com/narendradasara99/15-JRTP-02-app.git

HP@DESKTOP-6E2LN7K MINGW64 /d/GIt Practice (main)

$ git push -u origin main

Enumerating objects: 3, done.

Counting objects: 100% (3/3), done.

Writing objects: 100% (3/3), 235 bytes | 235.00 KiB/s, done.

Total 3 (delta 0), reused 0 (delta 0), pack-reused 0

To https://github.com/narendradasara99/15-JRTP-02-app.git

\* [new branch] main -> main

Branch 'main' set up to track remote branch 'main' from 'origin'.

s

------------------------------------------------------------------------

20-may

Last session : Git Bash Commands

------------------------------------------------------------------------

git help

git help <command-name>

git config --global user.name "ashok"

git config --global user.email "ashokitschool@gmail.com"

git init

git status

git add <filename>

git commit -m 'msg'

git remote add origin <repo-url>

git push

---------------------------------------------------------------------------------------------

git clone <repo-url> : To take existing project from git central repo to local system

Staged : The files which are added for commit (will display in green color)

Un-staged : The files which are already available in git repo but modified in local (Will display in red color)

Un-Tracked : The files which are newly created in local (Will diplay in red color)

-> In order to commit un-staged and un-tracked files we have to add those files to staging using git add command.

-> In order to display staged, un-staged and un-tracked files we will use 'git status' command.

--------------------------------------------------------

22-may

Last session : Git Commands

-------------------------------------------------------------------------

git help

git help <command-name>

git config

git init

git add <file-name>

git add --a

git status

git commit -m <msg>

git remoted add

git push

git reset

git clone

git checkout

git log

git rm

git pull

git stash

What is staging?

What is un-staging?

What is un-tracked?

----------------------------------------------------------------------------------

Git branches & Pull request (4th video in playlist)

Merge Conflicts (5th video in playlist)

-----------------------------------------------------------------------------------

Setting up existing project

------------------------------

-> When we join in the project, team member or TL will give project repo URL then we have to clone the project and setup workspace.

git clone <repo-url>

git clone -b <branch-name><repo-url>

Setting Up New repository

--------------------------

-> When we need to start new project development we need a repository for that.

-> Git Repository will be created by Git Admin.

-> As a developer we need to send request to Git admin team to create Git Repository for our project.

---------------------------------------------------------------------------------------------

To : gitadminteam@ibm.com

cc: manager-email, scrum-master, team-members

Subject : Git Repo Creation Request | IRCTC

Body:

Hi Team,

Please create new git repo for IRCTC project.

Repository Name : IRCTC\_Payments\_Service

Thanks,

Ashok.

-----------------------------------------------------------------------------------------------

-> After sending this email, git admin team will create repository and will share repository url in email as a reply.

-> Once repository URL is recieved, we will create project with folder strcture and will push to git repo.

-> Other developers will clone the project from repository and will start development.

---------------------------------------------------------------------------------------------

Repo URL : https://github.com/Ashok-IT-School/15-JRTP-SBApp.git

---------------------------------------------------------------------------------------------

git clone <https://javaproject027am@bitbucket.org/javaproject027am/15-jrtp-03-app.git>­­

---------------------------------------------------------

23-may

Last session : BitBucket

-------------------------------------------------------------------------

-> BitBucket is an Atlasian company product

-> BitBucket providing version control mechanism

-> BitBucket internally uses git a version control software

-> Bitbucket is having rich UI when compared with Git Hub.

-------------------------------------------------------------------------

-> We can use git bash as a client to communicate with BitBucket repositories.

-> In Bitbucket also we can create both public and private repositories.

-------------------------------------------------------------------------

-> We can use Source Tree as a client software to communicate with BitBucket repositories.

-> Source Tree is an open source GUI based application.

-------------------------------------------------------------------------

Interview Questions

---------------------

1) Which version control software is using in your project?

2) Which client software you are using to perform operations in source code repository ?

3) Who will create repository in your project?

4) What are branches in repository and why we need them ?

5) Have you ever faced conflicts ?

6) What is pull request ?

7) What is the difference between push and pull ?

8) What is the difference between clone and pull?

9) How code integration will happen in your project?

10) Have you ever created repo in bitbucket ?

11) Have you ever created branches ?

12) What is branch locking?

13) What is code freeze ?

Maven

Agile with JIRA

Git Hub

BitBucket

Source Tree

Debugging: https://www.youtube.com/watch?v=2WxsClYhreE

31-may

Last Session (23-May-2021) - Version Control Software Completed

------------------------------------------------------------------------

I have shared videos for log4j and Jenkins

------------------------------------------------------------------------

Today's session : Application Environments

------------------------------------------------------------------------

-> Environment is nothing but a platform which is used to run our application.

-> To run one application we need below setup

1) Machine

2) Operating system

3) JEE Server

4) Database Server

-> In Realtime we will have mulitple environments to run our application.

1) Local Env

2) DEV Env

3) SIT Env (QA Env)

4) UAT Env

5) Pilot Env

6) Prod Env

-> Local Env is used for the development

-> DEV env, SIT Env, UAT Env and Pilot Env are used to test our application behaviour.

-> Prod Env is used for 'Go Live'.

Note: The project which is in Prod env is accessible by end users.

1-jun

Last session : Application Environments

-------------------------------------------------------------------------

-> In Realtime every application will have multiple environments like below

1) Local Env

2) Dev Env

3) SIT Env (QA Env)

4) UAT Env

5) Pilot Env

6) Prod Env

-> In Local env we will do the development

-> Dev env will be used by developers to perform Integration testing

-> SIT env will be used by Testing Team to perform System Integration Testing

-> UAT Env will be used by client side team to perform User Acceptance Testing.

-> Pilot env is used to perform Cluster setup. Pilot is also called as Pre-Prod env.

-> Prod env is used to for Live Deployment.

-------------------------------------------------------------------------

-> All these environments setup will be done by DevOps team.

-> DevOps team will create Jobs in Jenkins Software for application deployment.

-------------------------------------------------------------------------

To : irctcdevops@ibm.com

CC : irctc@ibm.com

Subject : Jenkins Jobs Creation Request For IRCTC

Body :

Hi Team,

Please create jenins jobs for IRCTC project for below environments

1) DEV

2) SIT

3) UAT

4) Pilot

5) PROD

BitBucket URL : <repo-url>

Thanks,

Ashok.

-----------------------------------------------------------------------------------

2-jun

Last session : Application Environments & Deployment Process

-------------------------------------------------------------------------

-> For every project devops team will be available and they are responsible to create environments required for our application.

-> We will send an email for devops team to create jobs for our application deployment.

-> When we need to deploy the application we have to run the job which is created by devops team in jenkins software.

-------------------------------------------------------------------------

-> If our code is not working as expected in local env then we will use Debugging technique to understand the problem and we will fix it.

-> If code is not working in higher environments then we will check log files to identify the problem.

\*\*\*\*\*\* For every project logging will be implemented in realtime\*\*\*\*\*\*\*\*\*

package com.nare;

import java.io.IOException;

import org.apache.log4j.ConsoleAppender;

import org.apache.log4j.FileAppender;

import org.apache.log4j.HTMLLayout;

import org.apache.log4j.Level;

import org.apache.log4j.Logger;

import org.apache.log4j.PatternLayout;

import org.apache.log4j.SimpleLayout;

import org.apache.log4j.xml.XMLLayout;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

public class Test {

static Logger l = Logger.getLogger(Test.class);

static {

//SimpleLayout sl=new SimpleLayout();

// HTMLLayout hl= new HTMLLayout();

//XMLLayout hl=new XMLLayout();

PatternLayout hl=new PatternLayout();

// ConsoleAppender c=new ConsoleAppender(hl);

try {

FileAppender f=new FileAppender(hl,"log.html");

l.addAppender(f);

l.setLevel(Level.DEBUG);

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

public static void main(String[] args) {

l.debug("debug");

l.info("ifno");

l.warn(" warn messages");

l.fatal("fatal");

l.error("error");

// TODO Auto-generated method stub

ApplicationContext ac = new ClassPathXmlApplicationContext("AppConfig.xml");

Atm a = (Atm) ac.getBean("atm");

a.print("hello factory method is called");

}

}

Log messages we can write by level

DEBUG<INFO<WARN<FATAL<ERROR

1. DEBUG : in debug we write just information about class like main method executed
2. INFO: here we can write connection established like messages we can write
3. WARN: warning messages like this method is deprecated we can write here
4. Fatal : in the try block we can write
5. Error: in catch block we can write

Note: we can configure log properties in properties file

In java class we can write like this

static Logger l = Logger.getLogger(Test.class);

PropertyConfigurator.configure("src\\main\\resources\\log.properties");

l.debug("debug");

l.info("ifno");

l.warn(" warn messages");

l.fatal("fatal");

l.error("error");

log4j.rootLogger=DEBUG,r

log4j.appender.r=org.apache.log4j.FileAppender

log4j.appender.r.file=D:/info.html

log4j.appender.r.layout=org.apache.log4j.HTMLLayout

-> We will use 'Log Monitoring Tools' to check log messages of our application.

1) putty

2) winscp

3) splunk (Trending)

-> Putty is a commandline based software (We have to execute commands to see log messages of our application)

-> WinScp is a gui based application which is used to download log files from remote system to local system.

-> Splunk is an advanced web application which is used for log monitoring.

Maven

GIT Hub

BitBucket

Source Tree

Agile with JIRA

Log4j

Log Monitoring Tool

Jenkins (Ci & Cd)

3-jun

Last session : Log Monitoring Tools

------------------------------------------------------------------------

-> To identify the problems in application we will use log files

-> To monitor log files we will use log monitoring tools

1) Putty (CLI Based)

2) WinScp (GUI Based)

3) Splunk - ( Web app )(Highly demanding tool)

-> Devops team will provide details of remote mahcine or splunk URL to access logs of our application.

-------------------------------------------------------------------------

Maven : Build Tool (To automate application build process)

GIT Hub : Version Control Software

BitBucket : Version Control Software

Source Tree : Client software for BitBucket

Agile with JIRA : Project Management Software

Log4j : To perform logging in the application

Putty, WinScp, Splunk : For logs monitoring

Jenkins : To automate application deployment process (CI & CD)

-----------------------------------------------------------------------------------

Part-3 : Mini Projects Development

-----------------------------------------------------------------------------------

-> For next one month we will do Mini projects implementation

Note: Everybody has to implement code for mini projects in this one month.

-> We will implement these mini projects with Realtime standards.

-> In these 3 mini projects we are going to use tools also.

Note: Daily assignments will be available related to mini projects and assignment checking will be there. The zoom link will be provided only for assignment completed students.

-----------------------------------------------------------------------------------

What is project?

----------------

Set of programs we can call as a project

Why we need to develop project?

-------------------------------

Software projects are used to reduce human efforts

Ex: Netbanking, Ticket Booking, Online Shopping

----------------------------------------------------------------------------------

-> Service based companies will take requirements from client and will develop the project based on client requirement.

-> Product based companies will develop the product and will sell that product to public in the market.

-----------------------------------------------------------------------------------

-> In Every company functional team will be available.

-> Functional team is responsible to collect requirements from the client.

-> Functional team will prepare BRD (Bussiness Requirement Document).

-> Functional team will submit BRD to client for approval.

-> Client will review BRD and will approve BRD.

-> Once BRD got approved, Functional team will prepare FDD (Functional Design Document).

-> Functional team will share FDD to client for review

-> Client will review FDD and will approve the FDD if everything is correct.

-> Once FDD got approved, functional team will share FDD to Developers & Testers.

-> Developers and Testers should read the FDD and should understand requirements clearly.

-> If any developer or tester having doubts in FDD then they should note down their doubts and they should ask functional team for clarifications.

4-jun

Last session : BRD and FDD

-------------------------------------------------------------------------

-> BRD stands for Business Requirement Document

-> BRD will be prepared by functional team

-> BRD document contains highlevel requirements

-> FDD stands for Functional Design Document

-> FDD will be prepared by functional team

-> FDD contains detailed information related to requirements

-> Developers will write the code based on FDD provided information

-> Testers will test the application behaviour based on FDD provided information

-> Once FDD got approved by client, developers and testers will read the fdd and will notedown the queries on FDD.

----------------------------------------------------------------------------------

Q-1) Do we need client side validations or not?

Q-2) What is minimum and maximum length for Name field?

Q-3) Phone Number should take how many digits ?

Q-4) Do we need to implement Soft-Delete or Hard-Delete? - Softdelete

Q-5) Can we save duplicate contact ? - No

Q-6) Do we need to implement Pagination or not ? -

-----------------------------------------------------------------------------------

-> After understanding FDD, developers and testers will get some queries.

-> Developers and testers will write their queries in query log document.

-> Functional team will read query log and will answer queries

-----------------------------------------------------------------------------------

Today's assignment : Analyze how many classes/interfaces required for 01-mini project.

-> Analyze methods which are required for each class with proper methodname, method parameters and method return type.

ContactEntity.java

--------------------

ContactRepo.java

-----------------

ContactService.java

---------------------

ContactController.java

-----------------------