**Spring**

**26th april**

* Spring Core
* Spring MVC
* RestAPI
* MongoDB
* Spring Boot
* JWT
* Mockito
* Swagger
* Microservices
* Angular & Spring
* Spring Cloud
* Docker

**Maven - POM -(Project Object Model**)

* it is a powerful project management tool.
* it is used for projects to build, depedency and documentation.

ANT - it is similar to maven

problem which we will not face when we go for maven

1. adding set of jars file for a project
2. creating the right project structure
3. building and deploy the project

Repository - packaged jar files

added depedency in the pom.xml file

Maven repository

1. local

it is located in the local system. it is created by the maven it self.

user profile

.m2 folder (repository)

2. central repository

it is located on the web. it is created by apache community itself.

https://repo1.maven.org/maven2/

3. remote repository

it is located on the web. Most of the libraries can be missing from the central repository

such as JBoss library, weblogic, websphere (web server)

**POM.xml**

* it contains all the information of project and configuration information
* it contains all the elements
* project - project name - root element
  + modelversion - version sub
  + groupid - sub element of the project
  + artifactid - package name
  + version - sub element version

* 1. packaging - jar (console application), war (web application)
  2. name - project name
  3. url - define url of the project
  4. dependencies - root element which contains all the depedency
  5. depedency - all the jar which need to be loaded in the project
  6. scope - compile, test, runtime

**maven –**

* clean - clean the project before install
* install - install the dependencies
* build – compile
* test - running test cases

maven -> update project -> for updating the existing project

**28th april**

Spring it is a framework which makes easy in developing the JavaEE application.

* Java enterprise edition
* lightweight framework.
* framework contains many such as struts, hibernate, jsp, jsf, EJB and so on.

Spring framework consists of many modules

* IOC (inversion of control),
* DAO - data access object,
* ORM - object relational mapping,
* Web MVC (Model view controller) , etc...

**IOC - dependency injection**

* design pattern (used to remove depedency in the code).
* code easier to test and maintain.

class Employee {

Address address;

}

* tight coupling - (dependent)
* loose coupling - dependency injection
* can be implemented using XML configuration and
* Annotation

**Advantages - Spring Framework**

1. Predefined templates

JDBCTemplate, JPARepository

2. Loose coupling

3. Easy to test - unit testing (mockito)

4. Light Weight

5. Fast development

**DI - implemented using two ways**

* by setter method
* by constructor

@AutoWired

@Component

JDBCTemplate

Spring MVC

JDBCTemplate

Java Database Connectivity

connect to any database - MySQL, SQL Server, Oracle

JDBCTemplate is a powerful mechanism connecting to database and perform DML, DQL statement - insert, delete, update, select.

**Problems which u faced when u go for JDBC API**

1. open and close connection

2. handle exception handling

3. few code consume a lot of time to perform a task

insert

select

**Advantages of using JDBCTemplate**

* it contains a list of functions to perform DML, DQL statements
* it saves a lot of time and work.

org.springframework.dao package

* update()
* execute()
* query()

**Spring MVC (Model, View, Controller)**

* + it is used to build web applications.
  + MVC - design pattern.
  + MVC - DispatchServlet is a class which receives the incoming request and maps it to the right resource.
  + such as controllers, models and views.

model

FrontController -> Controller -> View

* + Model - which contains the data of the application. (single object or a collection of objects)
  + Controller - @Controller
    - which is used to mark the class.
      * business logic of an application
  + View - represent information in a particular format.
  + JSP + JSTL is used to create view pages.

FrontController

used for dispatcherservlet class for working and also to represent the flow of the appication.

**Advantages of Spring MVC**

1. Separate roles

loose coupled

2. light weight

develop and deploy the application is done using maven project

3. powerful configuration

4. Resuable of business code

5. Rapid development

faster and parallel development

6. Easy to test

7. Flexiable mapping

redirect to the different pages