Phase 2

SQL Self Learning using My SQL

JDBC: Connecting MySQL or Oracle database using Java Technologies.

We can insert, delete , update and retrieve records from database.

Hibernate : ORM (Object Relation Mapping ) : Advanced of JDBC to store, retrieve, update and insert record using Java Technologies.

Servlet and JSP : Using these two modules we can create the web application using Java technologies.

In Virtual open the terminal and type

mysql –u root –p

Simplilearn

mysql >

Database is use to the data in table format.

We can connect the mysql database two ways

1. Using command prompt
2. Workbench (GUI base ).

After connected mysql database using command prompt

show databases; This command is use to display all databases present in mysql database.

Create database databaseName;

create database maybank

use databasename; This command is use to switch in database.

Syntax to create the table

Employee --🡪Table

Three column id name salary

Number String float

create table tableName(colummName datatype, columnName datatype);

create table employee(id int primary key,name varchar(10), salary float);

command to insert the record in Table

insert into employee values(1,’Raj’,12000);

To view the records from a table

select \* from employee

retrieve the record using conditions

select \* from employee where id=1;

select \* from employee where name = ‘Raj’;

select \* from employee where salary > 14000;

update query

update employee set salary = 22000 where id=1;

delete query

delete from employee where id=1;

JDBC : Java Database Connectivity : JDBC is a API (Application programming interfaces) which help to connect the RDBMS (oracle or Mysql) using Java technology.

Relational Database Management System.

JDBC always throw checked exception. So we have to handle this exception using try-catch or throws.

1 we have to load the Driver : Driver is a pre-defined class provider by vendor in the form of jar file which help to connect the database.

Java provided pre-defined class Class. which contains pre-defined method forName() and it is a static method.

Class.forName(“driverName”);

com.mysql.jdbc.Driver :5.x

com.mysql.cj.jdbc.Driver :8.x

Establish the connection

Now we have to create the reference of PreparedStatement. Which provide set of method which help to do insert, delete, update and retrieve.

Don’t write any business logic and database logic in main method.

Maven Project :

Maven is known as build tool. This tool responsible to compile the program, run the program, creating jar or war. Downloading the dependencies for the project.

Pom.xml ( project object model) file. This file is known as a maven deployment description file.

Database Table -------🡪 Employee ---🡪 ID,Name,Salary (columns)

Java (JavaBean) -🡪Employee--🡪id,name,salary (variable)

DAO Layer (Data Access Object ) : This class is responsible to write pure database logic.

EmployeeDao : this class contains set method which help to do database operation on table. Like insert, delete, update and retrieve.

Service layer : This layer is responsible to write business logic. We can write business logic before interact with database or after retrieve records from database.

EmployeeService : pure business logic

Class Employee {

Id,name,age

Setter and getter methods

}

class Information {

void passInformation(Employee emp) {

}

}

1st person

Information in = new Information();

Employee emp = new Employee();

emp.setId(1);

In.passInformation(emp); hold only id

2nd person

Information in = new Information();

Employee emp = new Employee();

emp.setId(1);

emp.setName(“Ravi”);

In.passInformation(emp);

3rd person

Information in = new Information();

Employee emp = new Employee();

emp.setId(1);

emp.setName(“Ravi”);

emp.setAge(21);

In.passInformation(emp);

**ORM : Object Relation Mapping**

**ORM is a concept.**

**Limitation of JDBC**

1. **Using JDBC we can’t store Java object into database or we can’t retrieve Java object from a database. In DAO layer we have to convert Java object into sql query format and sql query format into Java object.**
2. **JDBC use SQL Language. SQL is database dependent language. So when we move from one database to another database we have to change the query.**
3. **JDBC through checked exception. All exception hierarchy is database dependent.**
4. **JDBC doesn’t relationship is a and has relationship.**

**Object Relation Mapping**

**JavaBean or Entity class**

**@Entity**

**class Employee { Employee table**

**@Id**

**id,name,salary id,name,salary**

**}**

**Mapping**

**Employee ---🡪Employee**

**Id -🡪ID PK**

**Name🡪name**

**Salary 🡪salary**

**Old Version we are doing mapping using xml file**

**New version we are doing mapping using annotation**

**In ORM we provide all database details in xml configuration file.**

**driverName**

**url**

**username**

**password**

**mapping class details.**

**Hibernate and JPA (Java Persistence API) are implementation of ORM.**

**Hibernate is a third party framework which support ORM features.**

25-04-2022

We have to the Entity class means class bean class on that class we have to use the annotation @Entity and @Id the column which contains primary key.

If we do any operation like Insert, Delete and Update using JDBC it will auto commit.

But if we want to do the operation through Hibernate we have to use Transaction. Through Hibernate it will not auto commit.

26-04-2022

If we are planning retrieve multiple records using hibernate. Hibernate provide their own query language

Ie HQL (Hibernate Query Language).

Sql (Structured Query Language) : using sql we can insert, delete, update and retrieve.

Select \* from employee; (employee is table name it is not a case sensitive).

We are retrieving all column from a table.

Select \* from employee where id=100;

Select \* from employee where name = ‘Ravi’;

Select \* from employee where salary > 12000;

HQL use only for retrieve purpose

Hql (Hibernate Query Language)

Select emp from Employee emp (Employee is java bean class name and emp is object consider)

Select emp from Employee emp where emp.id = 100; (emp is object and id is variable of java bean class)

Master / main ---🡪

Demo.java (some code)

Push this code in remote repository

Git clone URL

Demo.java (some code)

Master branch or main branch (if you do directly changes on my code with master branch or main branch)

Git pull

Create new folder (new repository)

Inside that folder

Git clone https://github.com/Kaleakash/maybank\_2022\_java\_full\_stack\_batch.git

int a; your code

int a; my code

SQL relationship

One to many : Trainer --🡪 Student (PK and FK)

One to One : Person 🡪 Passport (one to One)

Many to one : Employee -🡪 Department

Many to Many : Employees -🡪 SkillSet

TrainerAndStudent

TId TName tech SId SName age

1 Raj Java 100 Seeta 21

1 Raj Java 101 Reeta 22

1 Raj Java 102 Meeta 23

Trainer

PK

TId TName tech

1 Raj Java

2 Ravi Python

Student

PK FK

SId SName Age TSId

100 Seeta 21 1

101 Reeta 22 1

102 Meeta 23 1

103 Leeta 24 2

FK : Foreign key: Foreign key is use to connect the pk of another table. And it allow only those values which present in primary key and it can allow duplicate. Primary key doesn’t allow duplicate.

PK : Primary key : if column contains primary key it doesn’t allow duplicate.

create table trainer(tid int primary key,

tname varchar(10),

tech varchar(10));

create table student(sid int primary key,

sname varchar(10),

age int,

tsid int,

foreign key(tsid) references trainer(tid));

Hibernate Relationship

17-05-2022

JEE : Java Enterprise Edition : This is use to create the web application.

<http://www.google.com> 🡪 URL (Uniform Resource Locator)

http: protocol : hyper text transfer protocol secure (set of rules

while help to communicate more than one machine)

www🡪 word wide web

google 🡪 domain name

com 🡪 commercial application

request (http/https)---------🡪

Client Server

🡨----------Response (http/https) html

Css

JS(JavaScript )

Html : hyper text mark up language : which is use to create the web page. Html is use to display the content on browser.

CSS : it provided lot of pre-defined properties which help apply good look and feel for web page.

JS (Java Script) : it is use to do client side validation.

Now way day we are doing client side validation using JavaScript or HTML5 features.

JSE (Core Java ) : Standalone application or desktop application

JEE (Web Application ) : Web application mean web sites.

Server side technologies

Java

JEE (Servlet /JSP)

Or

Spring framework

Spring boot

Asp.net

Php

Python

Node JS

Servlet : Servlet is normal Java program which help to create dynamic web page on server side.

To run Servlet or JSP program we require a server.

There are lot of different type of server

Tomcat : it is a type of web server which help to run the servlet and jsp program.

We can’t write main method inside servlet or jsp.

Inside a Server like Tomcat which contains container. Container is a part of server which is also known as engine. Container is responsible to execute the servlet and jsp program.

Container is responsible to load the class, create the object, call the life cycle method and destroy the object.

We can use different type of server

Tomcat : it is open source server.

Web Sphere (WAS)

Glashfish

Web Logic

JBoss

24-05-2022

Servlet provide pre-defined API that is RequestDispatcher. It is use to move from one page to another page base upo condition like navigation.

RequestDispatcher rd1 = request.getRequestDispacher(“path”);

If target page is another servlet page then we have to give path as target page URL pattern.

If target page is html page then we have to give URL pattern as pagename.html

RequestDispatcher rd1 = request.getRequestDispacher(“Home”);

RequestDispacher rd2 = request.getRequestDispacher(“login.html”);

rd1.forward(request,response); we get output of target page only.

or

rd1.include(request,response); we get source + target page as one output.

rd2.forward(request,reponse);

or

rd2.include(request,reponse);

JSP : Java Server Pages : JSP is type of server side scripting language which help to create dynamic web page on server side.

Limitation of servlet

1. Servlet is normal java program if we do any changes in servlet we have to recompile and redeploy (re run ) the program once again on server.
2. Inside a servlet if we want to write any html code we have to write inside pw.println(“<b>Welcome to Servlet </b>”); inside a double quotation it is string consider.
3. Servlet is a complex. If you want to display any simple message through servlet program we have to create normal java class and that class must be extends or implements type of servlet. Then we have to override life cycle method or doGet or doPost. Then we have to create the PrintWriter class object. We have to provide the servlet configuration details in web.xml file.

JSP provide lot of pre-defined tags

1. Scripting tag
   1. Scriptlet tag

<%

Java code means the code which we write inside a doGet or doPost

%>

Inside a scriplet tag what are code we write it is consider as method code so if we declare any variable inside scriptlet tag is consider as a local and local variable doesn’t default value.

* 1. Declarative tag

<%! Variable declaration %>

* 1. Expression tag

<%=

%>

* 1. Jsp comments

<%-- JSP Comments --%>

1. Implicit object : web container provided lot of pre-defined object and those object we no need to create ie known as implicit object.
   1. out : out it like a PrintWriter class object.
   2. request : it is like a HttpServletRequest object.
   3. response : it is like a HttpServletResponse object

Limitation of JSP

1. JSP is not secure. If we write any business logic or database login using jdbc or hibernate. That code any one can see.
2. JSP is a type of servlet only. JSP internally convert to Servlet program. That phase is known as Page Translation phase. So performance wise jsp is slower than servlet.

If we write any business logic or database logic using jdbc or hibernate inside a servlet that code become local to that servlet.

MVC : Model View Controller or MVC is architecture or Design pattern.

View 🡪 presentation logic or look and feel -----🡪 HTML or JSP

Controller --🡪 Servlet

Controller -🡪

Controller 🡪

Model layer ----🡪 It is normal Java classes which is responsible to write the business logic or database logic.

31-05-2022

MVC : Model View Controller

View 🡪 Presentation logic : Look and feel or designing -🡪 HTML or CSS or jsp

Controller -🡪 Servlet (LoginController)

Model 🡪 JavaBean class, Service class, Dao class etc.

Create the table in database as

create table login(email varchar(25) primary key,password varchar(25))

View ----🡪 index.jsp, signIn.jsp and signUp.jsp, Home.jsp (JSP)

Controller 🡪 LoginController (Servlet )

Model 🡪 Login ---🡪 JavaBean class or Entity class (Normal Java classes)

LoginService --🡪 business method class

LoginDao 🡪 Database code using JDBC or Hibernate

hibernate.cfg.xml 🡪 database details

index.jsp --🡪 In this page we include signIn.jsp page 🡪 In signIn.jsp page we created login page with action as LoginController with method is get and we are passing email and password to LoginController to doGet method. This page contains signUp.jsp as hyperlink. If we click on signUp.jsp page it will open another page with form tags. This form also contains email and password with action as LoginController and method is post.

If we don’t have account first we will create the account using signUp.jsp page. On we click on signUp button the value past to LoginController doPost method.

In doPost method we will create the PrintWriter class object. Receive the value as email and password from form. Then we will create the Login class object and set the value. Then we will create the LoginService class object and pass this object to service layer. Then base upon service layer output we will re direct to different jsp page with the help of ReqeustDispatcher with include or forward method.

Now flow come to LoginService class business method is createLoginDetails.

Before passing the login object to DAO layer please apply business logic details upon application requirements. Ie converted password in reverse order. After business method we will create the LoginDao class object and pass the login class object to Dao layer ie LoginDao.

In LoginDao we can use jdbc or Hibernate to store, delete, update and retrieve records from database.

If you want to please clone my repository

Git clone <https://github.com/Kaleakash/maybank_2022_java_full_stack_batch.git>

Or

In existing repository use git pull

git checkoub my\_branch

**13-06-2022**

**create table product(pid int primary key,pname varchar(25), price float);**

**14-06-2022**

**Login table**

**Create table login(emailed varchar(20) primary key,password varchar(20));**

**Insert into login values(‘admin@gmail.com’,’admin’);**

classes, subjects, students, and teachers

classes -🡪 table

PK

Cid class

11 1st

22 2nd

33 3rd

subjects 🡪 table

PK

Sid Subject

* + 1. Maths
    2. Phy
    3. Che

students -🡪 table

PK FK

Sid SName classid

111 Ravi 11

teachers -🡪 table

PK

TId TName

1111 Ajay

2222 Vijay

**Spring**

**Flight**

**FI Source destination date numberOfTicket price**

**5 records**

**Search the flight**

**Source**

**Destination**

**Date**

**2 flight available**

**date**

**Abc123 Book**

**Xyz456 Book**

**FName**

**LName**

**Age**

**Fid**

**Source**

**Destination**

**Price**