Day 14 Revisit

What is Maven?

* Maven is a project Management Tool.
* Maven helps to build, test, package, deploy, install & manage dependencies of project
* Build means converting all source code to byte code (.java 🡪 .class conversion)
* Test means Unit Testing the application code
* Package means converting the project as a single jar/war file (That can be executed/run anywhere)
* Install – downloading all the dependencies & generating boiler plate codes
* Dependency Mgmt means taking care of dependent jars of the project

List of maven commands

mvn init

mvn build

mvn test

mvn clean

mvn package

mvn install

What is the backbone of maven project?

POM.xml is the backbone of maven project.

What is POM stands for?

POM Stands for Project Object Model. It is a XML file which contains project details along with it’s dependencies.

What are important/mandatory details of maven project?

Each maven project will have unique groupid, artifactid & version number.

JDBC CRUD Operation

What is mean by CRUD operation?

C – Create /Insert Operation

R – Read Operation (Select all/Select particular)

U – Update Operation

D – Delete Operation

What are the important classes & interfaces used in JDBC api?

Classes :

* Date
* Time
* TimeStamp
* DriverManager

Interfaces:

* Driver
* Connection
* Statement
* PreparedStatement
* CallableStatement
* ResultSet
* DatabaseMetaData
* ResultSetMetaData
* RowSet

Where the JDBC api’s are defined?

All the classes & interfaces of JDBC api are defined in java.sql package.

Steps in interacting with Database

1. Load & Register the Driver [Add the Driver Jar to Build path of the Java Project]
2. Establish the connection between Java and RDBMS
3. Create & Execute the Query using Statement/PreparedStatement object.
4. Store & Process the result using Resultset object
5. Close all the resources

Regular methods used for CRUD operation

1. readAll()
2. readById(int id)
3. insert( Entity obj)
4. update (int id, Entity obj)
5. delete(int id)

Model or Entity or Bean represents same.

CREATE TABLE `lrs`.`user\_table` (

`id` INT NOT NULL AUTO\_INCREMENT,

`username` VARCHAR(45) NOT NULL,

`password` VARCHAR(45) NOT NULL,

`first\_name` VARCHAR(45) NULL,

`last\_name` VARCHAR(45) NOT NULL,

`email` VARCHAR(45) NOT NULL,

`phone` BIGINT(10) NULL,

`role` VARCHAR(45) NOT NULL,

PRIMARY KEY (`id`),

UNIQUE INDEX `email\_UNIQUE` (`email` ASC) VISIBLE);

INSERT INTO `lrs`.`user\_table` (`username`, `password`, `first\_name`, `last\_name`, `email`, `phone`, `role`) VALUES ('siva123', 'siva123', 'siva', 'os', 'siva.os@gmail.com', '9867564534', 'Employee');

INSERT INTO `lrs`.`user\_table` (`username`, `password`, `first\_name`, `last\_name`, `email`, `phone`, `role`) VALUES ('admin123', 'admin123', 'Admin', 'User', 'admin.user@gmail.com', '9878675667', 'Manager');

CREATE TABLE `lrs`.`leave\_request` (

`id` INT NOT NULL AUTO\_INCREMENT,

`no\_of\_days` INT NOT NULL,

`requestor` INT NOT NULL,

`approver` INT NULL,

`status` VARCHAR(15) NOT NULL,

PRIMARY KEY (`id`),

INDEX `req\_fk\_idx` (`requestor` ASC) VISIBLE,

INDEX `appr\_fk\_idx` (`approver` ASC) VISIBLE,

CONSTRAINT `req\_fk`

FOREIGN KEY (`requestor`)

REFERENCES `lrs`.`user\_table` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `appr\_fk`

FOREIGN KEY (`approver`)

REFERENCES `lrs`.`user\_table` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION);

ALTER TABLE `lrs`.`leave\_request`

ADD COLUMN `approve\_date` DATETIME NULL AFTER `request\_date`,

ADD COLUMN `reason` VARCHAR(75) NULL AFTER `approve\_date`,

CHANGE COLUMN `request\_date` `request\_date` DATETIME NULL DEFAULT CURRENT\_TIMESTAMP ;

INSERT INTO `lrs`.`leave\_request` (`no\_of\_days`, `requestor`, `status`) VALUES ('2', '1', 'pending');

User\_table

|  |  |  |
| --- | --- | --- |
| Sl No | Column Name | Column type |
| 1 | Id (PK) | Int |
| 2 | Username (NN) | Varchar(45) |
| 3 | Password (NN) | Varchar(45) |
| 4 | First\_name | Varchar(45) |
| 5 | Last\_name (NN) | Varchar(45) |
| 6 | Email (UQ NN) | Varchar(75) |
| 7 | Phone | Bigint |
| 8 | Role (NN) | Varchar(15) |
| 9 | address | Varchar(75) |

Leave\_request

|  |  |  |
| --- | --- | --- |
| Sl No | Column Name | Column Type |
| 1 | Id (NN) | Int (pk) |
| 2 | No\_of\_days (NN) | Int |
| 3 | Requestor (NN) | Int (fk) |
| 4 | Approver | Int (fk) |
| 5 | Status (NN) | Varchar(15) |
| 6 | Description | Varchar(75) |
| 7 | Request\_date (NN) | DATETIME Default CURRENT\_TIMESTAMP |
| 8 | Approved\_date | DATETIME |
| 9 | Reason | Varchar(75) |

DAO – Data Access Object