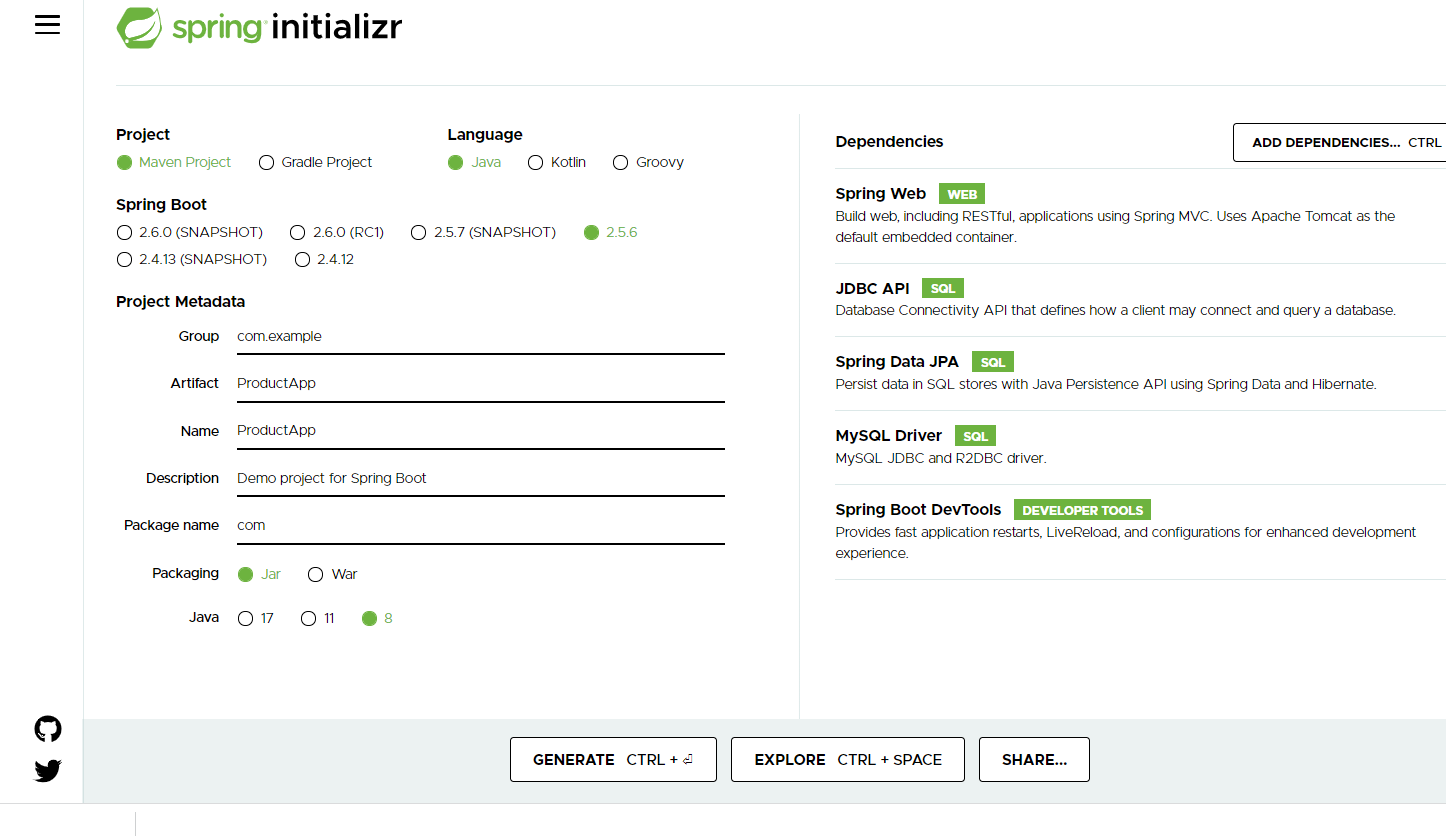
**10-11-2021**

First create the folder as

**Product Management System**

Then create two folder

Backend and frontend



After created backend using Spring boot with Reset and Spring data

Write set of REST API methods.

Then create the Angular project

ng new angular-product-app

routing 🡪 yes

styling 🡪 CSS

ng g c product component

ng g s product service

ng g class product model class

Insert operation

Using Template driven form

We are running two application or domain

Front end running on port number 4200

Backend running on port number 9090

CORS : Cross Origin Resource Sharing

By default HttpClient get, post, put and delete method return type is JSON consider ie Response.

12-11-2021

RDBMS : Relational Database Management System

My SQL or Oracle or Db2 or Postgres SQL etc

All RDBMS Database are schema base ie

Table

Emp

Id Name Age city PhNumber

100 Raj 21 null null

101 Ram 23 null null

102 Ravi 26 Bangalore null

103 Ramesh null null 9900

In Angular side we are using JSON and Spring boot we are using JSON

In Old version to store the object or retrieve object we were converting Json Data into java object and vice-versa.

JSON Format

Front end side backend side database side

In Json value can be number, string , boolean array type, array object type, complex object etc.

{“id”:100,”name”:”Ravi”,”age”:21,skillset:[“Java”,”Angular”],”address”:{“city”:”Bangalore”,”state”:”Kar”}};

Mongo DB is type of No SQL database. It is open source allow to store the data in JSON format.

RDBMS Mongo DB

Database database

Table Collection

Attribute attribute or property

Records document

In RDBMS every records must be type in single table.

But in Mongo DB every document can be same or different in collection.

Default path for mongo DB server files are

In C drive create the folder with name as

data inside data folder create sub folder with name as

db

now open the command prompt in bin folder of mongo DB database folder.

Ie

C:\Program Files\MongoDB\Server\5.0\bin

To run the server we have to open the command prompt in inside bin folder and run the command as

mongod (this command is use to run the mongo db database).

Then open another command prompt to run the mongo client shell

mongo

to Clear the mongo shell we have to use the command as

Cntr + L

show databases

Or

show dbs

use databasename; This command is use to create the database and switch inside a database if database is not present if database present it switch to that database.

To check the collection (tables)

show collections

Or

show tables

syntax to create the collection

db.createCollection(“Emp”);

insert document in collection

db.collectionName.insert({key:value,key:value});

db.Emp.insert({name:”Ravi”,age:21});

now if want to view the document from collection we have to use the command as

db.collectionName.find();

db.Emp.find();

In Mongo DB we can insert the document inside a collection without creating collection.

If we want to retrieve specific property or properties from a collection.

db.CollectionName.find({condition},{filter property});

db.Employee.find({},{name:1}) : it is display name as well as \_id pre-defined property

db.Employee.find({},{name:1,\_id:0}) : it is display only name.

db.Employee.find({},{name:1,\_id:0,age:1}) : it is display name and age.

Conditions

db.CollectionName.find({condition});

db.Employee.find({\_id:1});

db.Employee.find({city:"Bangalore"});

relational operators

db.Employee.find({age:{$gt:25}})

db.Employee.find({age:{$gte:25}})

db.Employee.find({age:{$lt:25}})

db.Employee.find({age:{$lte:25}})

db.Employee.find({age:{$eq:25}})

db.Employee.find({age:{$ne:25}})

and / or conditions

db.Employee.find({$and:[{\_id:1},{name:"Ravi"}]});

db.Employee.find({$or:[{\_id:2},{name:"Ravi"}]});

update the document using some conditions

db.collectionname.update({condition},{$set:{property:value}});

db.Employee.update({\_id:1},{$set:{age:30}});

db.Employee.updateMany({city:"Bangalore"},{$set:{city:"Pune"}});

remove the documents

db.collectionName.remove({condition});

Testing : Testing is use to find the defect or error or bugs.

Program

Input read a,b

Process compute sum = a+b;

Output write sum

Two types of testing

1. Black box testing
2. White box testing

Black box testing

Input Process Output

Username and password

Pass correct

Success

Else

Fail

If expectation and actual both are match test pass or fail.

White box testing

Input Process Output

Unit testing : Unit testing is a type of white box testing. Unit testing is use to check the individual unit of source code that code may be written in function or method or modules or procedure etc.

Java provided open source light weighted jUnit framework which help to do the unit testing.

public int add(int x, int y) {

int sum = x+y;

return sum;

}

Junit test case and test suite

Test case : it is type of junit test class which contains more than one test method which help to do the testing for particular functionality.

jUnit life cycle function or hook of jUnit

Test suite

Test case : it contains one or more than one test function which help to test the business logic functionality.

We can execute only one test case file.

Test suite : Test suite is use to execute more than one test case and test case contains more than one test function to test the business logic.

Test suite is a group of more than test cases.

In Test suite they use two annotation

Test suite is normal java class with annotation as

@Runwith(Suite.class)

And second annotation is

@SuiteClasses({provide all suite classes details here})

In Enterprise application every layer or tier depends upon another layer or tier.

Like

Controller is depends upon the Service layer

Service layer depends upon Dao layer

Dao layer depends upon resource layer

Resource layer depends upon the database.

Using mocking / fake : Controller layer do the mocking or provide the fake data of service layer.

Service layer can do the mocking for dao layer or provide the fake data.

Mockito / jMockit : these are open source framework we can do mock of object using jUnit or TestNG framework.

16-11-2021

TestNG TestNG is an automation testing open source framework.

NG “next generation”. Test NG is base upon the junit testing.

jUnit testing mainly to do the testing for functionality or methods.

TestNG mainly to do the end-to-end testing.

Using Test NG we can generate the report. we can check how many test are pass or fail. We can skip few testing methods.

We can do the testing selenium with TestNG

By default Test NG plugin not available in eclipse IDE.

TestNG framework provide the XML file. It is known as Test NG Configuration file which help to organize all testing methods properly. It allow tester or developer to handle more than test cases or classes, defined the test suite. Using this file we can control the flow the testing application.

Grouping the Test function :