FSD Training

Linux

GIT

Java

UI – HTML, CSS, Javascript

Angular & React Introduction

Hibernate

Spring

Spring Boot

Cloud computing technologies

GIT:

It is a version controlling system, which helps people to collaborate their work so that at the end every user will have other users work in their machine as well as in remote machine.

Git provides an Online Remote Repository which users need to clone

List of GIT commands:

git clone: Clones the remote repository in your local machine

git push: Pushes the local repository changes to the remote repository

git pull: Pulls the remote repository changes to the local repository

git commit: Commits the changes done by the user & creates one unique id

git add: Tracks the changes in the working directory for commit

git status: Shows list of tracked & untracked changes in the working directory

Things you need are:

Git account: Create it from your personal ID

Git Bash: Install GIT in your machine so that you get GIT bash, which is a terminal

Git Branch:

Branch is like a pointer which will have the work with series of some commit ids, by default GIT maintains a default branch with a name called master/main

Steps we did

1. Created a Remote Repository
2. Cloned the Remote repository in a folder
3. Navigated to the repository in the local machine
4. Created a file and added some content
5. Used git add command
6. Used git commit command
7. Used git push command
8. Observed that Remote repository got the updated from the local repository
9. Created another folder Developer1 and cloned the remote repository
10. Created another folder Developer2 and cloned the remote repository
11. In Developer1 folder navigated to the local repository & created some files, then entered git add, git commit, git push
12. In Developer2 folder navigated to the local repository & created some files, then entered git add, git commit, git push
13. Remote rejected the push, so in the local repository we entered git pull, here if its fast forward merge then it doesn’t create a commit else it asks us to create a commit for merge

What happens if both the developers try to edit the same file & uploads to the Remote Git

Then Git doesn’t automatically merge instead users need to manually resolve the conflicts and merge the work

Why we should not work with the master branch?

In Real time there could be chance that users may push errors in master and master branch will be automatically merged without any review by any users, hence we must control it by creating a feature branch which is a copy of master branch, so that users can work with the feature branch & push the feature branch so that somebody will review that branch before merge

Feature branch

>> git branch branch-name

The above command creates feature branch

Checkout to branch

When you want to work in a different branch you need to checkout, the work you do in one branch is not updated to another branch automatically

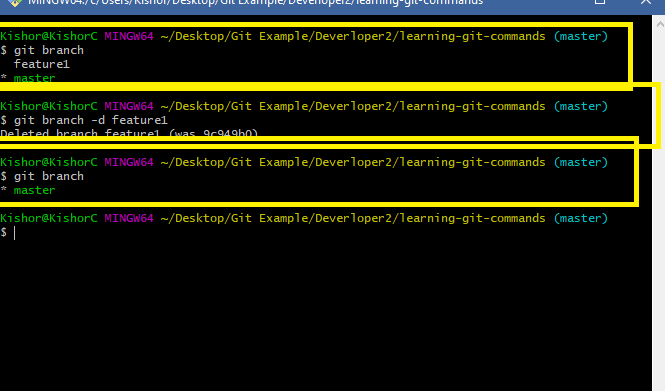
>> git checkout branch-name

Note:

1. Never work in the master branch
2. Always make commits or new changes in the Feature branch
3. Push the feature branch to the remote
4. In Remote the feature branch can be merged or it will be closed without merging

How to delete a branch

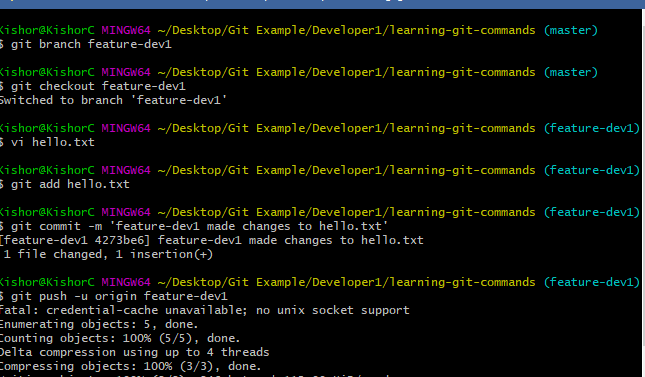
>> git branch -d branch-name



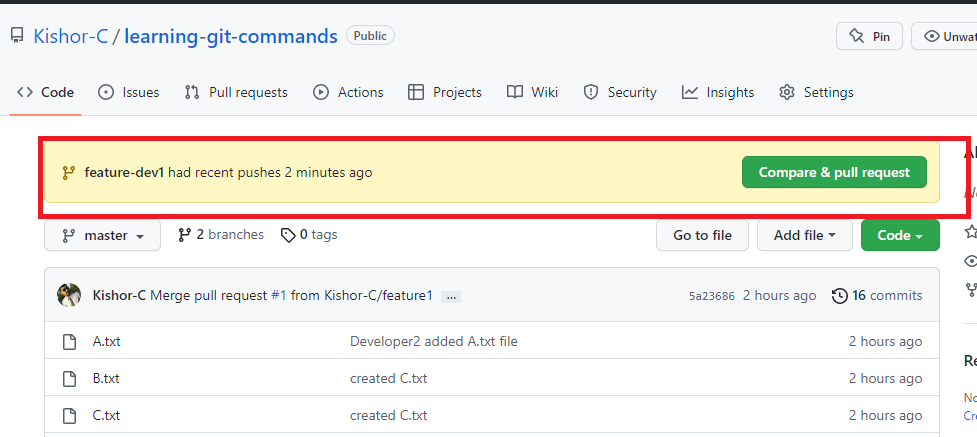
What to do when the remote can’t merge the feature branch

In the remote whenever there’s a Merge conflict you can close the pull request without merging & delete the branch, so that the user who pushed the branch should able to resolve the conflict and again push the feature branch

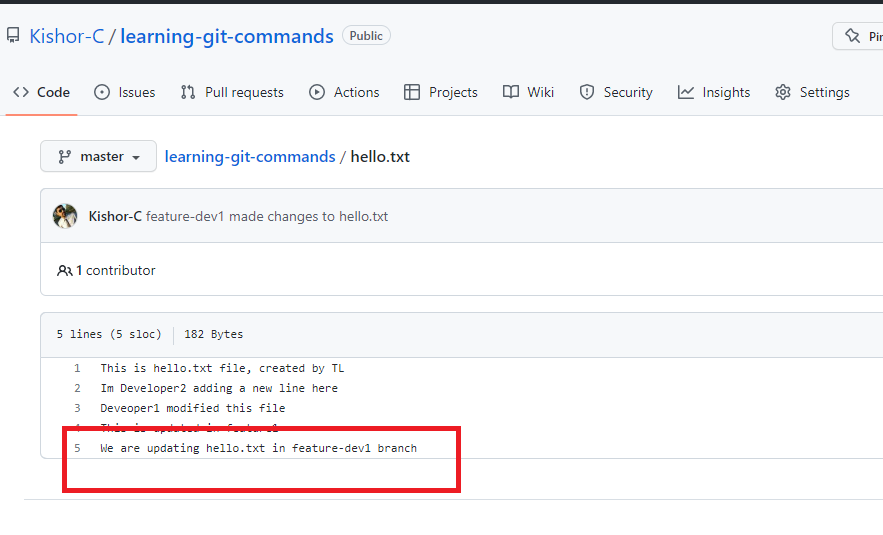
Let us create a new branch in the developer1 terminal & edit hello.txt with some content & push that new feature branch



We must see the same changes in the Remote, but we get a pull request in the Remote

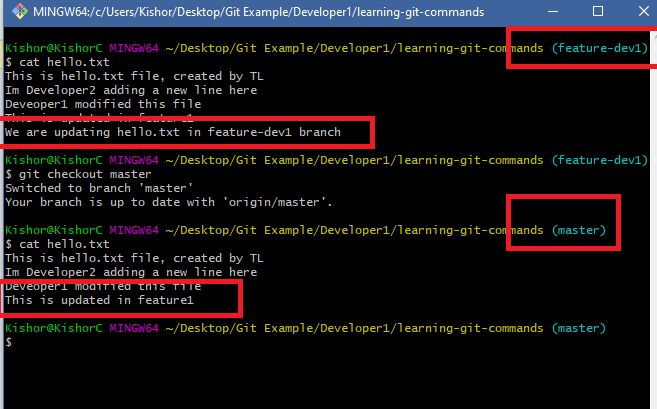


You need to click on Compare & pull request & check for merge option, if possible you can merge else you can close pull request



The highlighted part is the change done by developer1 in the feature-dev1 branch.

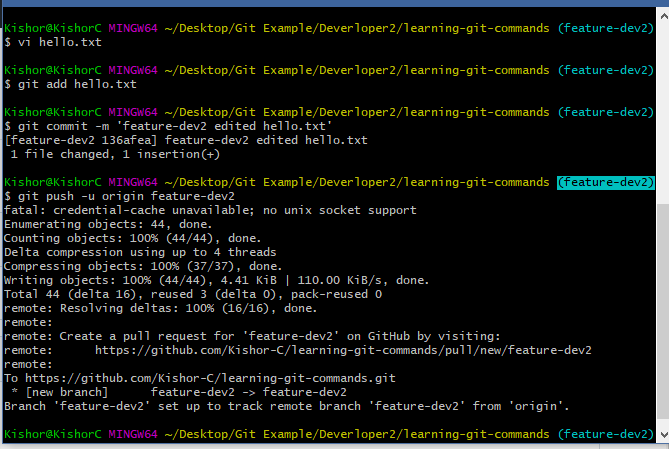
Note: Even in Developer1 terminal we have a master branch which is also not having this change



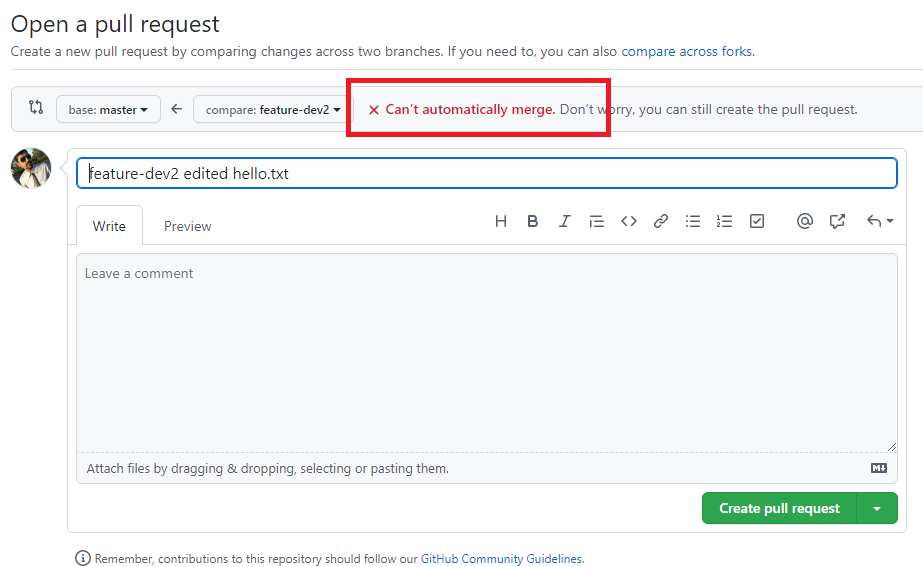
There are two ways in Developer1 Local repository to get master branch up to date changes of feature branch

1. use git merge in local repository in the master branch(git merge feature-dev1): This is not recommended as it gets only local repository changes, but it doesn’t get any remote repository changes
2. use git pull in local repository so that it pulls all the updates from remote repository: This is appropriate because it always gets changes done by other users from the remote

What happens if Developer2 terminal make changes to the ‘hello.txt’ without updating the local master



Here the Developer2 pushed feature-dev2 hence there was no rejection at the remote, however when the feature branch is tried to merge with remote master you may get conflict because the feature branch is not having other user work.



Here we can’t merge hence we can create pull request & close it so that the Developer2 will manually resolve, he must follow below steps

1. Pull the remote master to local master
2. Checkout to feature branch
3. Merge local master with feature branch if conflict resolve

Steps:

>> git checkout master

>> git pull

>> git checkout feature-dev2

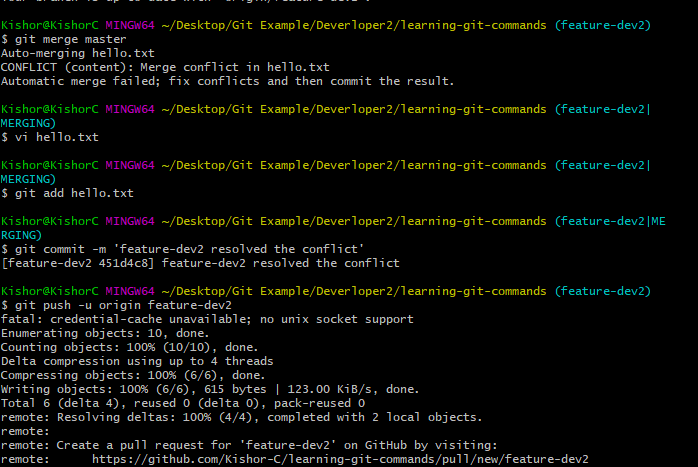
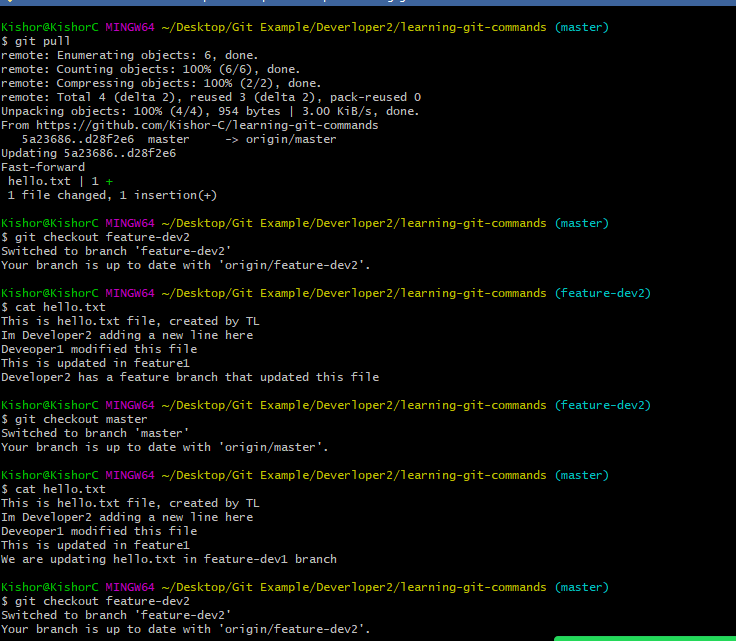
>> git merge master

# resolve the conflict when you get auto-merge failed

>> git add hello.txt

>> git commit -m ‘some message’

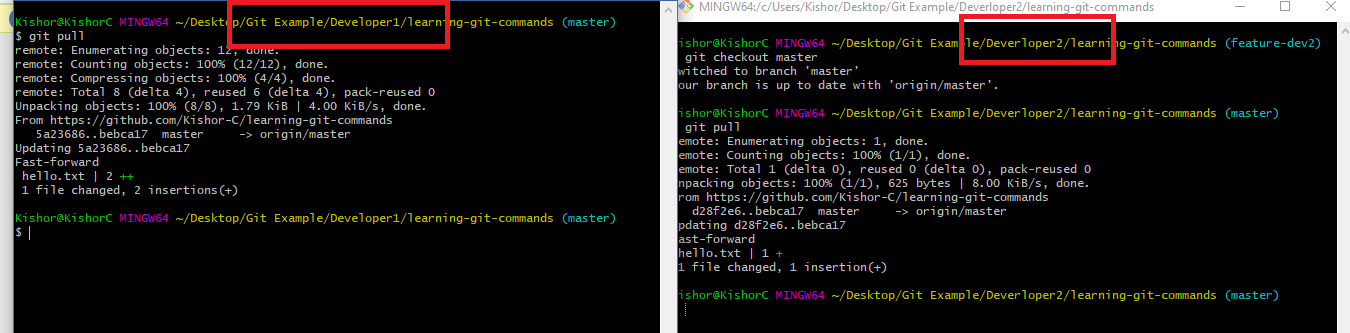
>> git push -u origin feature-dev2



Now in Remote there wouldn’t be any conflict to merge this feature branch

Final step:

Pull remote in local master in both the developers terminal to keep up to date.



Summary of Git steps to be followed when working in the project

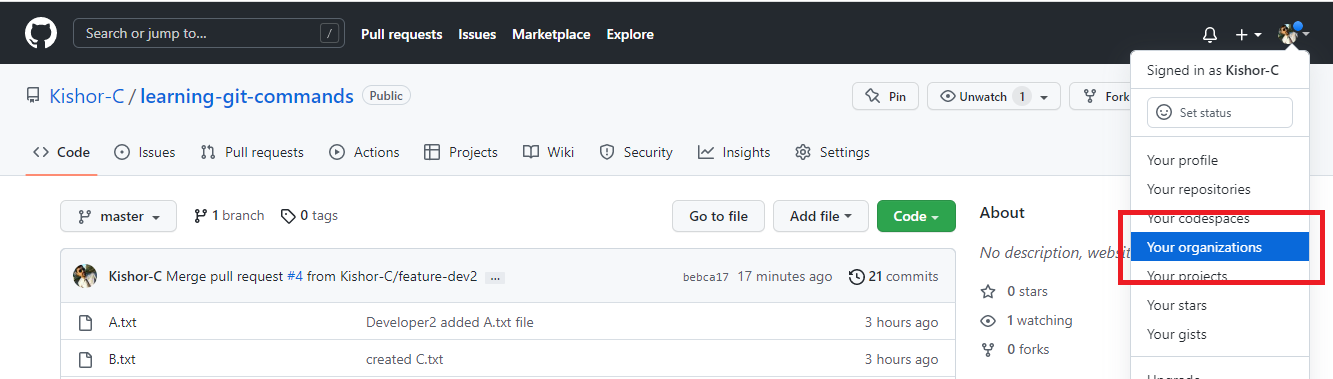
1. Git clone the remote repository if you don’t have local repository
2. Always make sure that you don’t work in master/main branch
3. Create feature branch to make any changes
4. Push the feature branch to the remote
5. If Pull request failed to merge then update the local master with remote master first using ‘git pull’
6. Once git pull updates the local master, checkout to feature branch and merge local master branch with feature branch using ‘git merge master’
7. You may get conflict while merging hence you can resolve it by editing the file having conflict.
8. Once conflict resolved commit & push the feature branch to the Remote repository for merge, but before that you can use git pull to ensure you have up to date changes

Git Organization:

It is a feature in the GIT which makes people to work in a team to collaborate their work, you need to add members here so that only members in the organization can work with the repository

Here someone must add the members by mentioning the mail-ids of their Git account so that each member will accept the invitation over the mail

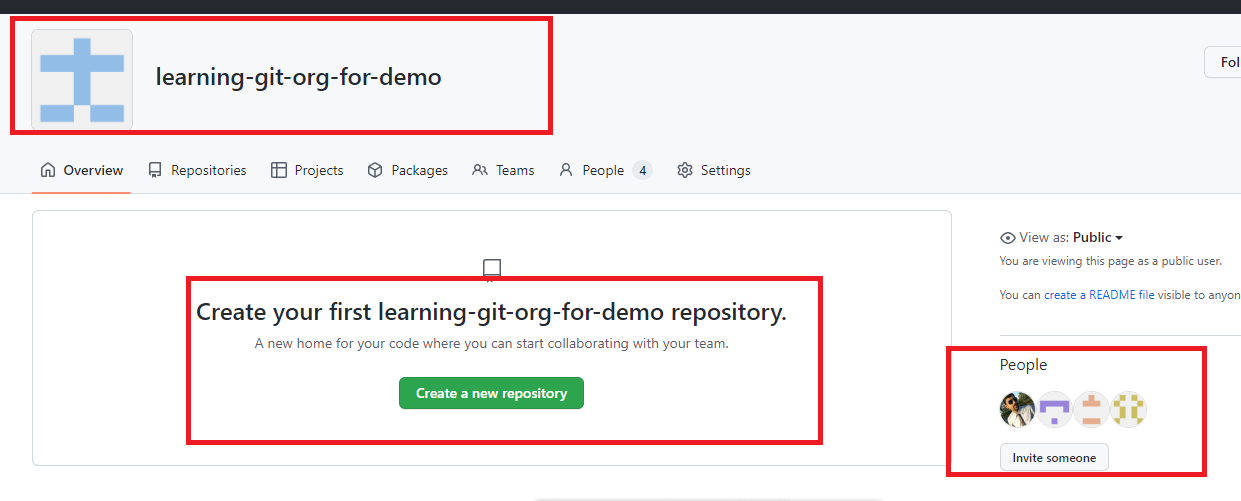
How to create Organization



Steps

1. You can create a new organization & add members.
2. You can pick Free plan
3. Enter organization name & your git mail id to create organization
4. Add members by entering mail ids
5. Members need to accept invitation
6. You can make all the members owner to enable them do push/pull task

You must able to see members in the organization



Activity

1. Try out all the GIT commands taught in the Session
2. Create an organization (any one person in the team), add the members, one person in the team must create the Repository & all the members must clone it
3. Each member can create a text file with their name in the feature branch & push that feature branch to the organization repository
4. Someone in the team must take care of merging the pull request

Java

Software required

* Eclipse IDE / STS

Web Development

There are 3 main technologies we use to develop web pages

1. HTML
2. CSS
3. Javascript

HTML stands for Hyper Text Markup Language, it is mainly used to create contents

CSS stands for Cascading Style Sheet, it is mainly used to add styles to the HTML

Javascript makes web page more dynamic by manipulating HTML & CSS at runtime

HTML uses elements or tags to create contents, it as many predefined elements that browser can understand some of them are

<p>, <span>, <div>, <table>, <img>, <form>, <h1>, <h2>, <h3>, <h4>, <h5>, <h6>, <ol>, <ul>, <li>

Software required for HTML, CSS, & Javascript

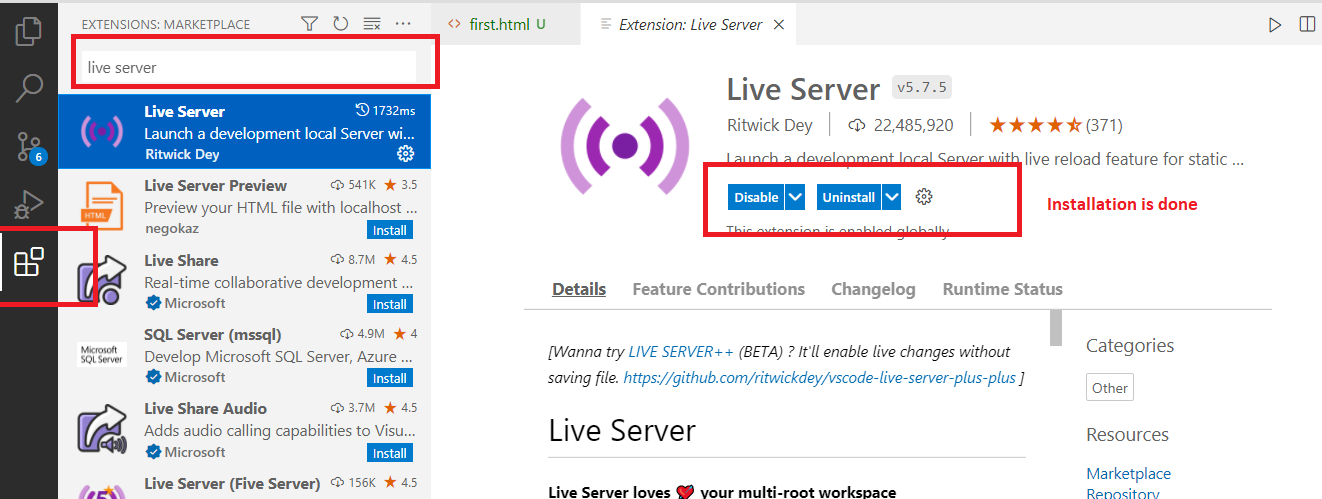
1. Editors: Notepad, VSCode, Brackets, Online editors(Code pen, vscode.dev)

You can get online VSCode editor from <https://vscode.dev/>

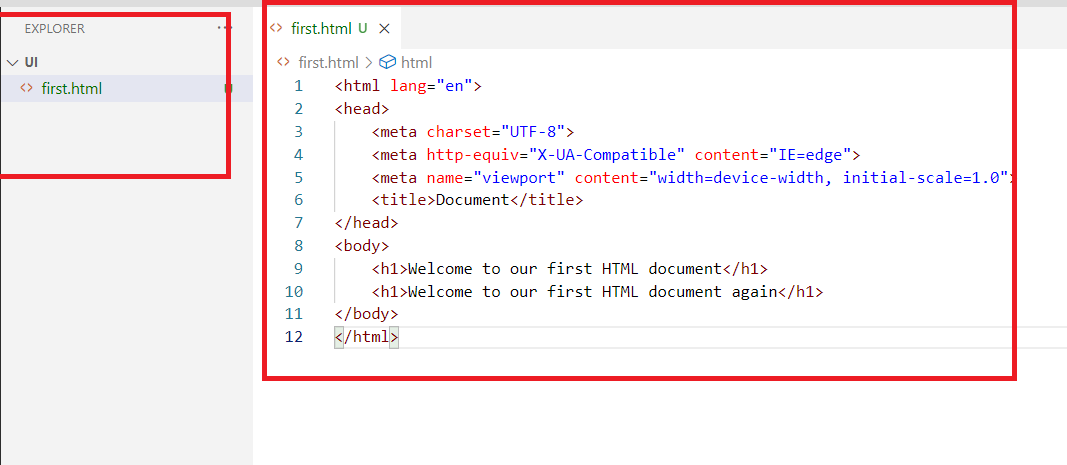
1. Browser to preview the output

VSCode: It is available both online as well offline, but in offline you can add plugins to the VSCode to get a better development experience

ex: Live Server which auto-reloads the browser when you modify the file

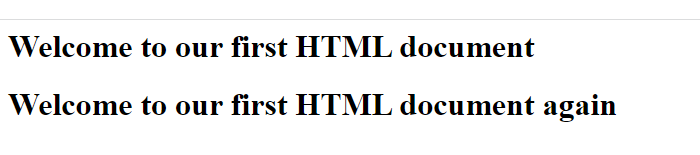


first.html



After you have installed Live Server extension, you can right click on the file and open with Live Server to see the output

Output:

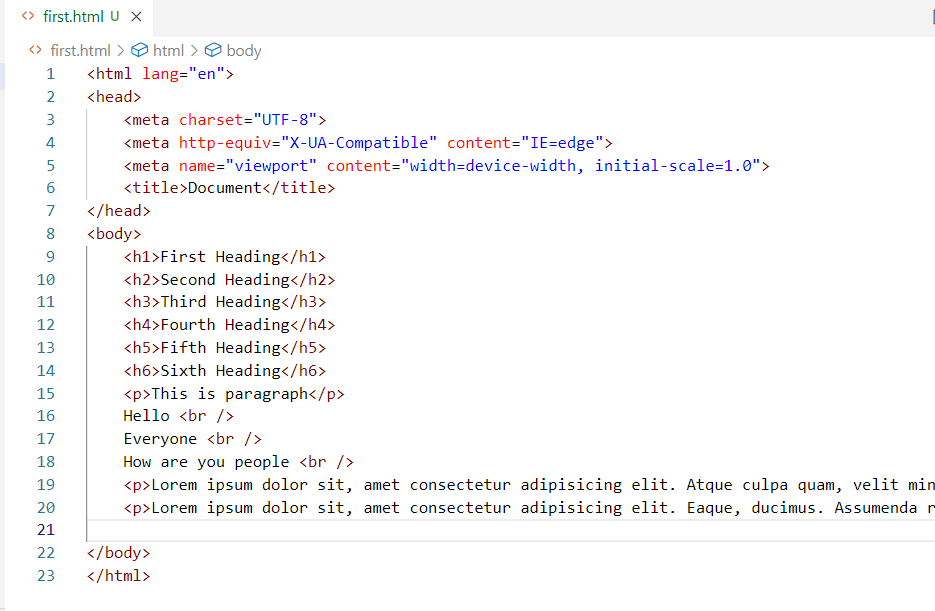


HTML has 6 heading tags

Their font size varies according to the heading tag number

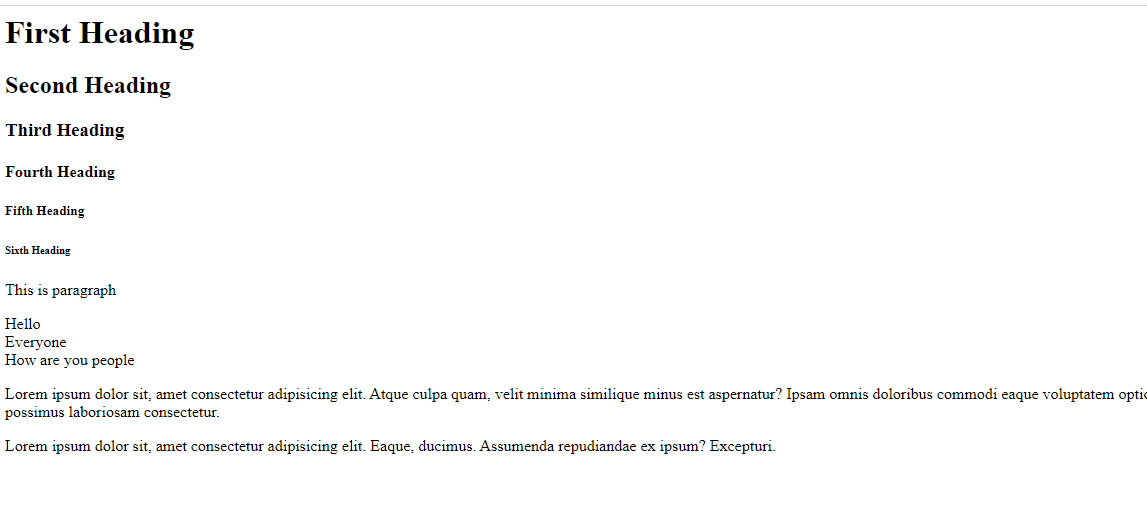
h1 will have a bigger font, h2 will be slightly lower than h1, same way you have h3, h4, h5, h6

first.html



You can type lorem10 to generate 10 words of Lorem ipsum

Output:



HTML Images & their attributes

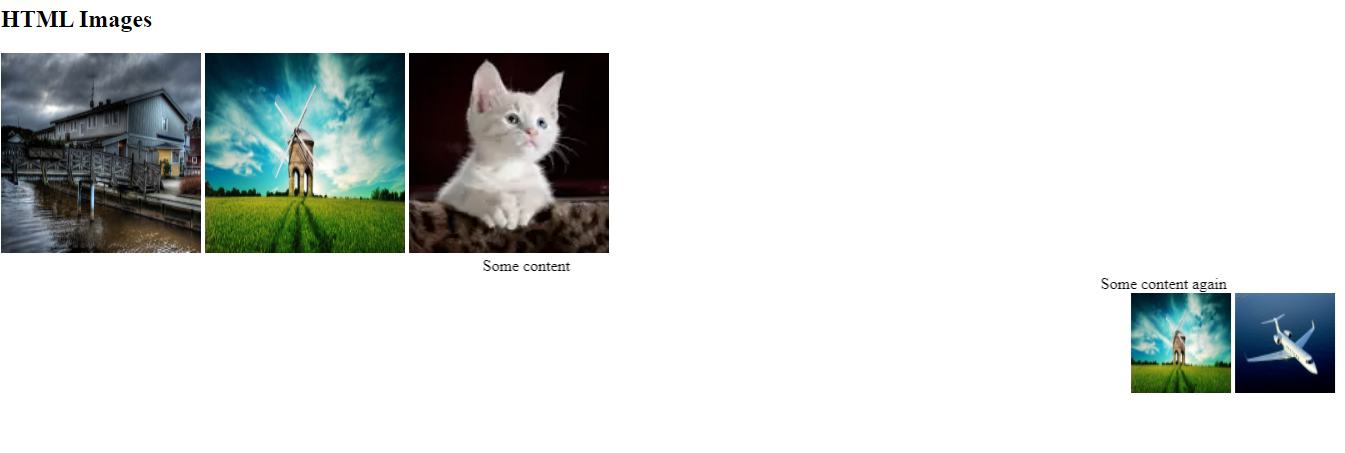
<img> tag is used to add images, we can add the image present in our local machine or we also add online image url and load in the web page.

It has attributes like src, width, height, alt



div: It is a container tag used to include other tags inside it, this is one of the very useful tag in HTML

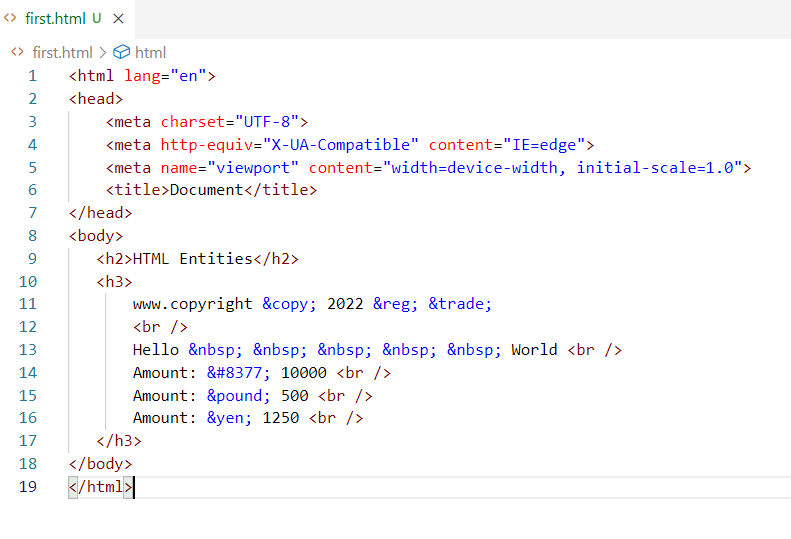
Output:



Entities

These are some special characters that will have a name or a number so that they can displayed, they may be not be present on the keyboard, they are used with &entityName; or #&entityNumber;

Ex: © ® ™ All these are some special characters that can be displayed using the entity names like &copy; creates ©, &reg; creates ®, &trade; creates ™, similarly we have entity numbers for currency symbols like pound, rupee, yen



Output:



Comments in HTML will be



This same comments works even in XML File also, they are multi line comments

HTML Lists

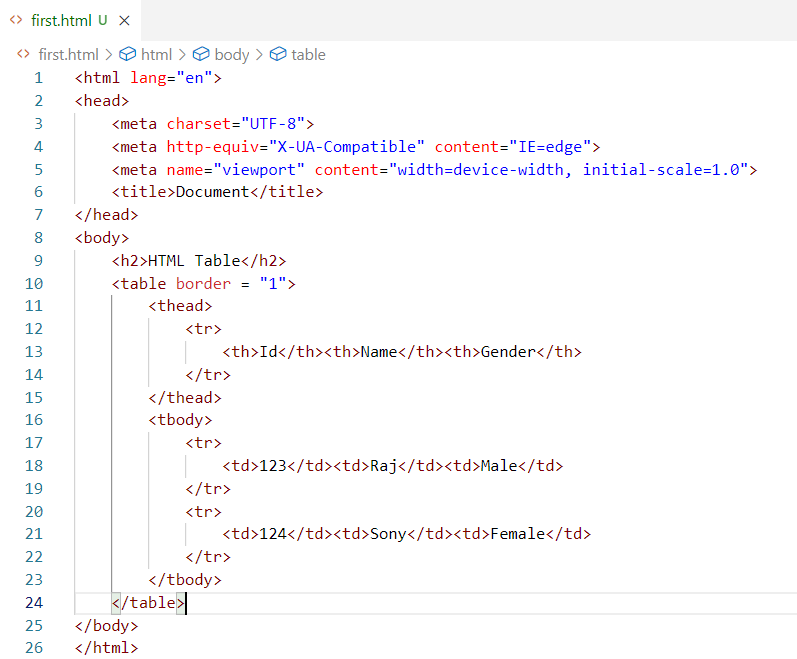
In HTML we have tags to show the contents in an ordered & unordered way, for that we have <ol> & <ul> tags both takes <li> as a child tag to show the list of items



In <ol> you can use type as 1, A, i, and in <ul> you can use type as disc, square, circle.

HTML Tables

This helps to create tables that will have rows & columns, <table> is used to create table & <tr> is used to create rows, in the row we can have either <th> or <td> to create the data for the columns, <table border = “1”> will give some border lines in the table, maximum number you can give is 8



Git Hub Activity Link

<https://github.com/Kishor-C/publicis-sapient-student-activities.git>

Steps to perform

1. Fork the remote repository so that it is cloned in your account remotely
2. Clone the remote repository of your account to your local machine (Ensure you have not cloned the below URL

<https://github.com/Kishor-C/publicis-sapient-student-activities.git>

1. Create day wise folder on each day and create separate files to perform the hands-on exercises
2. Push the changes to your remote repository

HTML Forms

Forms help user to enter inputs, select controls like radio, checkbox, dropdown and so on

<form> tag is used to create the form, it provides 2 attributes action & method

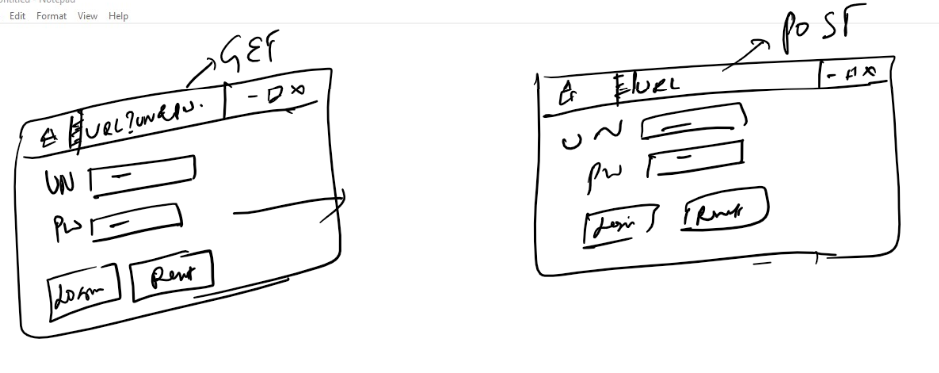
<form action=”url” method=”httpMethod”>

Form action is the backend resource url the form will submit the input,

Form method is http method like GET or POST which is used to wrap the data in the HTTP request either in the body or url of the request.

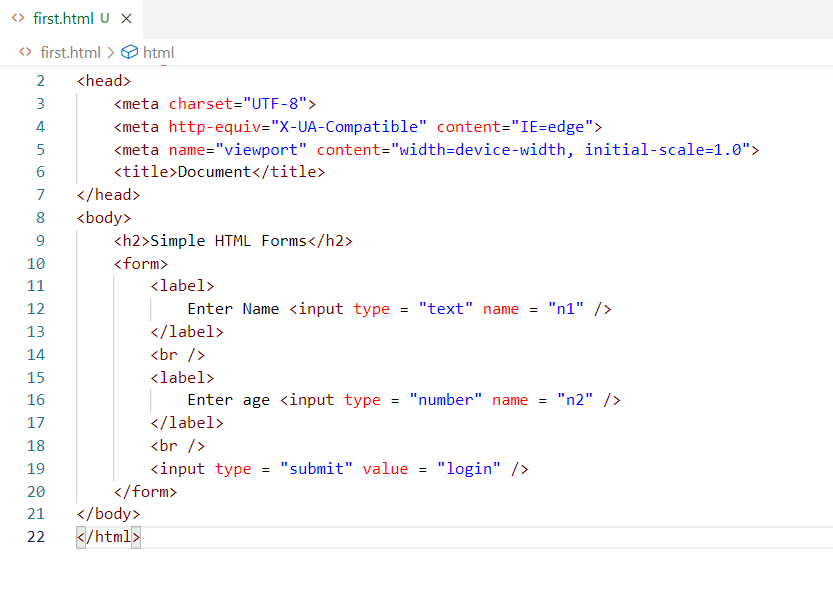
GET: It is the default method the form uses, here the data is submitted via URL, it means the data will be visible to the end user

POST: It sends the data via the request body & it wouldn’t be visible to the end user

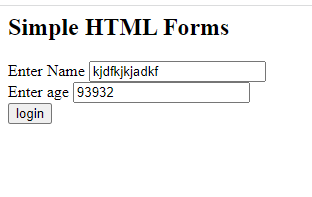


GET vs POST

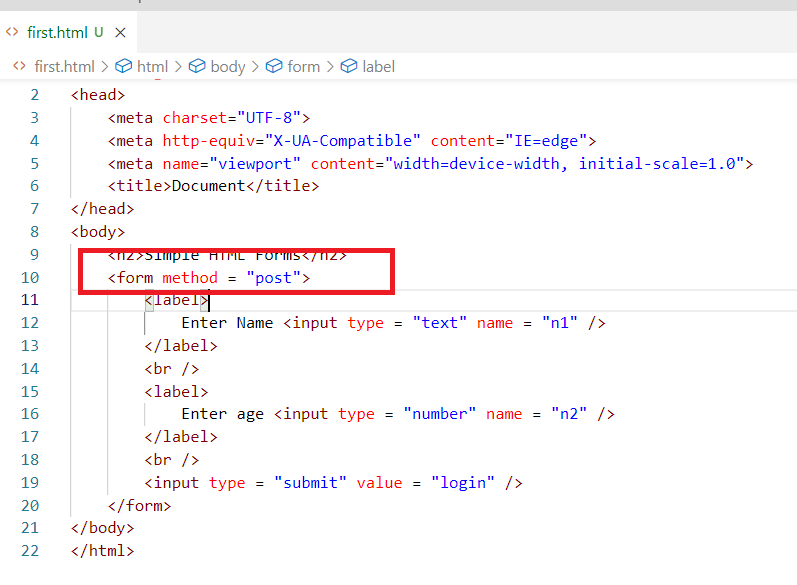
|  |  |
| --- | --- |
| GET | POST |
| GET sends data via URL | POST sends data via Request body |
| Insecure | Secured |
| Faster | Slower compare to GET |
| Supports only 256 characters | Supports unlimited characters |
| Default form submission is GET | It is not chosen by default, we need to use <form method = “POST”> to make form submission post |



Output:



You can use method = “Post”



Here you don’t see the data in the URL.

There are other controls you can create in forms like radio, checkbox, dropdown, date and so on.

For radio button you can use <input type = ‘radio’>,

For checkbox you can use <input type = ‘checkbox’>

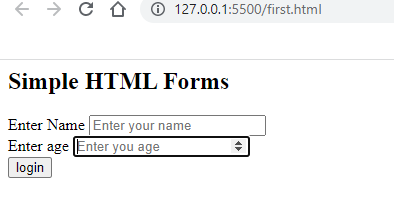
Form Control attributes

autocomplete: You can use this when the input box shouldn’t show the history or previous input, you can make it false.

placeholder: You can use this when you want to display the hint in the input box what to enter.



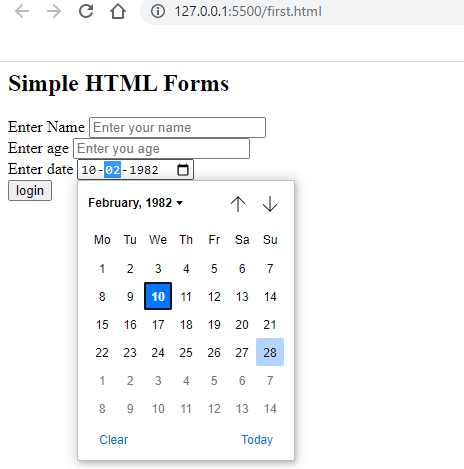
Output:



date: It is a type in the <input> which shows the calendar



Output:



input name: These are placeholder which make server side programs to read the values

Example in Java we have Servlets that can handle these input names in a request parameter as below



CSS: Cascading Style Sheet

It is used to style the HTML elements, it uses a syntax which will have css property & value to it

You can embed CSS into HTML in 3 ways

1. Inline CSS
2. Internal CSS
3. External CSS

Inline CSS: Adding styles to the HTML elements directly so that it applies only to that particular element, it can be added using ‘style’ attribute

<p style = ‘color:red’>Some content in p element </p>  
<p>some content again in p element </p>

Internal CSS: Adding styles to the entire HTML documents, the styles can be used by multiple elements in the same HTML document

<head>  
 <style>  
 p { color : red } /\* all the <p> gets red color\*/  
 h1 { color : blue } /\*all the <h1> tag gets blue color\*/  
 </style>  
</head>

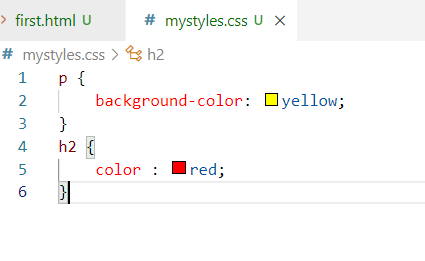
<body>  
 <p>…</p>,  
</body>

External CSS: It is a style that exists outside the html, it can be applied to multiple HTML documents, it should be created in a css file & you must link the CSS using <link> tag

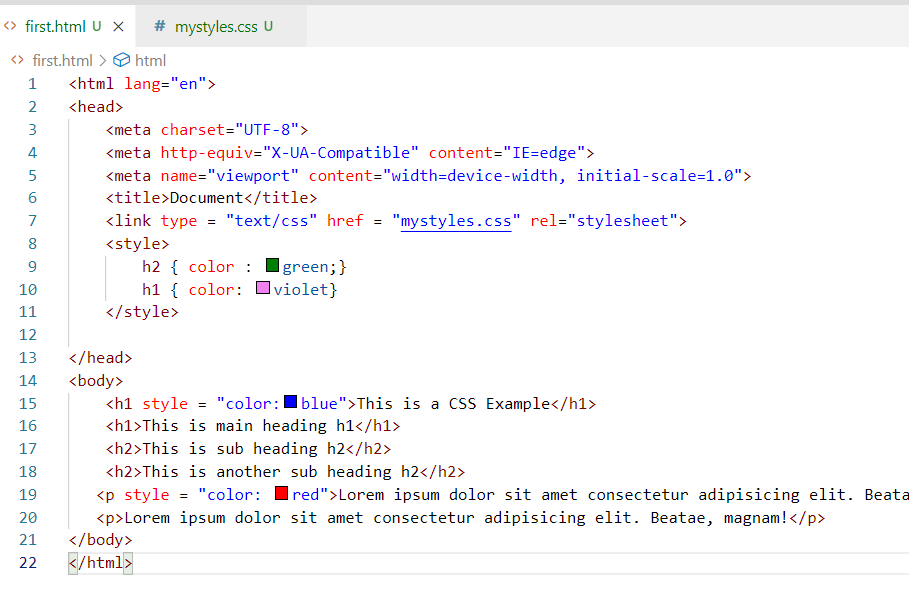
What happens if the same element is styles with External, Internal & Inline

Internal overrides External, and Inline overrides Internal

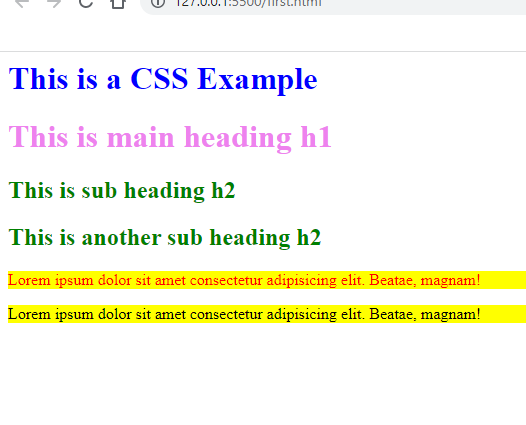
mystyles.css



first.html



Here external css is overridden by internal & internal is overridden by inline css



There are many CSS properties which makes HTML elements look better, but as a CSS developer we need to know the syntax of how to use CSS instead of remembering the CSS properties.

Different types of selectors

Multiple elements selector: This is used to add css to the multiple elements,

ex: h1, h2, h3 { property : value } /\* h1, h2, h3 gets same CSS\*/

ex: \* { property : value } /\* this to select all the elements \*/

Tag selector: This is to select a particular tag

ex: p { property : value }

Class selector: This is to select multiple elements with the same class

ex: div.c1 { property: value}

Here <div class = “c1”> get the style, we can have one or more div with the same class and also we can apply classes to multiple html elements

ex: .c2 { property : value }

Here <p class = “c2”> <div class = “c2”> can get the css of c2

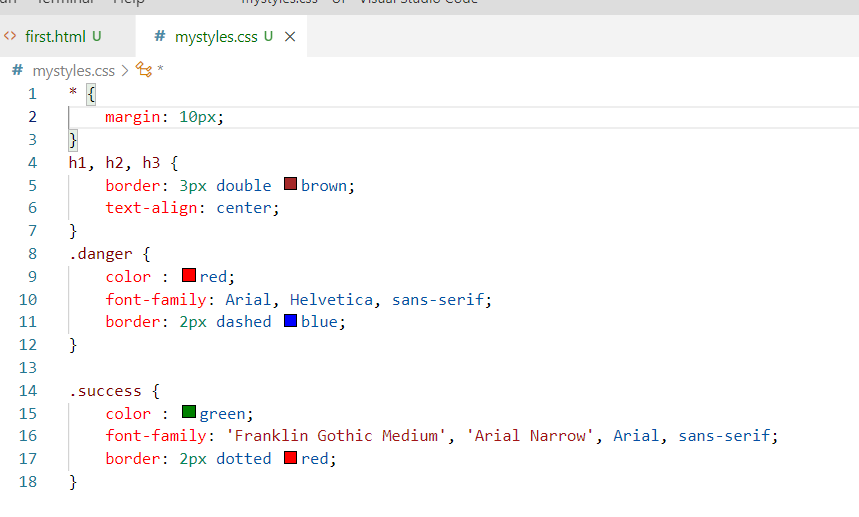
Id selector: This is used when you have an unique element with an unique id

#a { property : value}

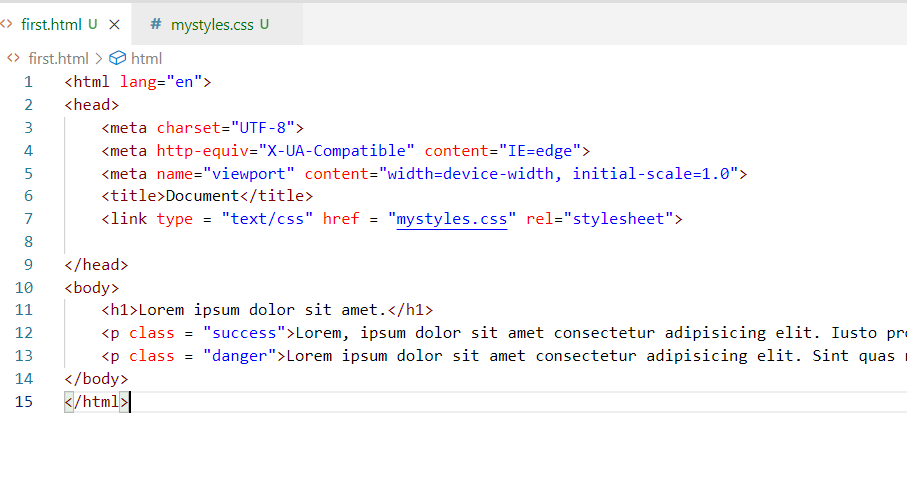
Here an element can have id a i.e., <p id = “a”>

Note: Id must be always unique, though it works in CSS, it doesn’t work in Javascript

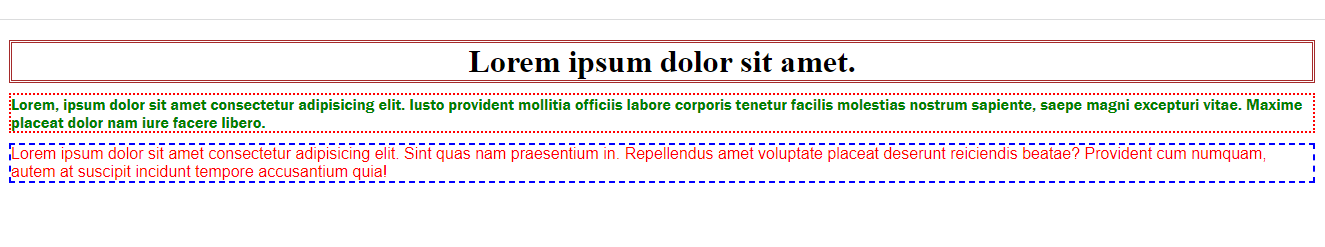
mystyles.css



first.html



Output:

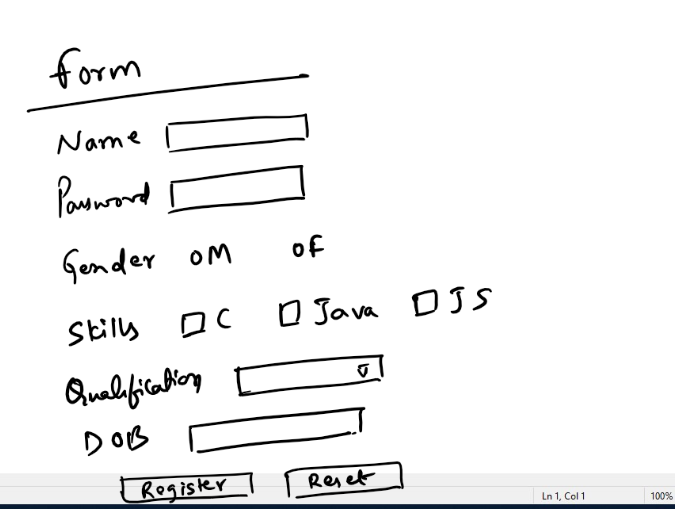


Activity:

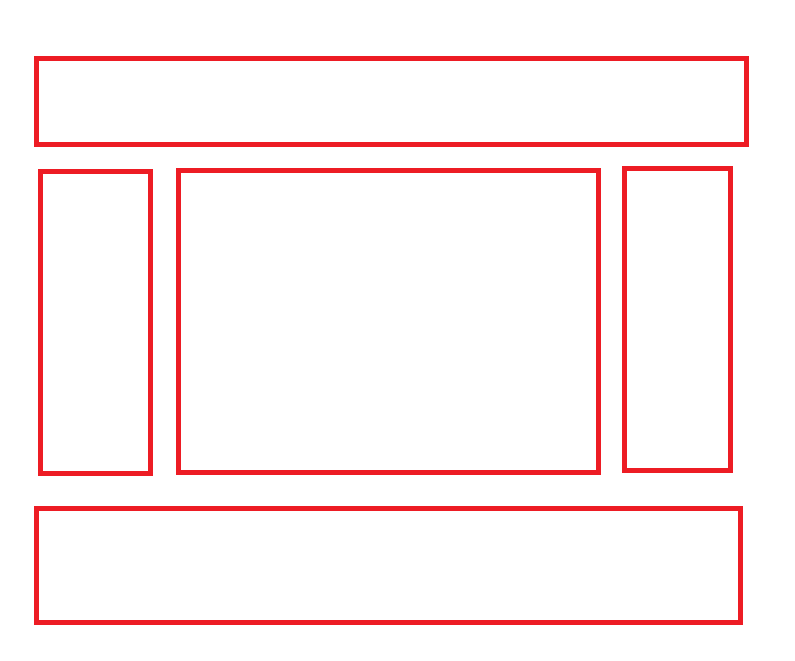
1. Create separate files for each examples & update in the Git repository that was forked

ex: day6/ex1.html, day6/ex2.html and so on (or) you can give meaningful names to the html files

1. Create a form that will have all the controls like text, password, radio, checkbox, dropdown, use <table> without border so that all the labels & controls will have proper alignment



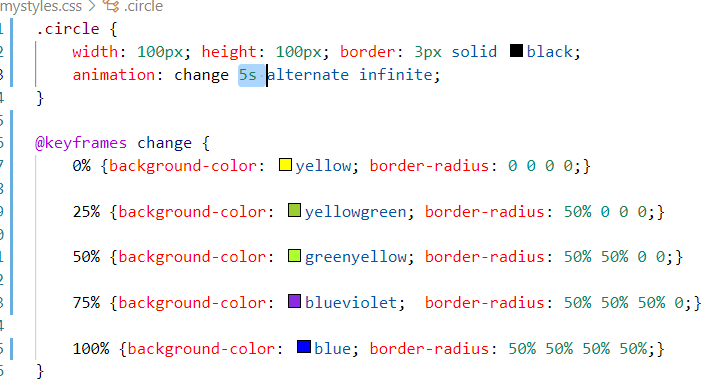
1. Using CSS try to create a web page with a layout having only borders instead of contents, use <div> as an element to create the layout, the layout must look as below



Animations & Transitions:

They add smooth effects to the HTML elements while applying animations to the HTML elements, we need to use @keyframes for animations

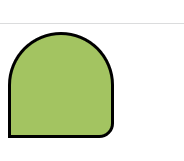
CSS



HTML



Output:



Few topics which are important in CSS

* Flex
* Media Query
* Grid & Containers

Javascript

Javascript is a scripting language which helps you to add dynamic content in the web page, it can

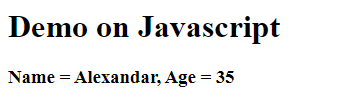
* Access HTML at runtime
* Access CSS at runtime
* Perform Client side validations
* Work on DOM to modify HTML & CSS
* Handle events generated when user interacts with the web page
* Also provide a way to implement backend services using node.js, i.e., you can run Javascript’s at the backend without browser with the help of node.js

HTML



HTML can use <script> to include Javascript

Output:



Fundamentals

Operators

Conditions & Loops

Variables

Functions

Inbuilt Objects

Operators: These help to perform operations in the javascript, there are many operators Javascript supports which most of the programming language supports

+, -, \*, /, <, >, <=, >=, ++, --, =, ==, ===, !=, &&, ||, \*\*

Conditions & Loops: These are used to apply conditions to perform some operations and also you can use loops to execute some statements until the condition is satisfied.

Conditions: if, if else, if else if … else, switch

Loops: for, while, do while

These conditions & loops have same syntax & works the same way like other programming languages like C, C++, Java, C#

Variables:

These can store simple data to complex data like arrays & objects

In Javascript you can create variables with 3 keywords

var, let, const

Note: var is an older approach to declare variables we need to use either let or const

Functions:

These are named block of code that will have logics which can be called from anywhere, it can be reused

Syntax:

function functionName() { …… }

Inbuilt objects

These are provided by browser to perform some actions, the objects will have properties & functions, some of the inbuilt objects are:

1. document
2. window
3. console
4. sessionStorage
5. localStorage

var, let & const

These are used to create variables, however var is avoided now a days, because it creates variable in global scope.



When you declare a variable with var, it will be always created as a global object part of window, even if we declare within the scope it is not going to be accessible within that scope, it will be global.

Output:

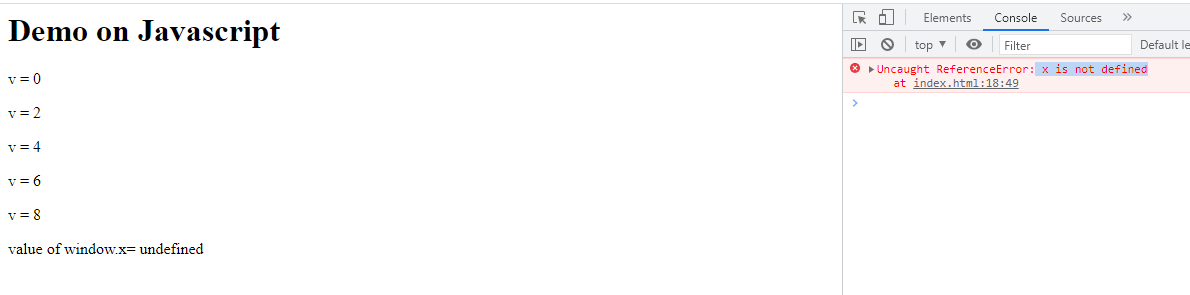


let & const

They help you to create variables within the scope so that they are not accessible outside the scope, let is modifiable, whereas const is for constants you can’t modify



Output:

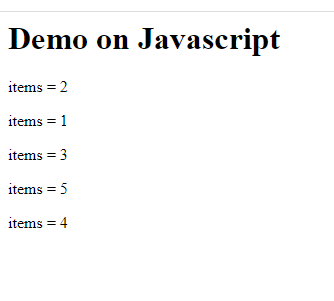


Arrays: It is a container that can store multiple values in a single variable, you can use for loop to iterate the array

let items = [2, 1, 4, 5];



Output:



Objects in Javascript

Objects are real world entities which will have properties & behaviours i.e., data & function, in Javascript you can create objects in many ways

1. literal style: let user = {name: “Alexandar”, age: 35}
2. function style

function User(name, age) { ….}

Here you pass name & age to initialize properties name & age of the user



The literal style is used by user1 & user2 is using functional style object creation, literal style is used when you want to initialize multiple objects in an easier way, functional style is used when you want to apply some logics while initializing.

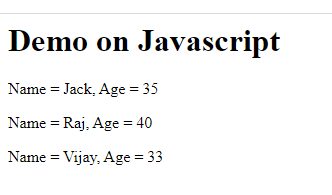
Creating arrays of objects

let items = [{…}, {…}, {…},….]

Here the object will be wrapped in the [], you can use for loop to iterate each object



Output:



Events:

What are events in Javascript?

These are the things that happen in the web page, you can handle this event and perform any action you want based on the events

Note: HTML elements generate events

Below are some events

* onclick
* onsubmit
* onmouseover
* onmouseout
* onkeyup
* onkeydown



Here the ‘event’ is an inbuilt property which refers to an event object that is generated and the handleClick can perform the job updating the div#i2

innerHTML: It is a property present in the element to add any HTML elements



Here <b> is used to bold the font.

Output:



Things we do by accessing the element

When we access the element either by id or by event.target we get an object of Javascript which is a Node, it has many properties & functions like

innerHTML, style, and other properties which are the attributes of the element.



Here ele is a node which has a style property and style.color adds color property to the style attribute of the element.

Output:



Note: Colors can also be applied using rgb() pattern

i.e., rgb(255,0,0) will give red color,

rgb(0, 255, 0) will give green color

These help in creating range colors



You also generate variety of colors by using range bars so that they can be scrolled from 0 to 255 and their values can be assigned to the colors.



Here the range bar generate values when you move and their values can be utilized in rgb() however here we are only showing them in the browser.

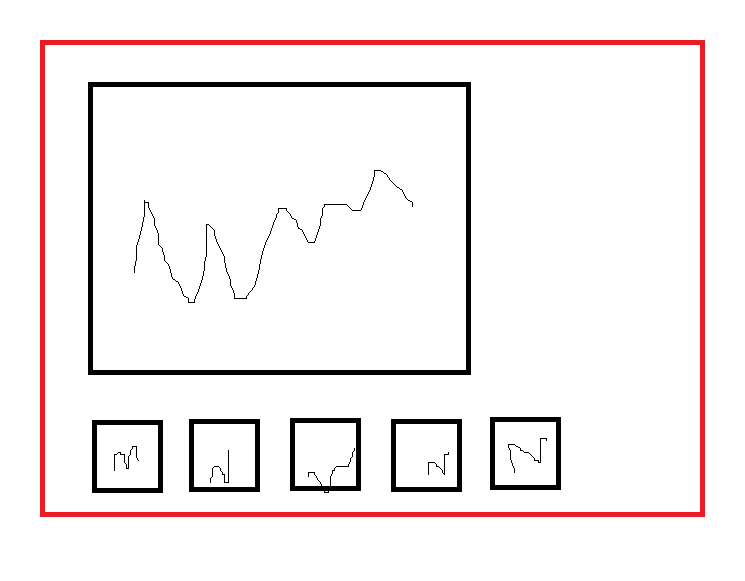
Output:



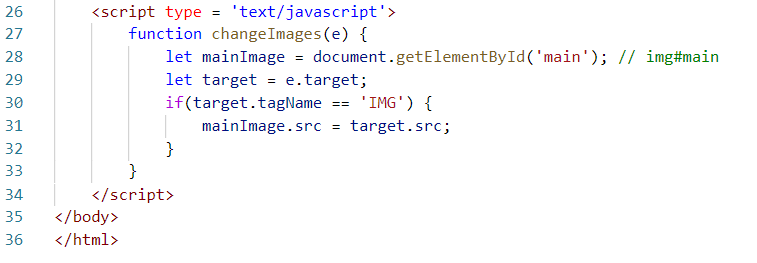
Javascript makes HTML & CSS to be accessed at runtime so that it changes certain content dynamically.

We can write a program to replace the images to one common place when you click on that image.

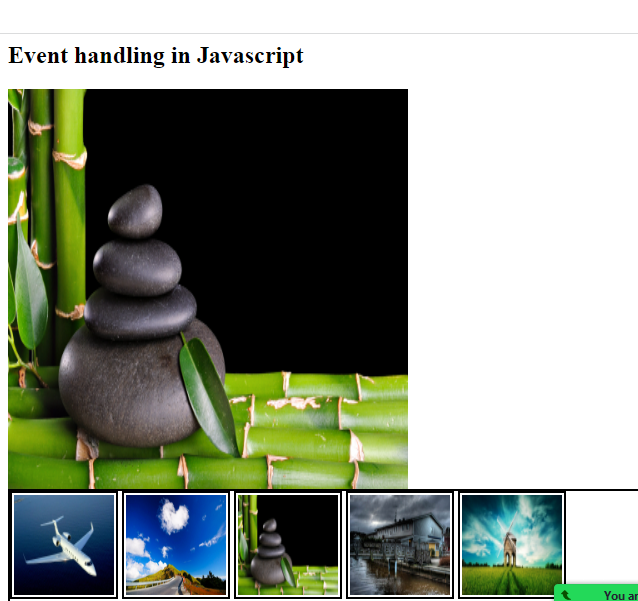
Requirement



Solution:



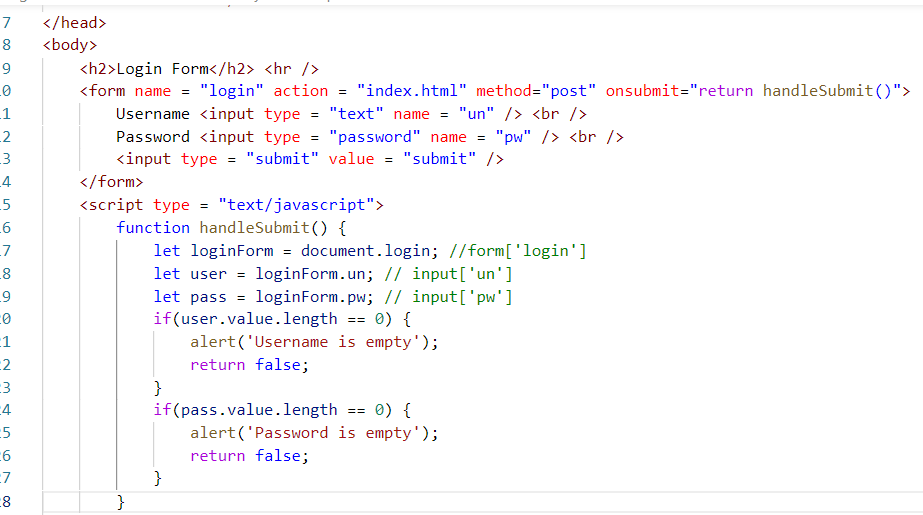
Output:



Form Validations

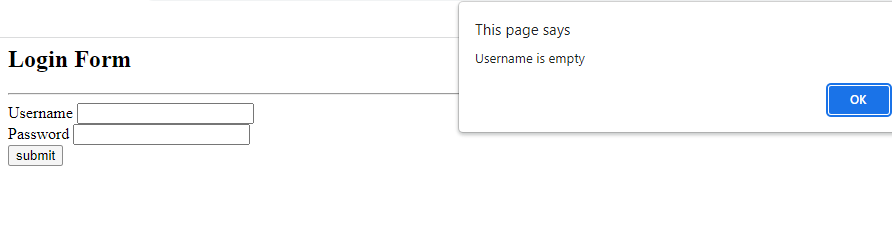
You can handle form data in Javascript and can only submit if the data seems valid

Note: Since we don’t have any server side programs to receive the data once the form is valid to submit, we can use a dummy HTML file to show when the form is valid



Note: Here index.html is loaded only when the form input is valid, index.html is just a dummy file to show when the form is valid, in real time you will have a server side program to receive the data & return the response when the form is valid

Output:



Canvas element:

It is used to create shapes using Javascript

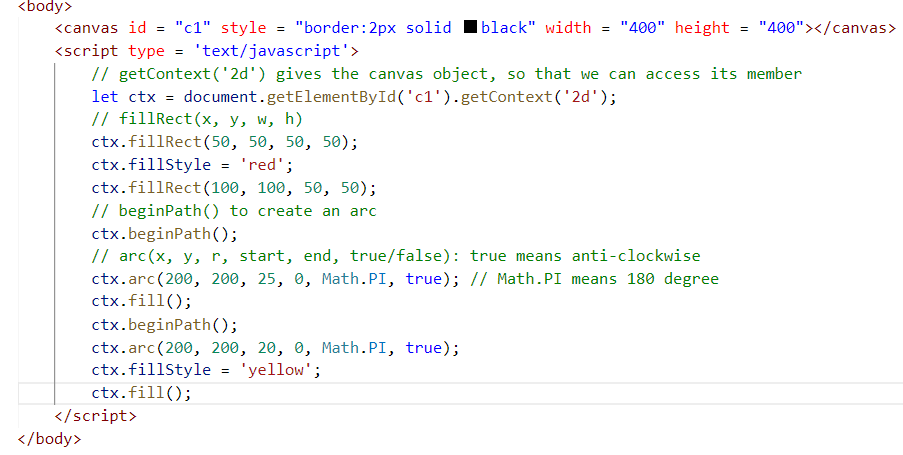
<canvas id = ‘c1’>

You can use a canvas related object to create shapes like circle, square and so on.

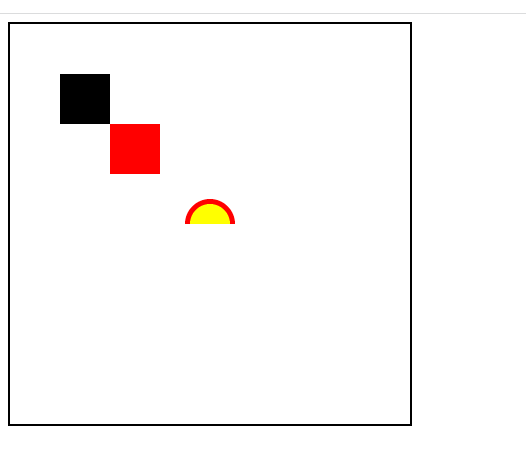
You need to use getContext(‘2d’) on the canvas element to use the canvas object properties i.e.,

let ctx = document.getElementById(‘c1’).getContext(‘2d’);

From ctx you can access all the canvas object properties & functions.



Output:

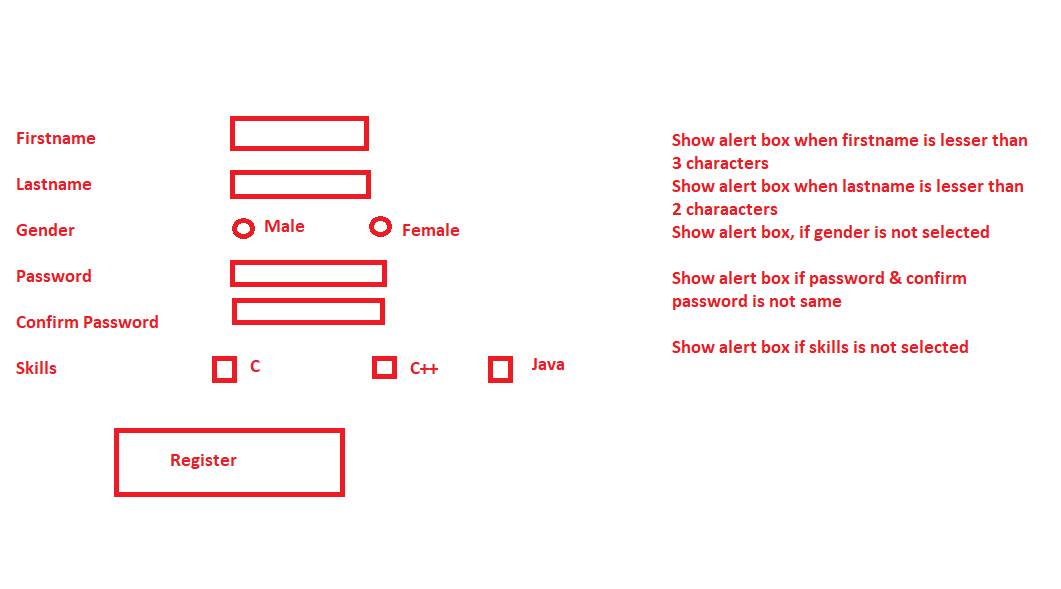


Activity 1:

Try all the above examples atleast 5 examples you can add to the GIT in separate files in day wise folder

Activity 2:

Validate a Form that will have form controls like input.text, input.password, input.radio, input.checkbox, and so on.



Note: Gender any one must be selected, it’s a radio button, skills you can have one or more selection it is a checkbox, if none of the skill is selected show the alert box,

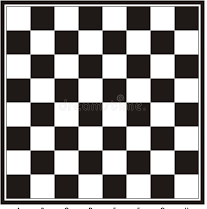
Password & confirm password must be same else show the alert box

Once done you can upload in the GIT in a separate file

Activity 3:

Using canvas element create a chess board which will have 8 \* 8 box with white & black,

Note: You need to use nested for loop



Activity 4:

Using canvas create arcs that looks like rainbow

Note: You must not hard code the arc creation instead by calling it 7 times to create Rainbow, instead use loops & arrays to create the arcs so that it calls arc() function only once but creates 7 arcs that looks like rainbow

Push all the activities in the GIT in separate files

Javascript Front end technologies

1. Angular Framework
2. React.js

Angular Framework:

It is a framework which is used to develop Single Page Applications where only part of the page is refreshed, it is Created by Google

Types of applications you can develop with angular

1. Web applications
2. Mobile applications

Angular helps you to create single page applications using components, these components are reusable UI’s (User Interfaces) which can be developed & maintained independently.

Components: These are the User Interfaces which you see in the page, it can be reused in other component i.e., it can be nested with other components

React.js

It is a Javascript library which is used to develop Single Page Applications where only part of the page is refreshed, it is Created by Facebook

Framework: It provides everything to create applications like all the industry best practices, but we are forced to use what framework provides, to remove the features of framework is a bit difficult task.

Framework provides all set of libraries to develop complete application

Library: It provides only the limited functionalities present in the library for your applications, you need to add more libraries to add more features to your application, it doesn’t follow any best practices, it will not have everything to build the application and we need to add the libraries based on what we need

Angular vs React.js

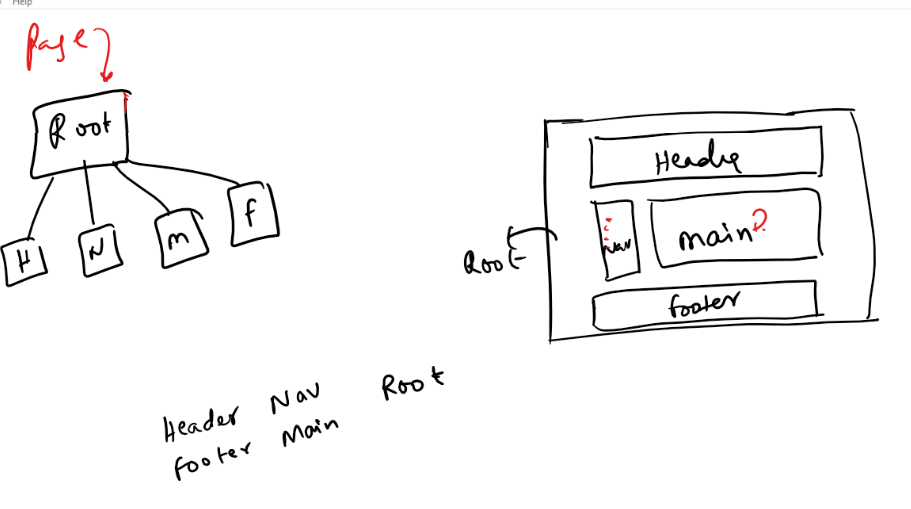
|  |  |
| --- | --- |
| Angular | React.js |
| Angular is a framework | React.js is a library |
| Provides all the features to develop the applications | Doesn’t provide all the features to develop the application |
| In Angular we have an inbuilt class HttpClient that can access backend service | In React we don’t have any inbuilt class or library but we can download a library called ‘axios’ to access backend service |
| In Angular we have inbuilt facilities to create routers to dynamically load components | In React.js we don’t have that facility but we need download a library called react-router-dom |
| Provides all the tools & commands to increase the speed of the development | Don’t provide any tools or commands, but it is easy to understand compare to Angular |
| Google is the founder | Facebook is the founder |
| Provides angular cli toolkit to develop application, its mandatory to have this toolkit | No toolkits are mandatory, you can develop applications in react with a few javascript files & html files |
| Angular uses Typescript & HTML to develop application | React.js uses JSX & HTML to develop application |

Typescript: It is a super set of Javascript which makes Javascript more consistent with the types, however browser doesn’t understand typescript, angular uses typescript compiler to convert typescript to javascript

JSX: It stands for Javascript XML which simplifies writing HTML code in the Javascript, it is an extended version of Javascript where browser doesn’t understand, hence React.js uses Babel to convert JSX to Javascript

How the Single page applications are created

These are created with components and nesting them with other components, at runtime a particular component will be refreshed instead of all the components, to get this feature we need entire page to be represented by one component called root component and all the other components must be part of the root component



Here you will have only one page which will have a root component and all other components are part of the root components and you must able to only load the particular component not the entire root components

Whether we work with Angular or React.js we need Node.js installed which gives us the way to download the toolkits for both Angular & React.js (But for React.js you don’t need toolkit you just need links to the react library)

JVM – Runtime environment for Java

Node.js – Runtime environment for Javascript

Usually Runtime environment provides lot of modules which helps to develop applications, since Browser doesn’t provide many modules and its limited for only front end development, Node.js provides many modules which can be even downloaded from the internet that allows you to develop event backend applications

To install node.js

<https://nodejs.org/en/download/>

Choose the version which is recommended for most users

Once you install you can open terminal & type node -v to see node version this confirms the installation of node.js

Node.js: Through node.js we can install any external javascript tools or libraries like angular toolkit, react toolkit, we will first install angular toolkit

Firstly we can just check the angular toolkit is available or not with ‘ng –version’ command

>> ng --version

ng will confirm that angular toolkit is available, it displays version number, if in case it shows ng command not found then we need to install ng i.e., angular toolkit.

How to install Angular CLI (Command line interface)

Node.js provides npm (node package manager) to install any online libraries or toolkits, we need to use `npm install -g @angular/cli` (or) `npm install @angular/cli`

but -g is preferred because angular is installed globally so that you can use ng from any location in the command prompt.

Software required

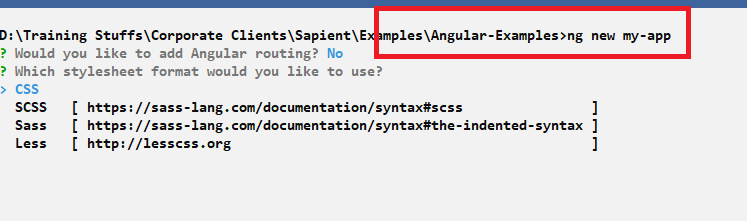
* node.js
* angular cli: `npm install -g @angular/cli` if its linux/mac you can use `sudo npm install -g @angular/cli`
* vscode

Creating angular project

Pre-requisites

* You need to have angular/cli installed
* You need to have vscode installed

‘ng new app-name’: It is the command used to create a new project, once you create project you can navigate to the project & run ng serve to launch the application

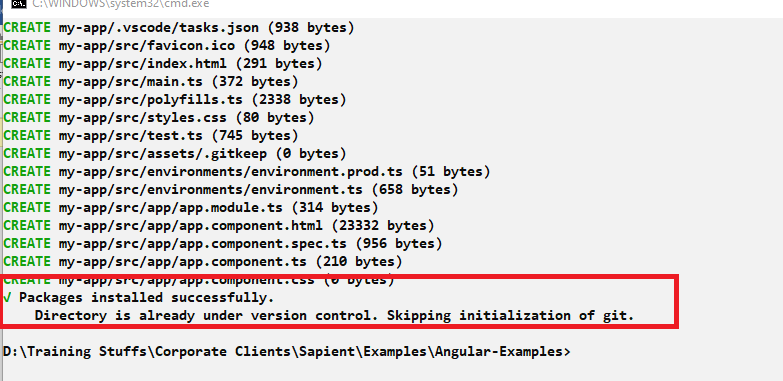


When you try to create new project it asks routing feature you can enter ‘N’ and asks the stylesheet format you can use arrow key and choose CSS

Note: It downloads some dependencies from the internet

What happens when you create a new project

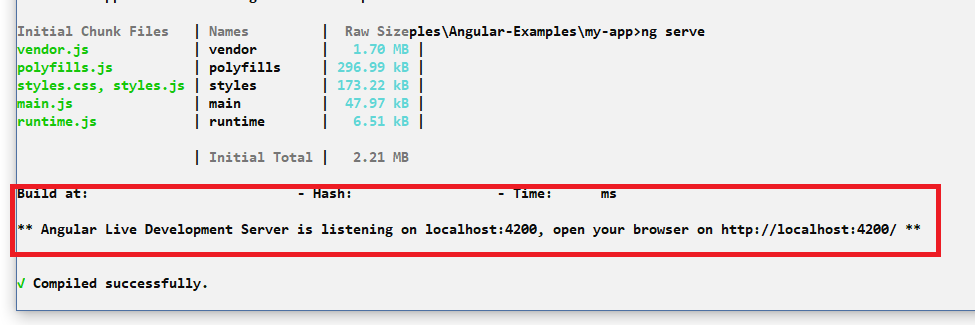
It creates a project folder with the name my-app, you need to navigate to my-app using cd my-app and run ng serve command

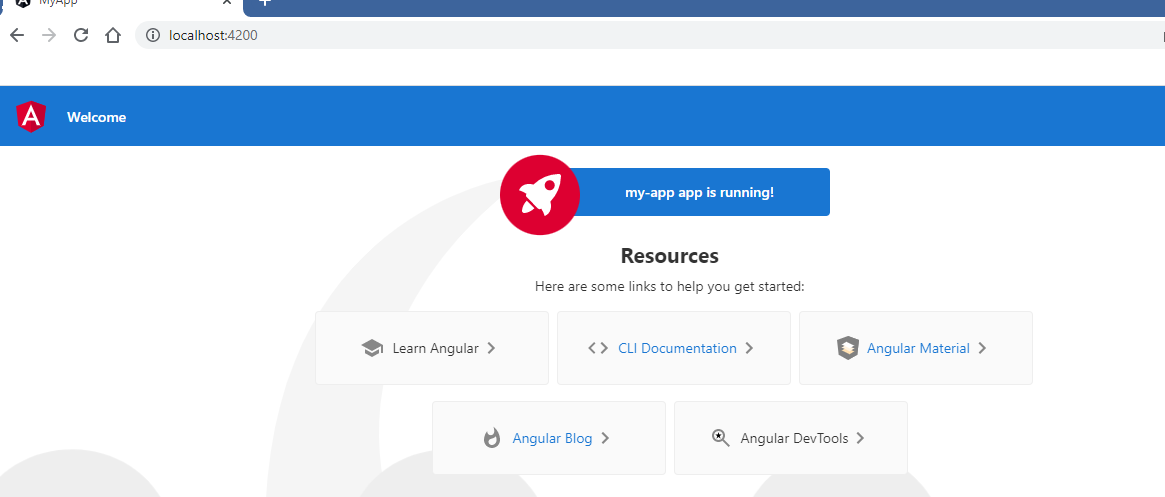


The above entries specifies the project is created, if git is installed in your machine it may show the info as above, else it may show some warning which you can ignore

ng serve: It is a command to launch angular application, it launches the application in port number 4200

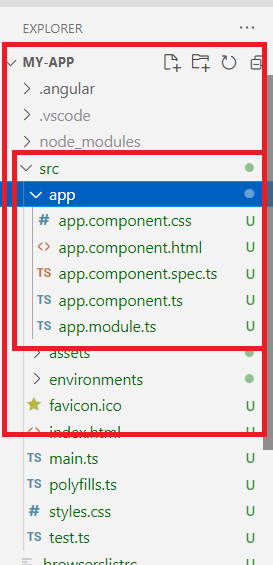
After entering ng serve you get below entry in the terminal





Now we may not need this content, hence you can open the my-app in the vscode & edit the components.

The below screen shot shows the structure of the project



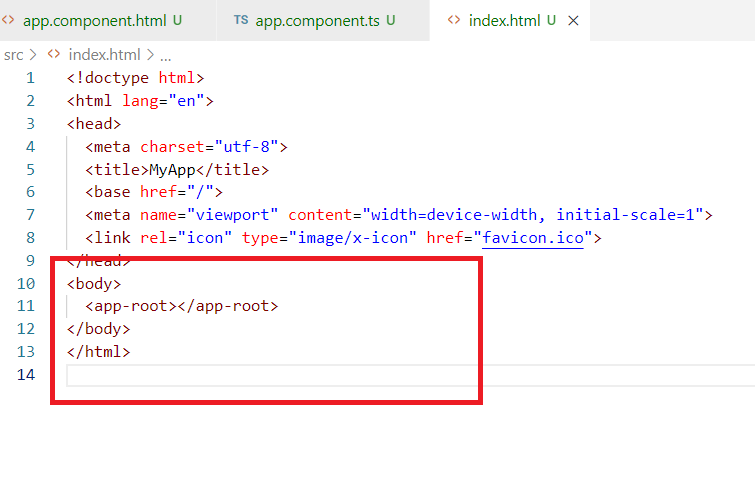
In the above structure there are some .ts files which are typescript files this is where you write all your application code

app.component.ts is the file that has root component code

app.component.html is the file that has root component content

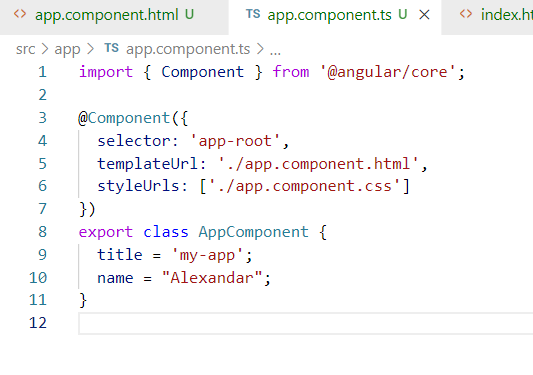
index.html: this is the only file that is loaded in the browser, root component tag <app-root> is used in this file

The index.html will by default have <app-root> this is root component



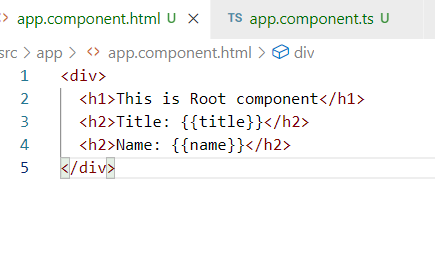
<app-root> is the selector for the root component, you will only work in the root component and other components

app.component.ts

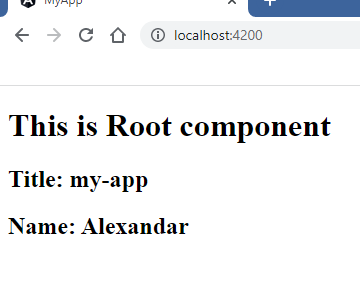


You can notice @Component({}) having selector which is the tag name ‘app-root’ used in index.html, the app-root will load the content of app.component.html

app.component.html



Output:



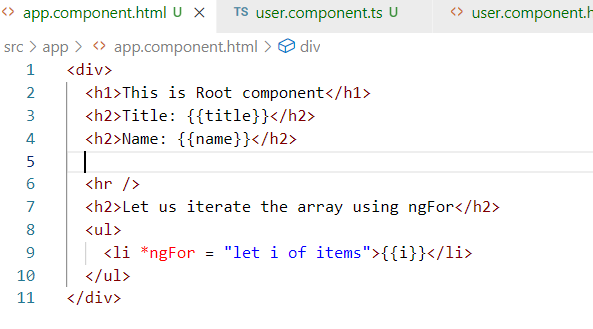
Components: These can be created using commands in angular like ‘ng g component-name’

Like we can display the data, we can also display the array elements using \*ngFor

app.component.ts

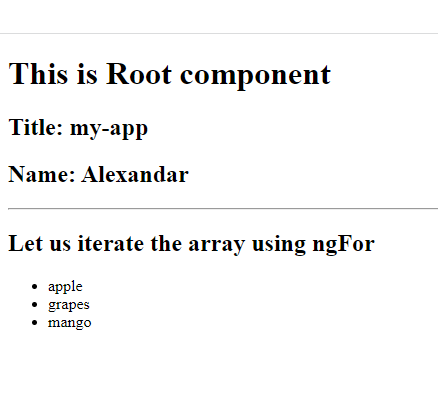


app.component.html

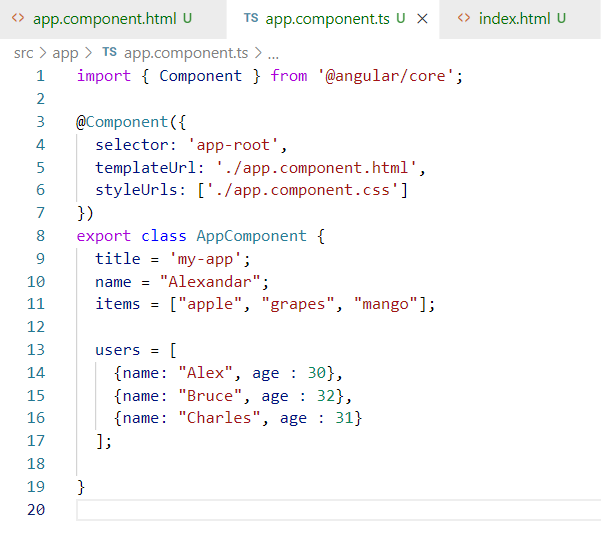


Here \*ngFor iterates items and in each iteration it displays the element using <li>

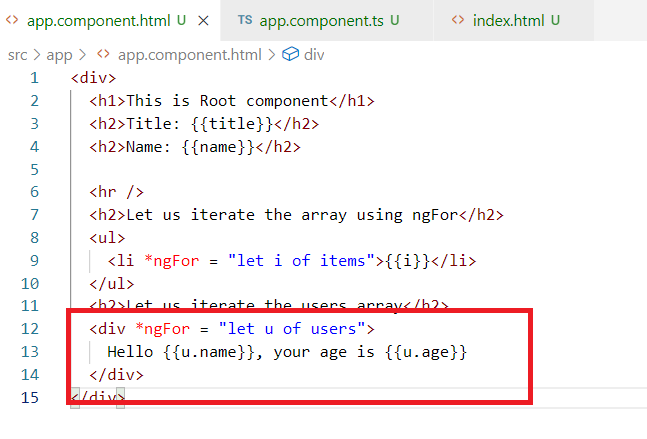
Output:



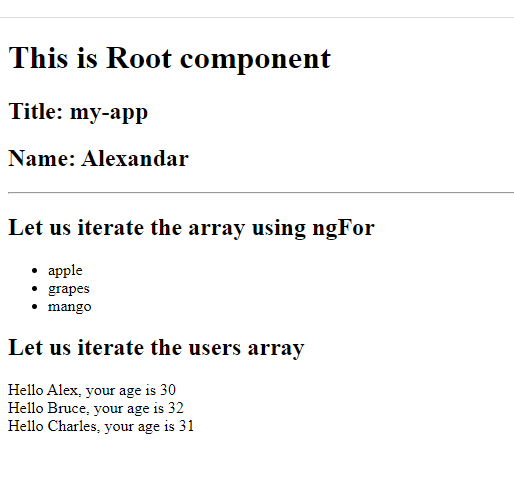
Similarly you can also iterate complex objects in the array



app.component.html



Output:



Creating same kind of components in React

React allows you to work without any tool kit, but you need to refer 3 library in React

1. React library
2. React DOM library
3. Babel library

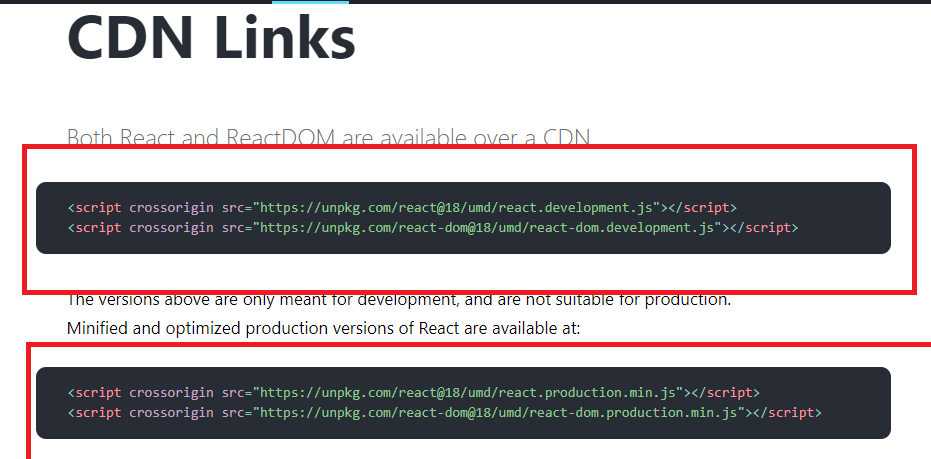
All the above libraries you can refer online using <script>

React library: It allows you to create components

React DOM: It patches the components to the Browser DOM

Babel: It is to convert JSX to Javascript

CDN Links for React & ReactDOM library



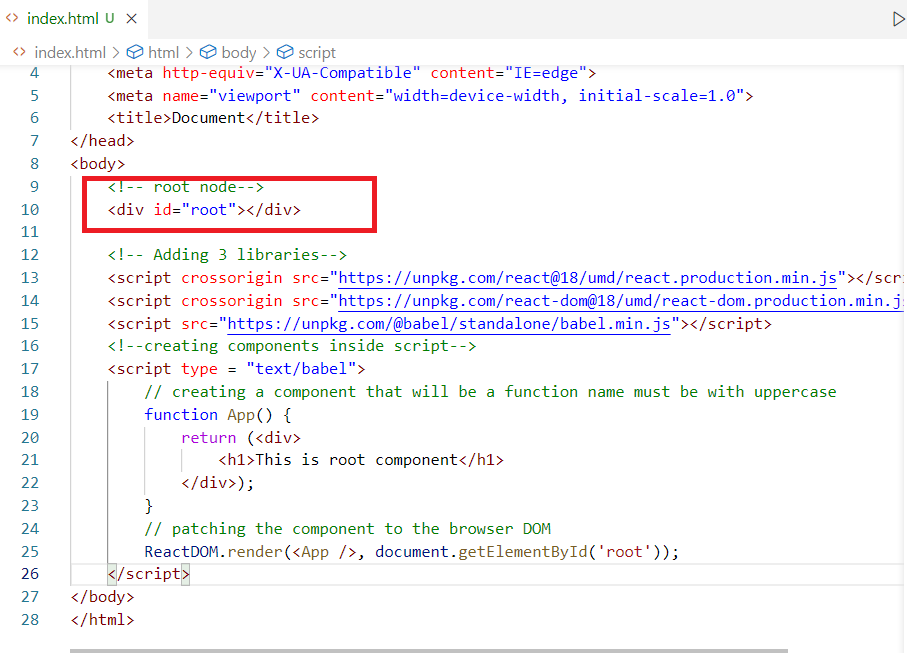
You can either use development edition or production edition

Babel cdn link

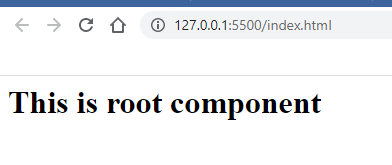
<script src="https://unpkg.com/@babel/standalone/babel.min.js"></script>

React also uses a root component where all the components are part of the root component.

index.html

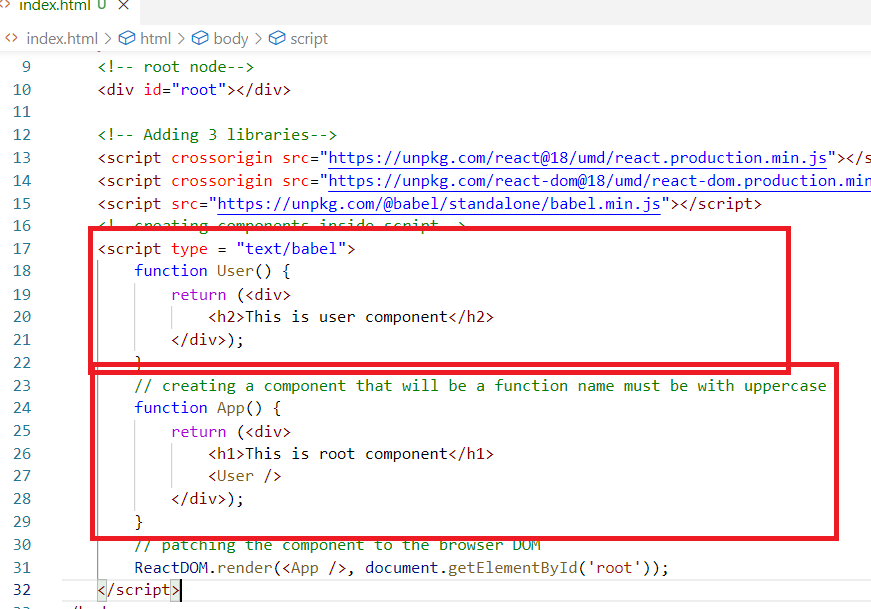


Output:

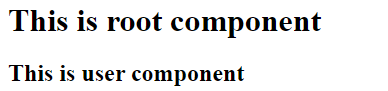


We can create multiple components and use it in other components (root component)

index.html



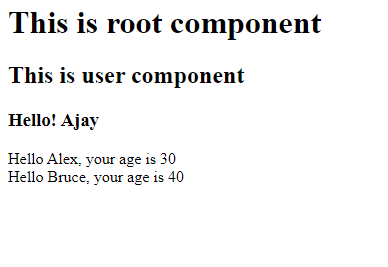
Output:



How to render the data in the React components



Output:



Activity:

1. in React create an array of users and display them in the table

the array users will have following data

users = [{name: "Alex", age: 35, gender: "Male"}, {name: "Suma", age:40, gender: "Female"}]

1. Same array use in the angular and show them in a table
2. Go through this official document and try out some examples

<https://reactjs.org/>

Databases:

Database helps you to store the data which you can manage through the query language it supports

RDBMS: RDBMS stands for Relational Database Management System, which maintains the data in a table format and the table can have relation with other tables using some constraints

List of RDBMS databases

* MySQL
* Oracle
* Derby
* Postgre SQL
* MS SQL

MySQL is an open source database from Oracle.

SQL: Structured Query Language, which is the language used by the database to maintain the data

SQL allows you to write different types of queries like insert, update, delete, create, alter, drop and so on, based on the operations we do SQL is divided into following sub-types

1. DDL: Data Definition Language
2. DML: Data Manipulation Language
3. DRL: Data Retrieval Language
4. TCL: Transaction Control Language
5. DCL: Data Control language

DDL: Deals with structure of the table like create, drop, alter, truncate

DML: Deals with data manipulation in the table like insert, update, delete

Truncate vs Delete

Truncate operations are permanent which can’t be undone and it deletes all the data in the table, you can’t delete only a few data like applying condition & deleting is not possible

Delete operations can be rolled back ie., you can undo the operations, you can apply conditions & delete only a few data

DRL: Deals with retrieving the data, it has select command

TCL: Deals with saving or undoing the DML operations, it has commit & rollback

DCL: Deals with providing permissions or revoking permissions to the users, it has grant & revoke commands

MySQL datatypes

* int
* varchar
* double
* date
* datetime

These datatypes tells the table what kind of data it can store in its column

Login to cloud labs -> open Readme.txt file & observe the command to open mysql terminal

Normalization:

It is a process where you reduce the data redundancy (duplicates), to make the data more consistent we need to use many constraints while creating the tables.

Different types of constraints

* primary key
* unique
* check
* not null
* foreign key

Create tables

Syntax:

create table table\_name(column\_name datatype <<constraint>>, …..);

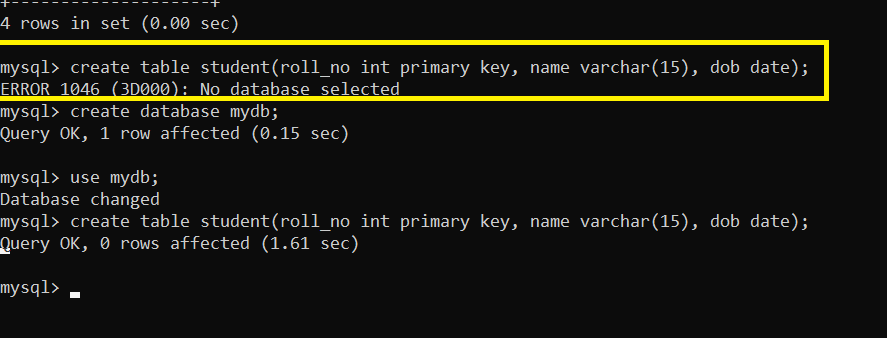
In MySQL we need to create a database name and use it to maintain the tables.

How to create database

create database database\_name;

How to use database

use database\_name;

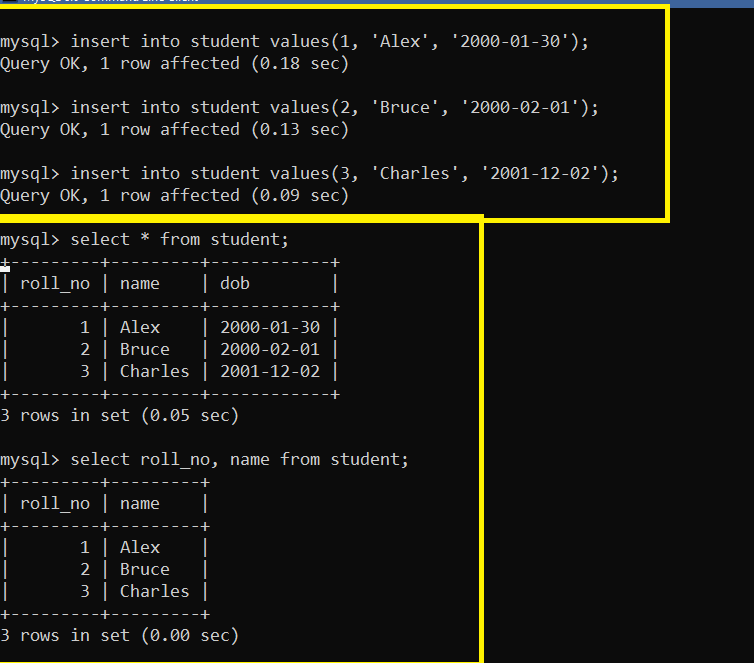


We can use different types of queries like insert, update, delete, select to perform the CRUD operations.

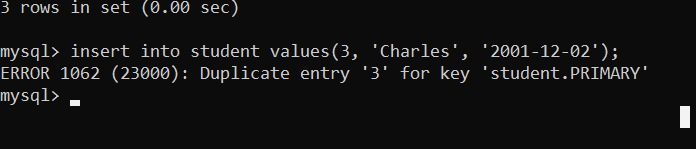
Insert the record

Syntax: insert into table\_name values(…, …., …., …..);

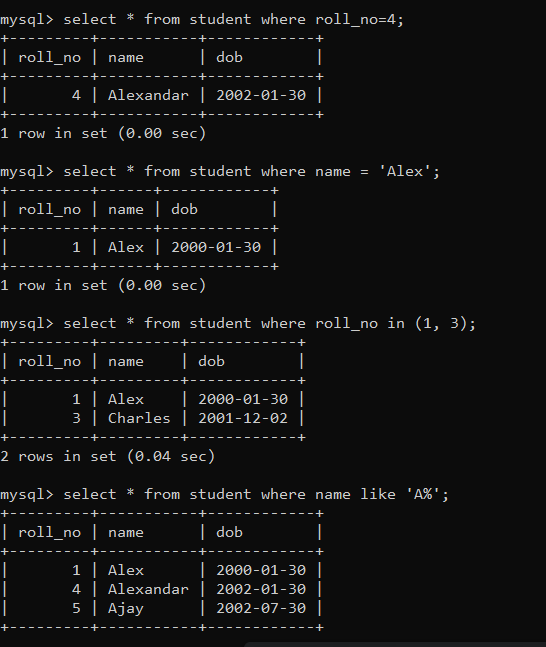
Note: date must be stored in YYYY-MM-DD format i.e., 2022-06-17



What happens if you try to insert duplicate primary key

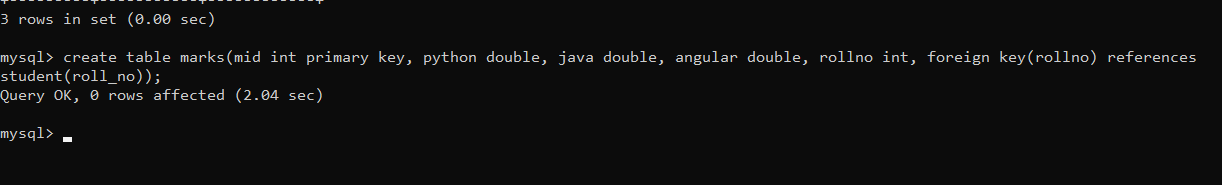


Select query with conditions

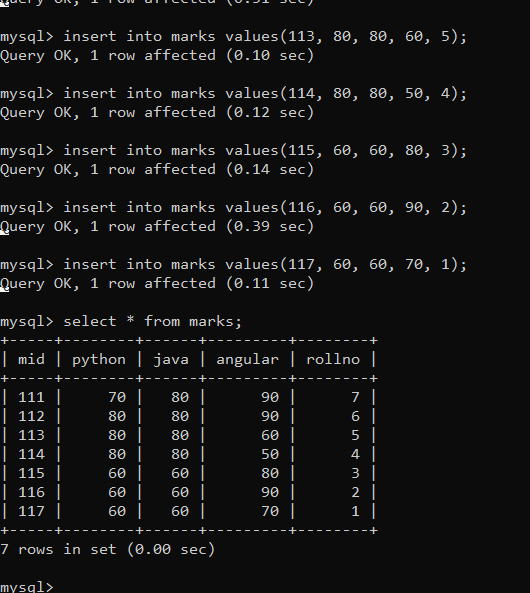


Creating foreign key: It is used to link the record with another table

Syntax: create table table\_name(column\_name type <<constraint>>, …, foreign key(column\_name) references parent\_table(primary\_key\_column\_name),…);



Foreign key column can have null, but its value must always be something which is present in the parent table



We need to join marks & student table to know which student has scored what marks

There are different of joins in sql, but here we are using inner join, which only gives the matching records of both the tables.

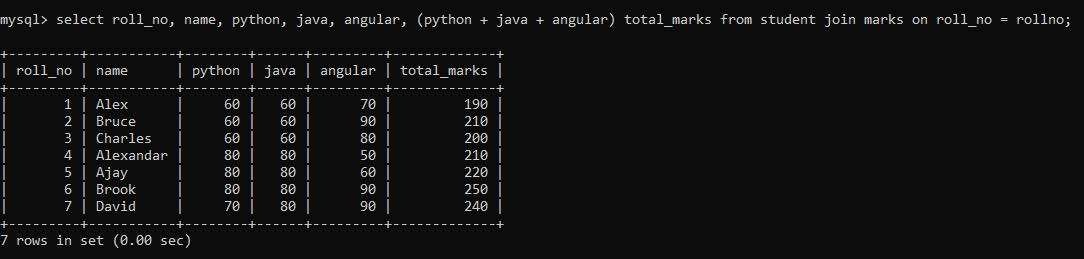
Inner Join syntax:

1st syntax:

select column\_name, column\_name from table\_name join table\_name on <<condition>>

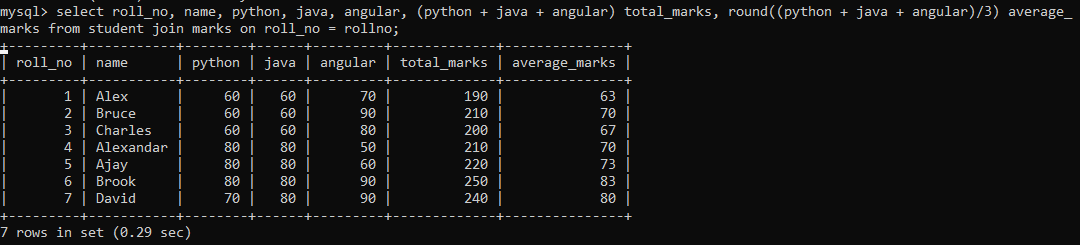
2nd syntax: you can use alias name to join when both the tables columns have same names

select alias\_name.column\_name, alias\_name.column\_name from table\_name alias\_name join table\_name alias\_name on <<condition>>



Note: It is not generating any table,

We can calculate the average and round up their values



NoSQL: Which stores the data in the database but it is not structured like SQL, in case of structured database the data will be limited however if its un-structured then we can unlimited data in any format/structure.

NoSQL being unstructured allows any data dynamically even those data which are new type of data that comes in the future.

MongoDB is one type NoSQL Database it keeps the data in the form documents and these documents(records) are javascript objects and they all are stored in a collection (like a table)

MongoDB provides predefined functions to store, update, retrieve and delete the documents from its collection.

MongoDB provides two important commands to work with the database

* mongod: This command starts the mongodb database in your machine
* mongo: This command opens the mongo shell and allows us to perform operations in the mongodb

How to work with mongodb

1. Firstly we need to start the database using the command mongod
2. Secondly we must keep that mongod running and enter another command mongo to enter into the mongo shell

Note: You need two terminals

Mongodb also needs some database instances to be created hence below are the commands you can enter

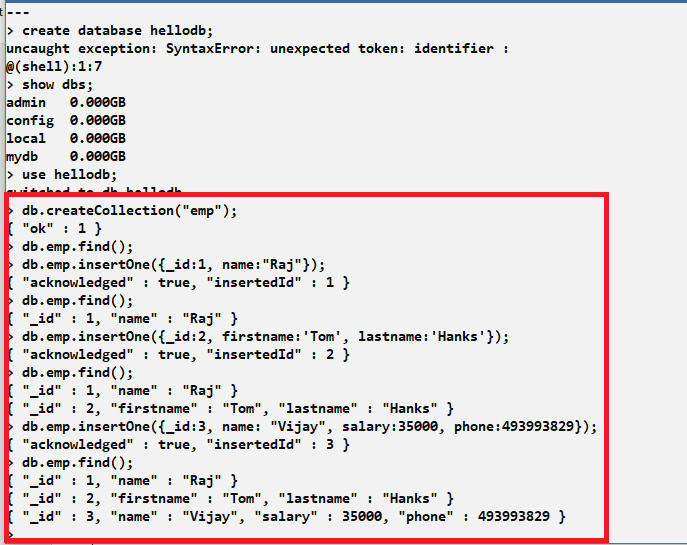
use database\_name;

db.collection(“emp”)

db.emp.insertOne({key: value,….});

db.emp.find()

|  |  |
| --- | --- |
| SQL | NoSQL |
| Table | Collection |
| Rows/Records | Documents |
| Columns | Properties |
| Structured | Unstructured |

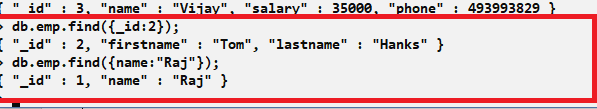


Query the documents with the conditions:

db.emp.find(): returns all the documents

db.emp.find({\_id:1}); returns the document having the \_id: 1

db.emp.find({name:’Raj”}); returns all the documents having name:Raj



Activity:

1. Create a collection with the name ‘messages’
2. Store a document having \_id & text
3. Store another document having \_id, text and comments
4. Store another document having \_id, text & likes
5. Store another document having \_id, text & a nested document having username and phonenumber
6. Query the documents based on the \_id

Note: text & comments can be some strings, likes can be some numbers

API:

API stands for Application Programming Interface

APIs connect software’s to communicate with each other regardless of the languages they are implemented in.

These are online API’s which makes other applications to communicate or exchange the data in a common format more preferably JSON/XML so that it can be converted to any formats like Java, Javascript, Python, C++, C, C#, Cobol and so on.

What is JSON/XML

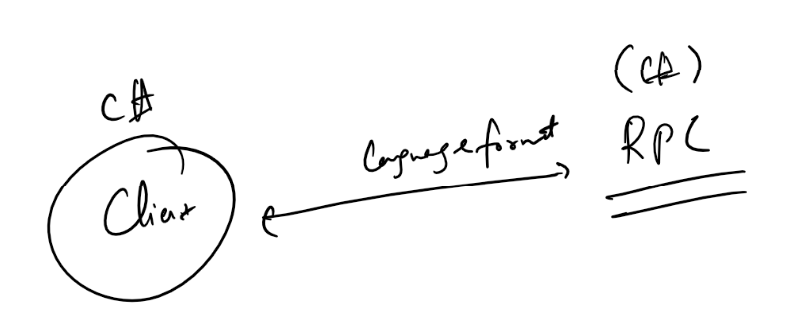
It is a structure which can carry the data that needs to parsed by some programming languages, it can be converted to any language.

There are many types of API’s, these API’s are treated as Remote API’s

Different types of API’s

1. RPC (Remote Procedure Call)
2. SOAP Webservices
3. ReSTful Webservices

RPC: Remote Procedure Calls are Online API’s but client & server both must exchange the data in the same language format



Drawback of RPC

1. Different programming languages can’t communicate
2. If same application needs to be written for different platforms it needs to be rewritten in different languages

To address the problem Webservcies came into picture

Webservices are online API’s makes two applications written in different languages to communicate with a common format that is easily converted to their respective language.

There are two types of Webservices

1. SOAP (Older)
2. ReST (Newer)

SOAP: Simple Object Access Protocol: It uses only XML format for data transfer

ReST: Representational State Transfer: It uses many formats for data transfer like JSON, XML, Text, CSV, HTML

Note: 99% of the time data format will be JSON as its easy to understand & parse

API’s can be written in any languages, these are called as Webservices because they are online services available to all the software’s

We can use Spring Framework to create Webservices

Spring is Java framework used to develop variety of applications like Desktop, Mobile, Web, Cloud related

What is a Framework?

It is like a semi implemented application which makes easy to implement complex applications.

Why Spring Framework is used

1. Provides lot of design patterns to implement lot of complex features
2. Gives us ready to use objects which are configured out of the box to perform lot of complex operations ex: Dependencies that are supplied to other objects, Database connections
3. Easy to write applications for different scenarios like Web, ReST API’s, Cloud computing
4. Provides annotations to make things automatic like object creation, component scanning

Spring Container:

It is a context used by the Spring to register all the objects that needs to be supplied or assigned to other objects.

Dependency Injection:

It is a design pattern where objects are supplied to depending objects, and ensures that the code is loosely coupled for the changes

Earlier we need to create object as

A a = new Impl();

With Dependency Injection

You will give chance to spring framework to create object & supply that object to the dependencies

@Autwowired // this injects implementation of A present in spring container  
A a;

Spring has 100’s of annotations for different purposes, some of the important annotations are

@Autowired: This is to inject objects

@Controller: This is to create controller class

@RestController: This is to create APIs

@Service: This is to create Service layer/business layer

@Repository: This is to create DAO layer

Practically to start with you need to know below things

1. Spring Dependency Injection
2. MVC architecture
3. Java

Software’s required

* Java 8 +
* Eclipse IDE
* Postman application

How to create REST API’s using Spring Boot

Spring Boot helps you to quickly create ready to run applications it provides you lot of automated features behind the scene.

To create spring boot projects you can use spring official website spring.io which has an initializr project.

Visit below URL to create spring boot project

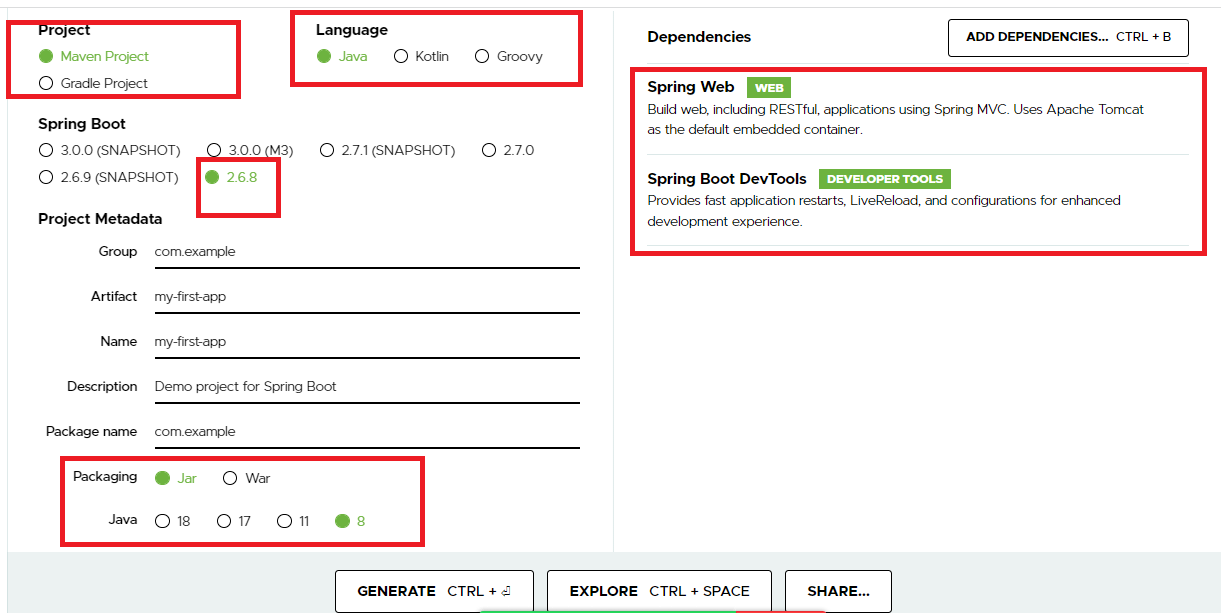
<https://start.spring.io/>

Maven Projects: These will automatically download libraries from the internet, in maven project you will have 3 main metadata

1. group id: an unique id of the organization like com.ibm, com.sapient, com.oracle, com.google and so on
2. artifcat id: project name
3. package name: Java package name

The website gives us ready to run projects, but you need to have some libraries, which you can get from the Add Dependencies, some of the libraries are:-

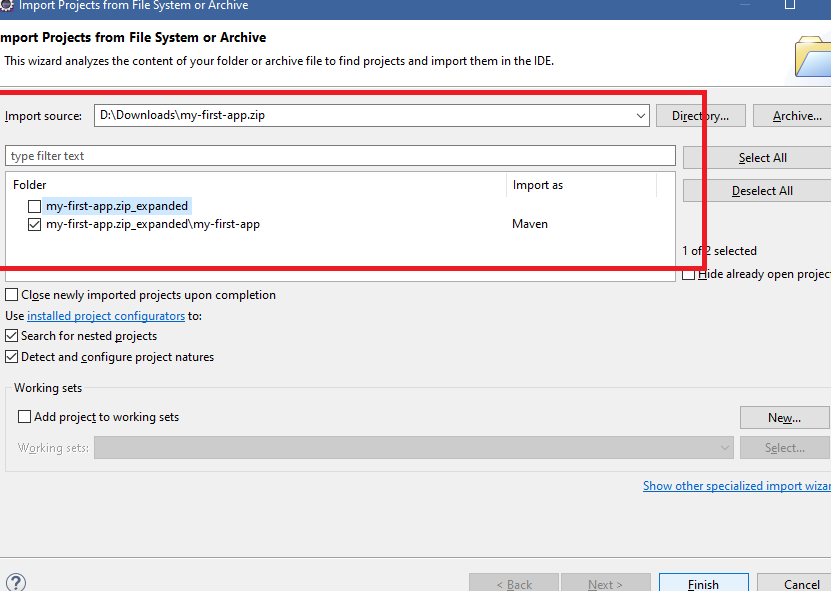
1. Web: It is to develop Restful Webservcies, it provides an inbuilt server, it gives you important annotations like @RestController, @Autowired, @Service, @Repository, …
2. Dev tools: It is to auto-reload the server when you make changes in the code



Note: Click on Generate it will download a zip file that needs to be opened from the eclipse

How to open from eclipse

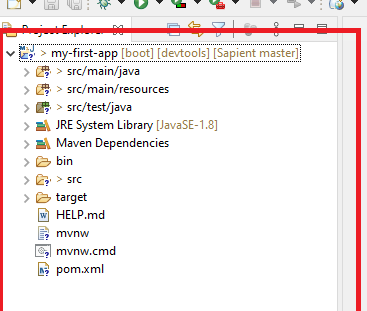
File -> Open Project from file system -> Select Archive -> Browser for the zip file -> select the project zip -> Select Import as Maven -> click OK or Finish.



You can notice that we have selected the 2nd checkbox that has Import as Maven

Note: Once project is selected you must able see the project in the eclipse, you need to wait for the project to complete download all the libraries from the internet.

Below is how the project structure looks like

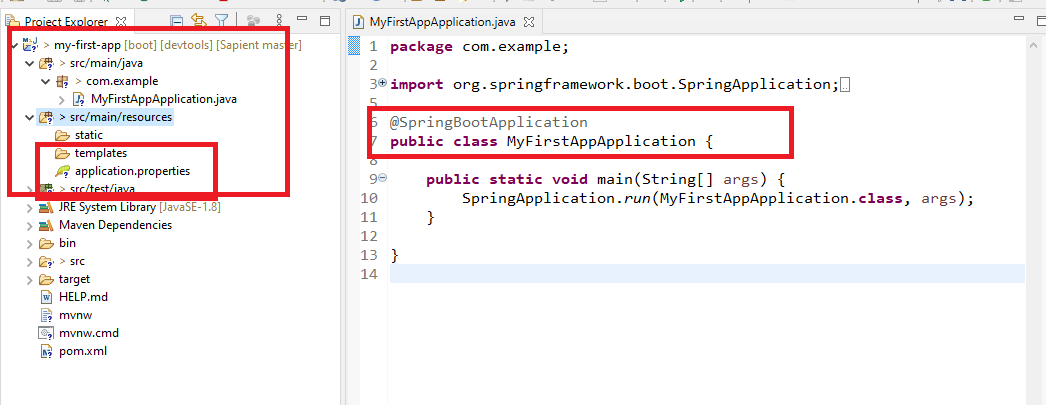


Most important files or folders we access are

* src/main/java
* src/main/resources
* pom.xml

Main method: Spring boot provides you a main method whose class is annotated with @SpringBootApplicaiton, this provides us all the automated features like inbuilt server, auto-scanning of classes like @Service, @RestController and so on

application.properties: This file helps us to write some application specific configurations



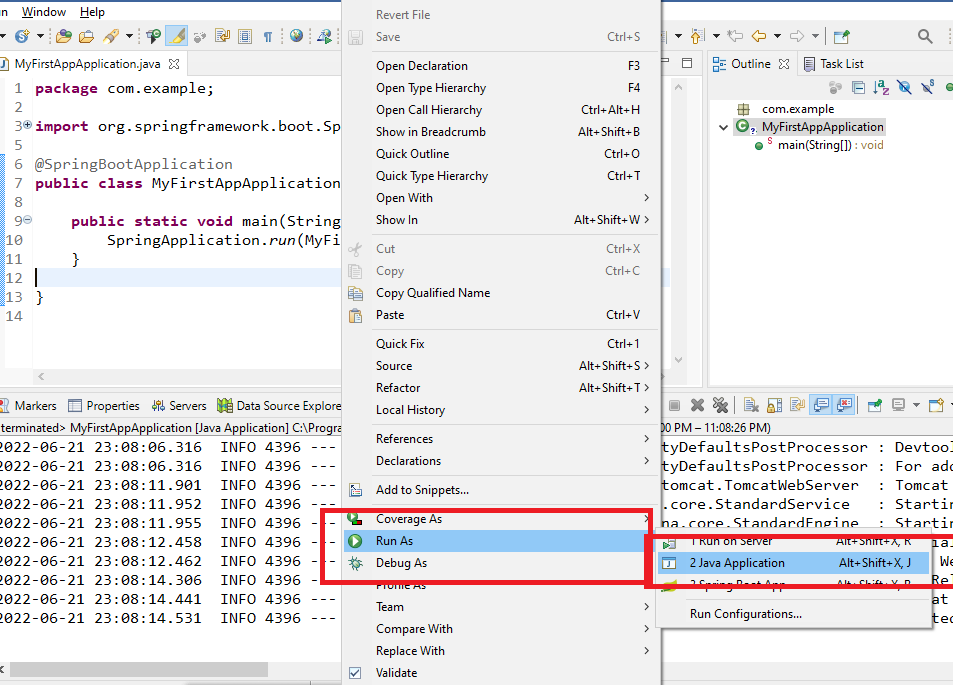
All our classes must be inside com.example strictly or in the sub-package of com.example

Where does the inbuilt server runs

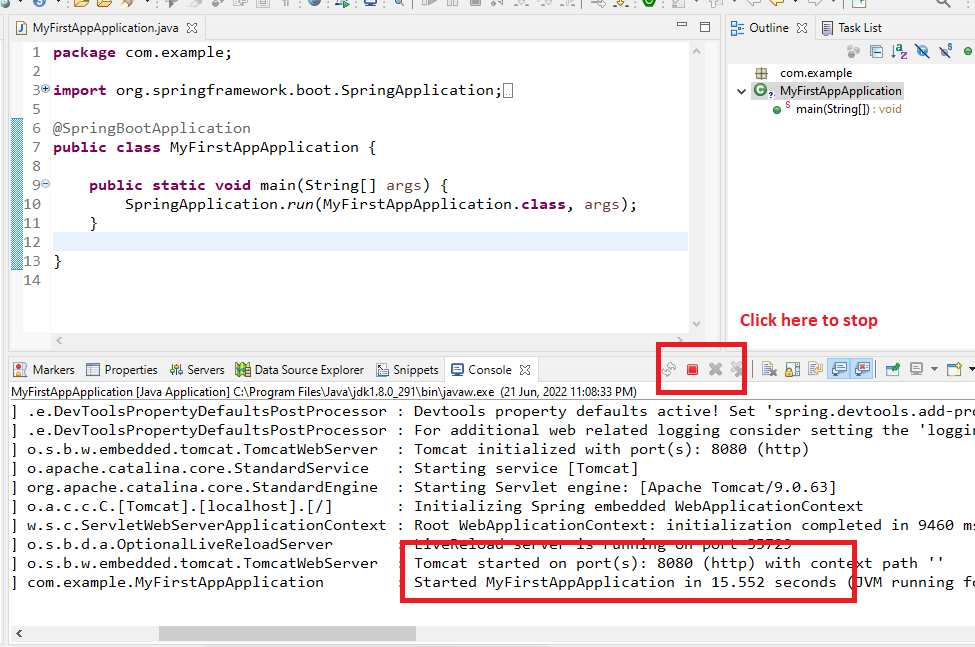
When you run your main method embedded server i.e., tomcat starts by default in 8080 port number.

How to run the application

Right click -> Run As -> Java Application



You will see console logs as below, at the end you can observe tomcat started at 8080 port



What we did?

1. Created the spring boot project from Spring Initializr
2. Hosted the empty project without any logics

Activity 1:

Create your first spring boot project

Duration: 15mins (complete by 2pm)

Designing webservice API’s

While designing webservices application use 3 main features

1. URL: to specify the webservice URL
2. Http methods: to specify the operation (Post, Get, Delete, Put)
3. Structure: to specify the data structure

URL suggests the webservice end point address

Http methods suggests what kind of operations webservice is trying to do like to create a new resource POST is used, to fetch the resource GET is used, to update the resource PUT is used, to remove the resource DELETE is used.

Structure suggests the type of data like JSON

Note: You can have multiple webservices with same URL but different HTTP methods.

abc() & xyz() methods can have same URL like /hi but different http methods one is POST & other is GET

How to create the webservices using Spring

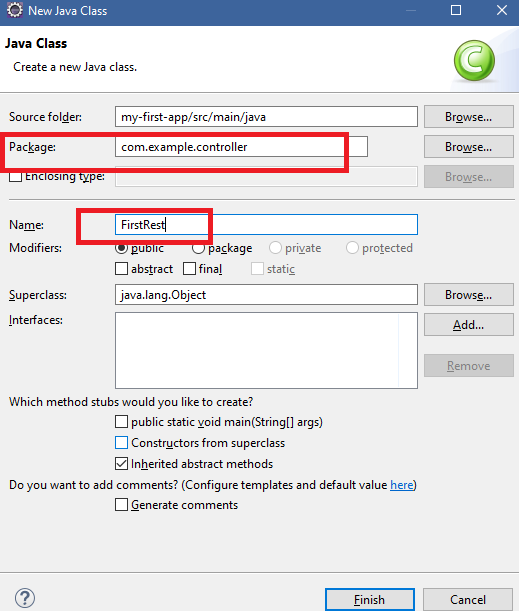
1. @RestController must be created on top of the class
2. @RequestMapping is used to specify the URL
3. Http methods are specified using some annotations like @PostMapping, @PutMapping, @GetMapping, DeleteMapping, even these methods can also specify URL like @RequestMapping

i.e.,

@RequestMapping(“/api”): It works for any http method, it is usually used for the entry level URL  
@GetMapping(path = “/greet”): works only for GET  
@PostMapping(path=”/store”): works only for POST

Creating webservices in the same project

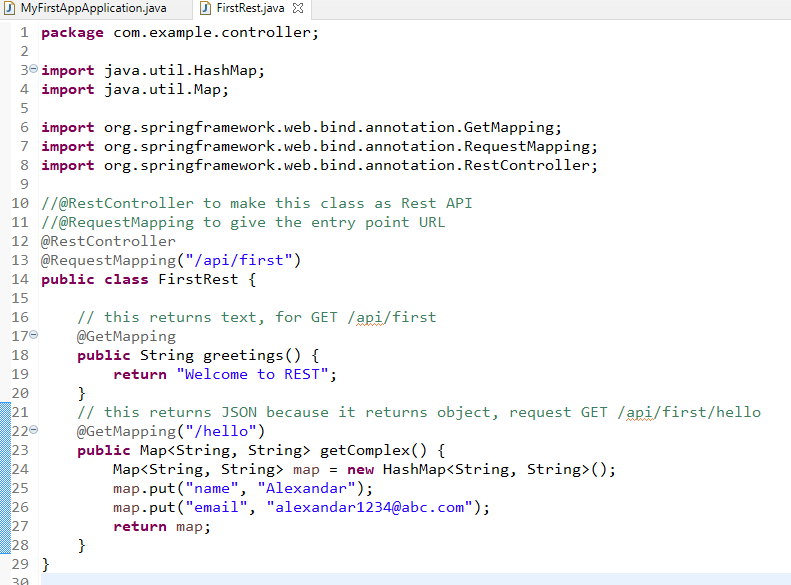
Note: Create a class with @RestController & ensure the class is inside the top-level package or subpackage of it.



This class needs to act as a Rest API, so you must use

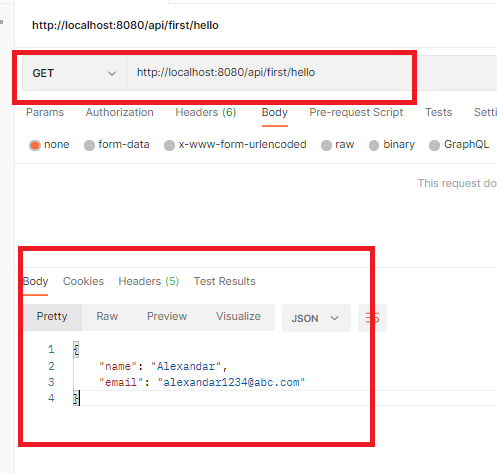
* @RestController on top of the class
* @RequestMapping on top of the class to mention the URL i.e., main entry point url
* Http methods mapping must be mentioned on top of the methods

FirstRest.java



You can run your application and test in postman application

Note: Since the server runs in 8080 port from postman you must use <http://localhost:8080/api/first>



Here this JSON is used by the consumer & converts that JSON to consumer language & should able to process the data either to show on Mobile client or Web client or for some other operations.

Activity:

Firstly try the above example & then

Use 3 more mappings @PostMapping, @PutMapping & @DeleteMapping with the same URL /hello and they all must return JSON data as below

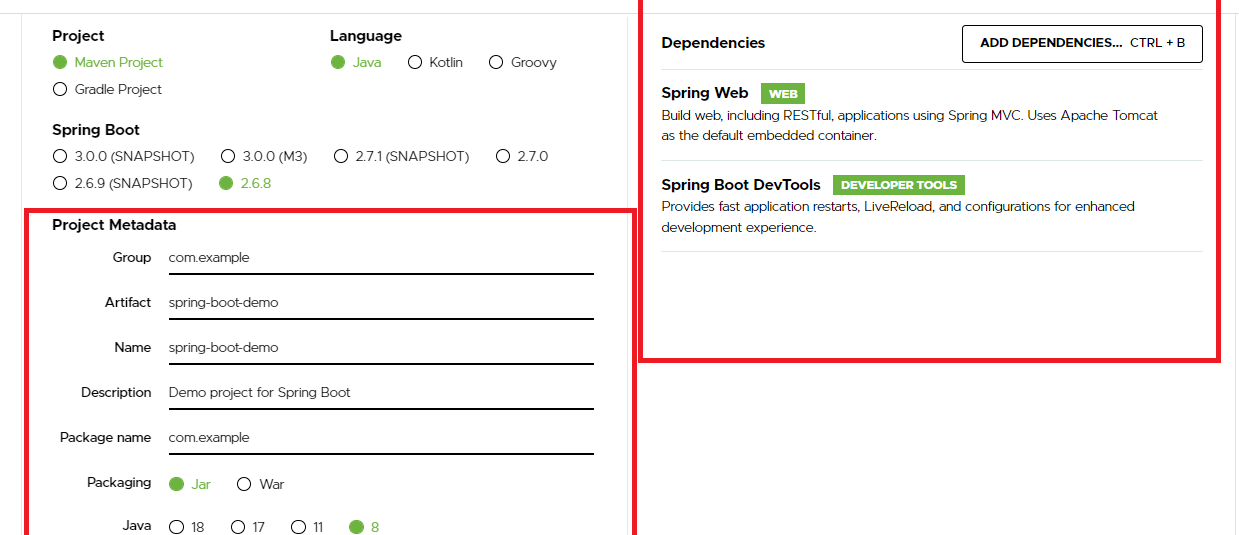
{“message”:”Hello <<method-name”}

Note: if you send request as POST /api/first/hello then you must see {“message”:”Hello post”} similarly for other methods you must see different message.

Things we can do in Spring boot

* Working with properties files
* Working with YAML files
* Modifying the server ports
* Creating an executable jar file
* Overriding the properties at runtime
* Adding different servers like Jetty / Undertow
* Service Layer
* DAO layer
* ORM Framework in Spring Boot
* H2 database (In Memory database)

Creating a new project spring-boot-demo



Every new project from spring boot gives a ready to run application.

@SpringBootApplication: This takes care of performing automatic configurations based on the libraries you add in your classpath, i.e.,

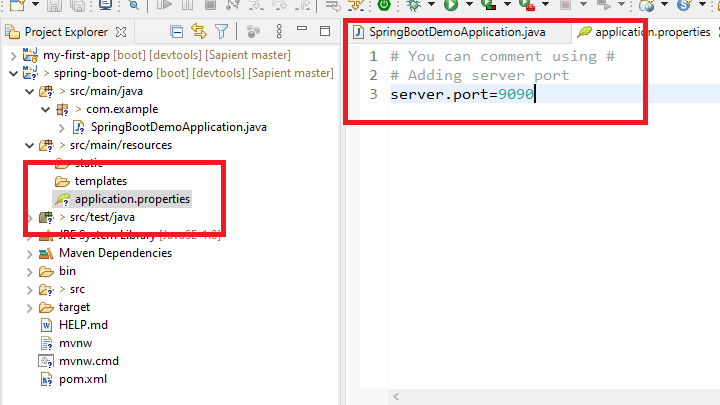
* Web: This library will add embedded tomcat server, sets the port to 8080 by default, configures all the required set up to create Web API’s
* Data JPA: This library helps spring boot to automatically configure connections with the database
* Actuator: This library provides you lot of url’s to monitor your application health, status

application.properties:

This file will have all the application related configurations like server.port, context path, datasource details, actuator details, microservice details and so on.

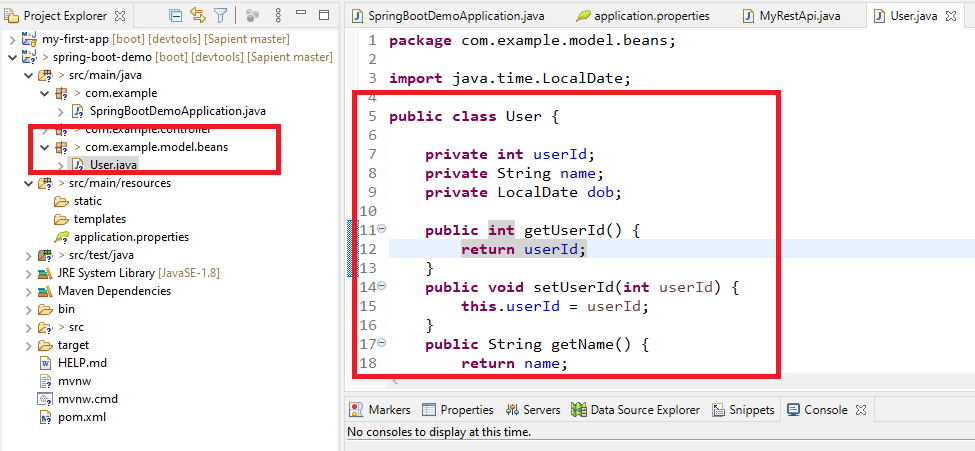
Suppose you want to change the port number of your server then you have a property called server.port = 9091 this sets server port to 9091, it uses lot of inbuilt properties

Configuring application.properties to modify the port & the context path

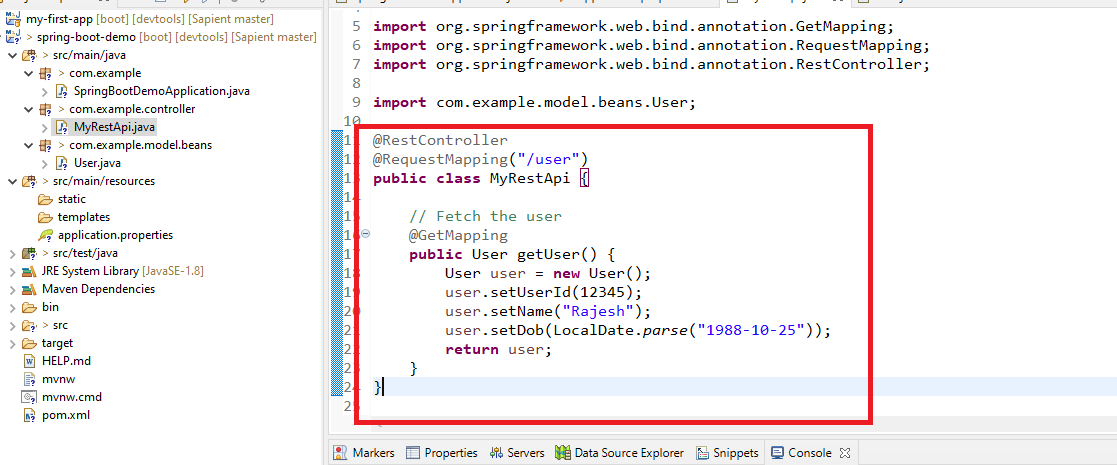


Now we can create a Java bean that can be returned as a JSON through some REST endpoints i.e., URL & HTTP methods

User.java

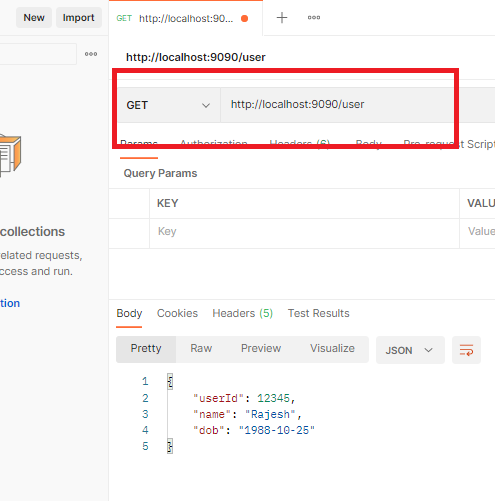


MyRestApi.java



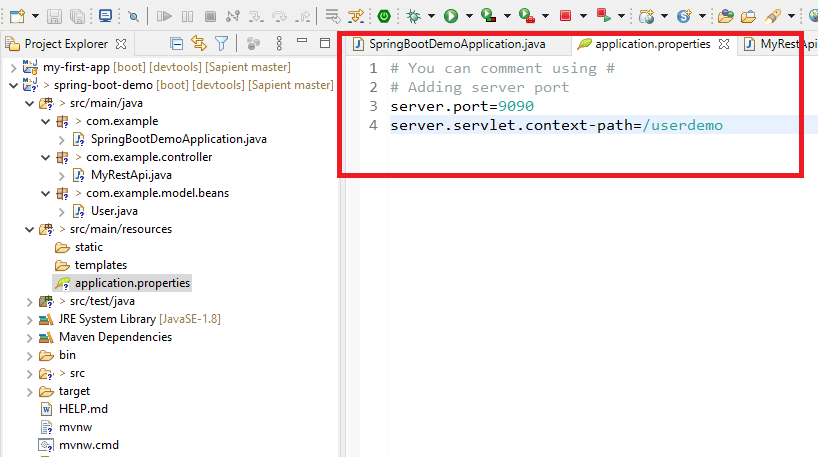
Now we can launch our application & see the dummy user details in the postman using GET /9090/user

Output:

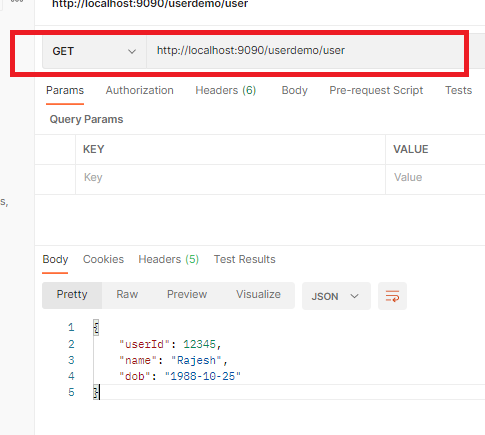


Note: context path is empty, you can configure it in the application.properties

application.properties



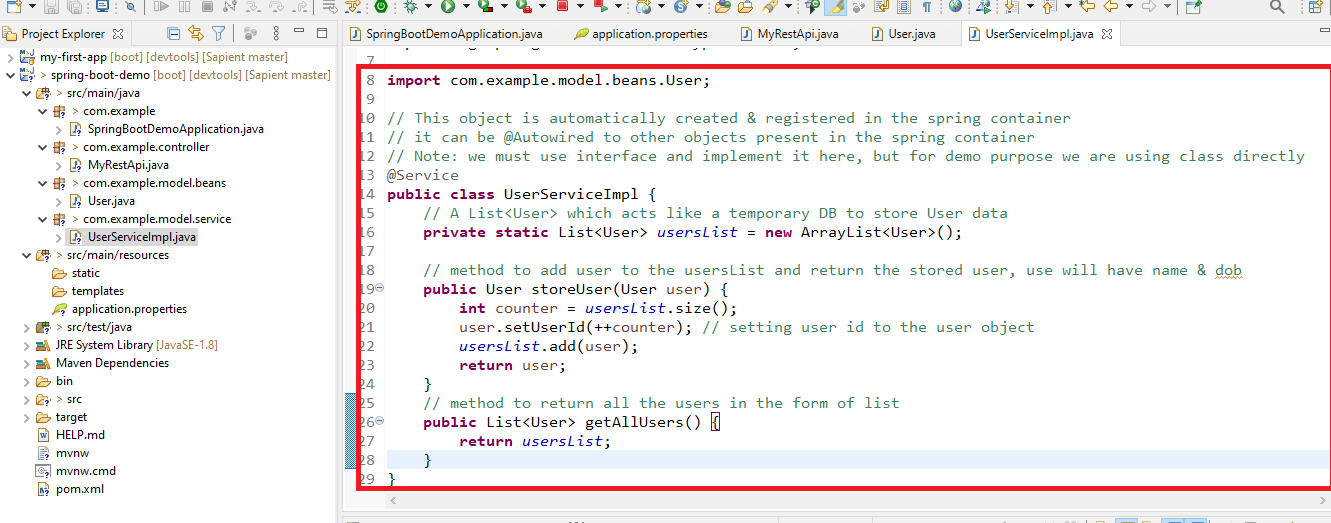
Now onwards you can access your webservices using <http://localhost:9090/userdemo/paths>



The above code is just using some temporary data but what we can use is we can create a Service layer that is accessed by Controller

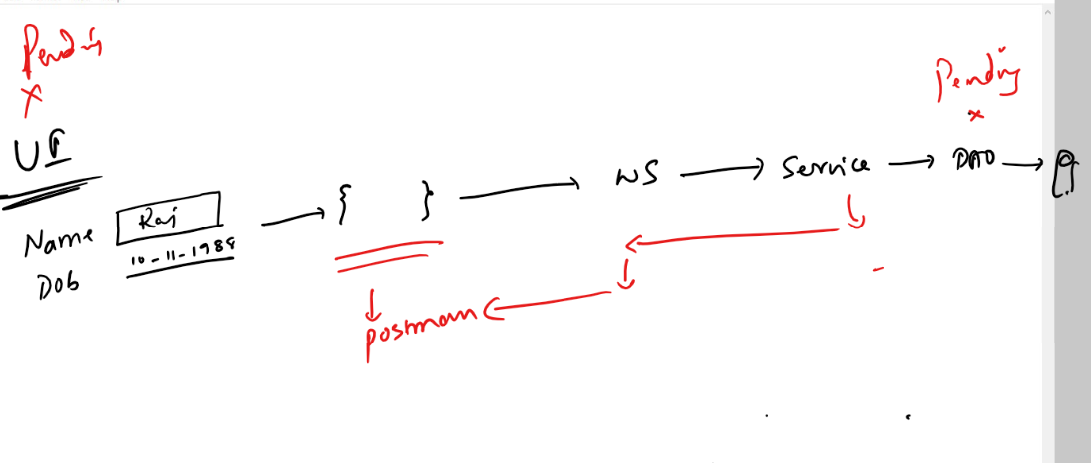
We can create business logics in the service layer

UserServiceImpl.java



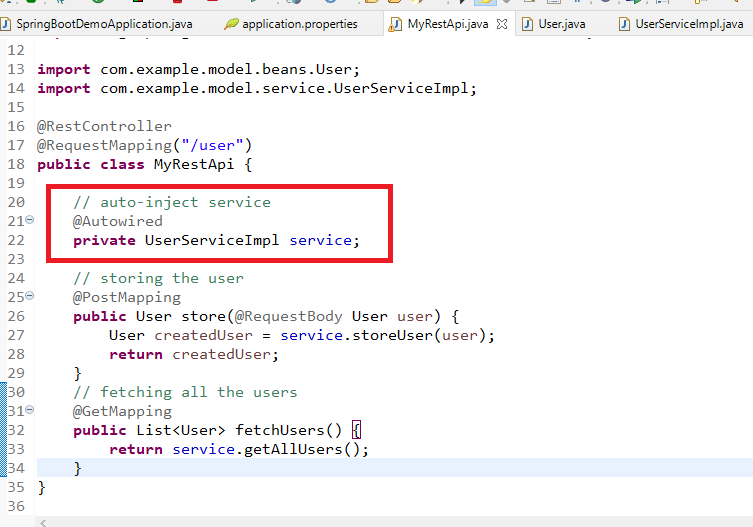
The storeUser & getAllUsers are invoked by Controller the storeUser gets the user object from the Controller, the controller gets it from the request made by the end user

i.e.,



Now we need to use the instance of Service in the Controller and call the methods like storeUser & getAllUsers

MyRestApi.java



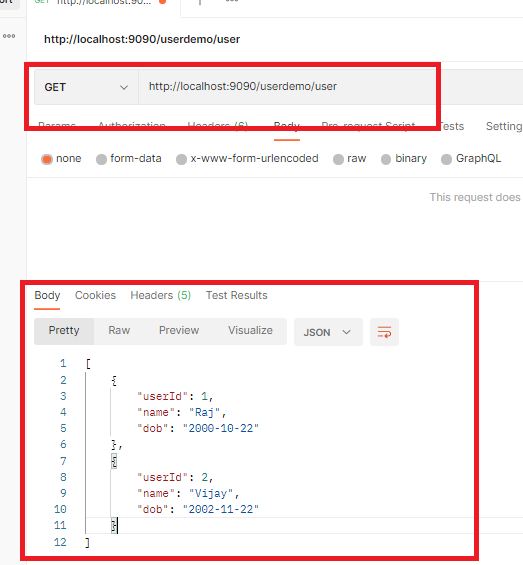
We have here 2 REST end points one does store & another does fetch, since we don’t have any UI we have to use postman on behalf the UI

Output:

Firstly we can store the data using POST



We can make a GET request



Note: To get the auto-suggestions for application.properties check for STS plugin in eclipse market place

Different ways of passing inputs to the webservices from the client

1. Request body
2. Request path parameters
3. Request query parameters

Request body: Here the entire data of the entity is supplied in the JSON format which your webservices can extract using @RequestBody

method(@RequestBody Type entityName)

Request path: Here the data is supplied in the path which your webservices can extract using @PathVariable

/{id}: This can take any value whose value is stored in id, this matches to /100, /200, /300 and so on

/{name}/{dob}: This will take 2 parameters name & dob it can take any value /Raj/2000-10-20

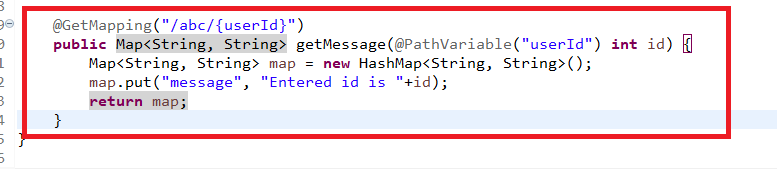
Ex:

@GetMapping(“/{id}”)  
Now you can send request to this using /10, /20, /30 and so on, the @PathVariable(“id”) can extract 10, 20 & 30

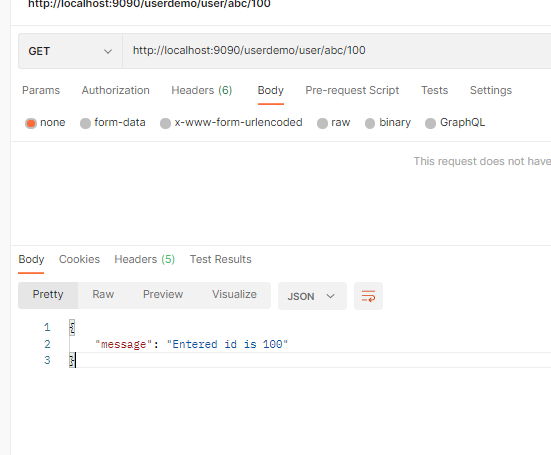
@PostMapping(“/{name}/{dob}”)

Now you can send request to this using /Raj/2000-10-20, /Rajesh/1998-11-22 and so on, the @PathVariable(“name”) extracts Raj or Rajesh & @PathVariable(“dob”) extracts 2000-10-20 or 1998-11-22

@GetMapping(“/id”): Then you need to enter /id itself, however if its @GetMapping(“/{id}”) then it can have dynamic value



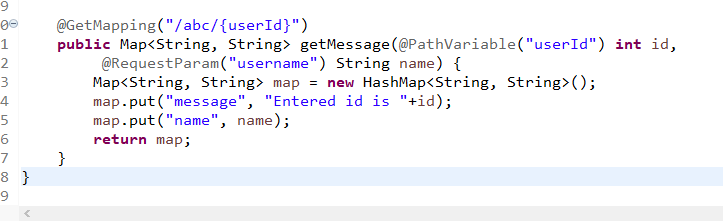
Output:



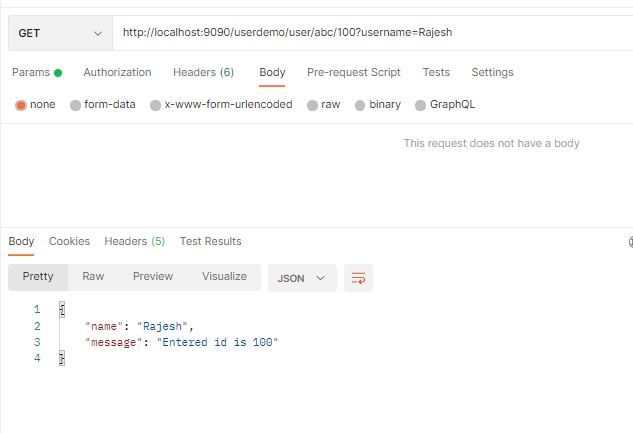
Request Query Parameter: These are query parameters sent in the request url at the end of the path using ? separating each key and value with ‘&’

URL?key=value&key=value

You have @RequestParam to extract the value associated with the key



Output:

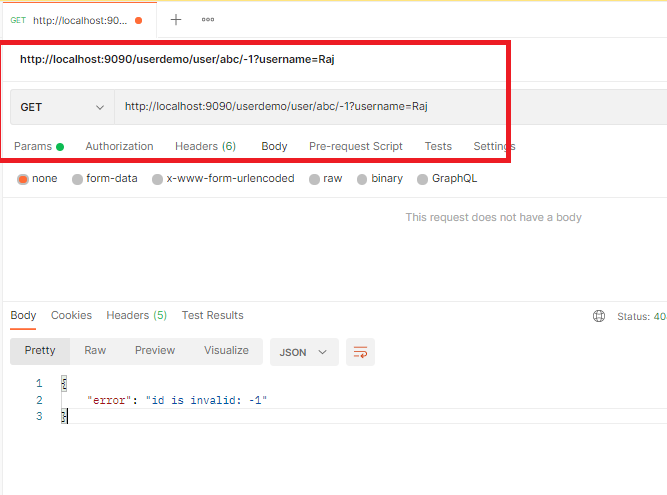


ResponseEntity:

It is used to change the response content & the status as by default response will only show the 200 as the status code & embeds the content what we return, when we want different status code or different contents based on the condition then we can have ResponseEntity as the return type.



Output:



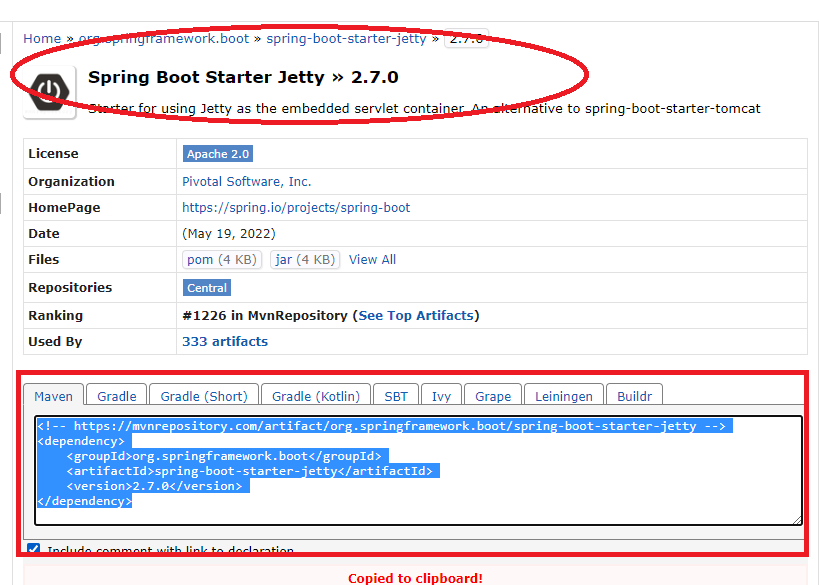
Adding different embedded servers in spring boot

The default embedded server is tomcat, but spring boot provides some other embedded servers like

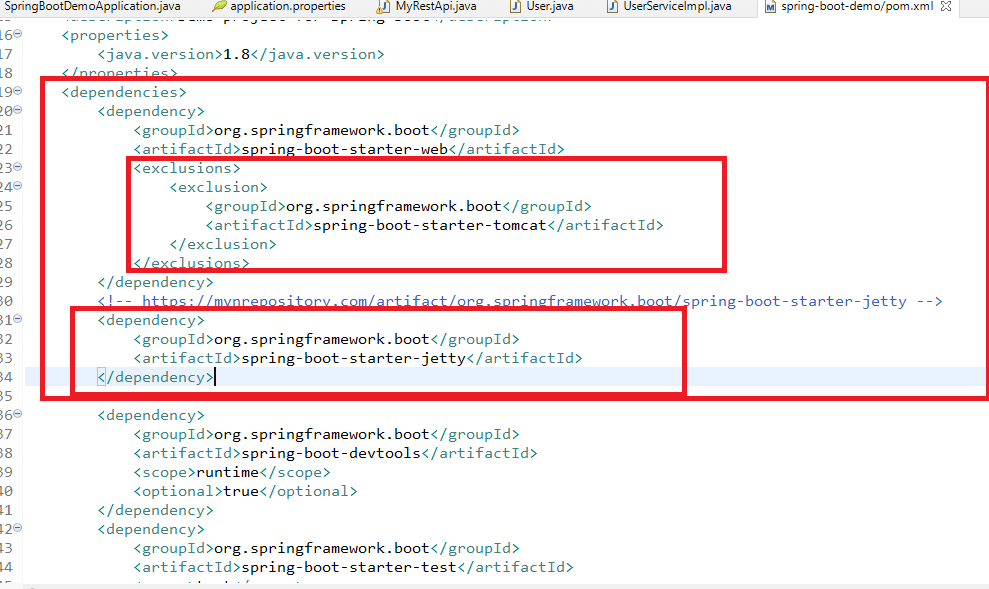
* Jetty server from eclipse
* Undertow server from Jboss

You need to exclude tomcat server in pom.xml file and add different servers in it, we need to search jetty & undertow in the maven repository.

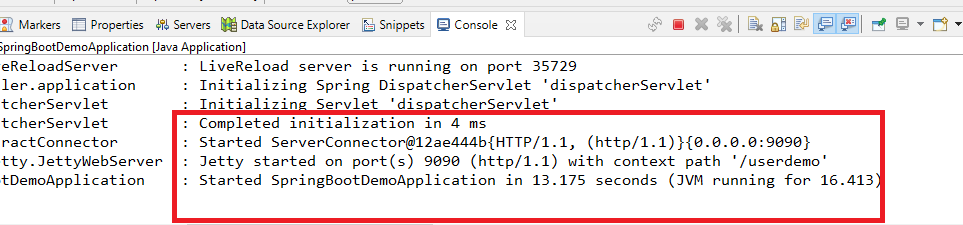
You can visit maven repository and search for the libraries



pom.xml



You can re-launch the application to see jetty server logs in the console



Note: Similarly you can also add another server Undertow which is from JBoss

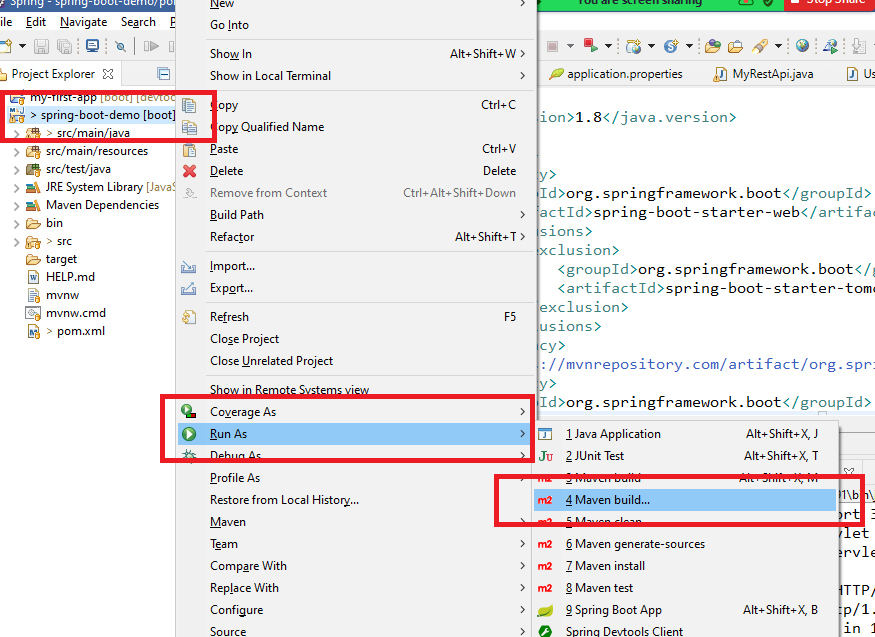
How to run our application in different machines or in cloud environment

You need to build your application as an executable jar & share that jar to run on different servers (using GIT or some other way you can share the jar files).

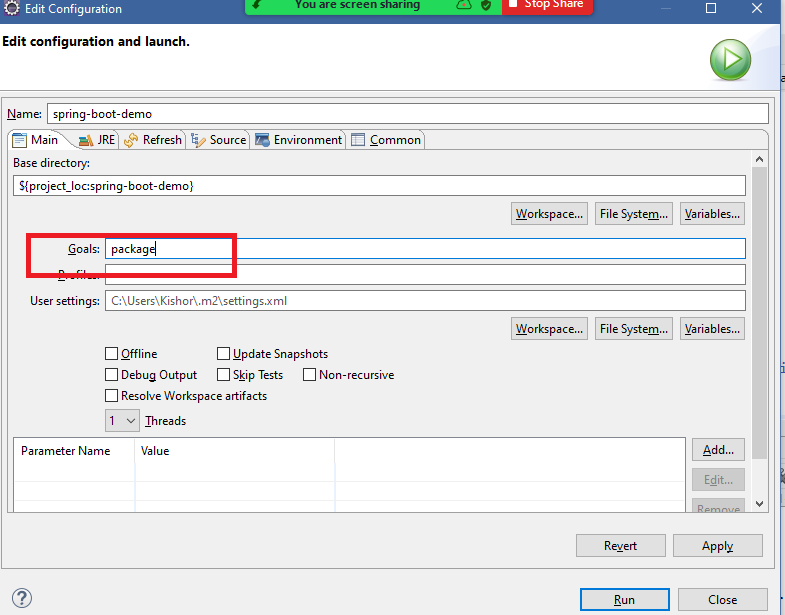
To create an executable jar we need to use Maven option package in the Eclipse that will build the jar file.

Steps:

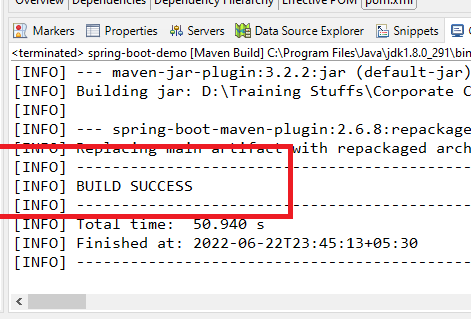
Right click on project -> Run As -> Maven Build… -> Goals: package -> This creates an executable jar file in the project/target folder



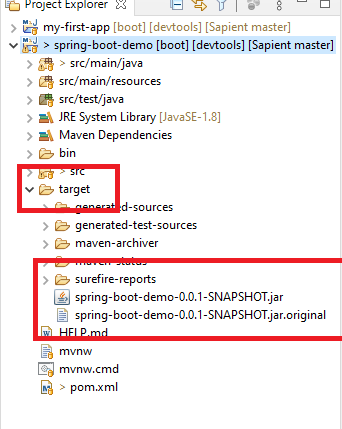
Choose package:



You will see Build Success in the console



Now you can refresh the project to see jar files in the target folder



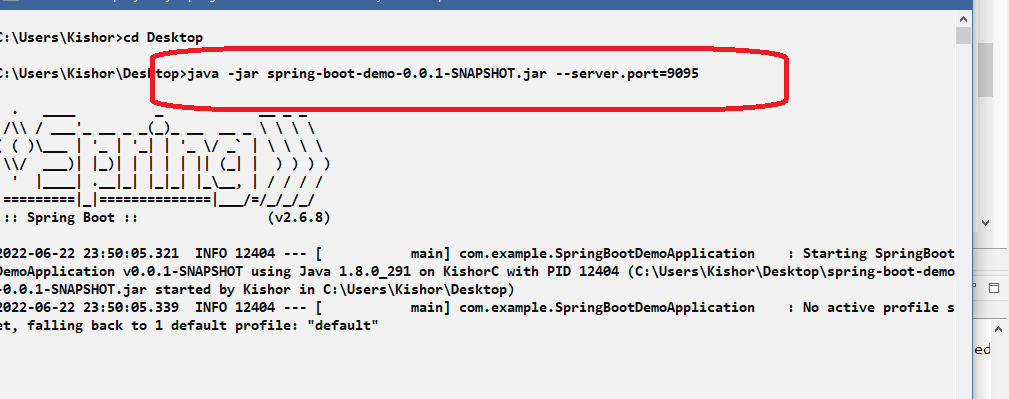
Now you can execute this jar file using java -jar command in any machine that has java installed.

Note: you can override the properties while running the jars as well

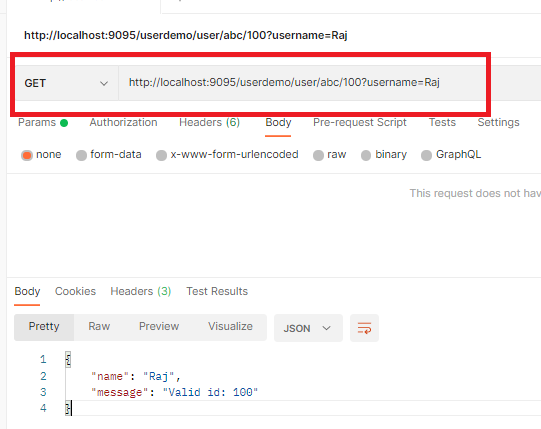
>> java -jar file-name.jar

The above command runs the application in the port you have given in application.properties

>> java -jar file-name.jar --server.port=9092

The above command uses different port that overrides port from application.properties

We can still send request to the port 9095 which is overridden from application.properties



Summary:

1. Created a spring boot project
2. Created Rest Controller & Service layer
3. Implemented @GetMapping & @PostMapping
4. Used ResponseEntity
5. Created an executable jar
6. Overridden the properties

Activity:

1. Add Undertow server instead of Jetty Or Tomcat & Observe whether it works
2. In the same example implement few more operations that manages the User in the Service layer that can fetch user by id and also update the user dob from the id and delete user by id
   1. Fetch User by Id: If you pass the userid from the postman in the path i.e., user/100 if it exists then you can show in postman the user data in json format else you show the error message in json format i.e., {“error”:”sorry user with an id 100 not found”}
   2. Update user dob from the user id: From postman pass userid & dob that may update the user dob i.e., /user/100/1998-10-25 and return the updated user in JSON format if in case id is found, if id is wrong then you must return the error message in JSON format like {“error”:”sorry user with an id 100 not found”}
   3. Delete user by id: From postman pass user id ie.., /user/100 and delete the user if id exists and return the message json format like {“message”:”100 is deleted”}, if id is not found return the error message in JSON format like {“error”:”sorry user with an id 100 not found”}

Note: Try to use appropriate HTTP methods and also for Delete by id & Fetch by id the URL will be same but will have different HTTP methods

Note: You have some inbuilt methods in the List to iterate, remove items in the List

Important points to consider in the above example

1. Always have an interface in the Service layer and implement it
2. Use that Service Layer Interface reference in the Controller instead of Implementation reference
3. Always throw the checked exceptions wherever required so that the caller could handle it and perform the task when exception is handled

ORM Framework

ORM stands for Object Relational Mapping, where you can map java objects to relational database tables without writing any lower-level code like establishing connection, writing queries to perform CRUD operations, closing connections, converting java types to sql types & vice versa and so on

JDBC is also a way to interact with the database similar to ORM Framework, but JDBC does lot of lower level task like handling exceptions, establishing connection, creating queries, performing type conversions, closing connections and so on

In JDBC if you write 10 lines, in ORM same thing you can achieve in 3 to 4 lines



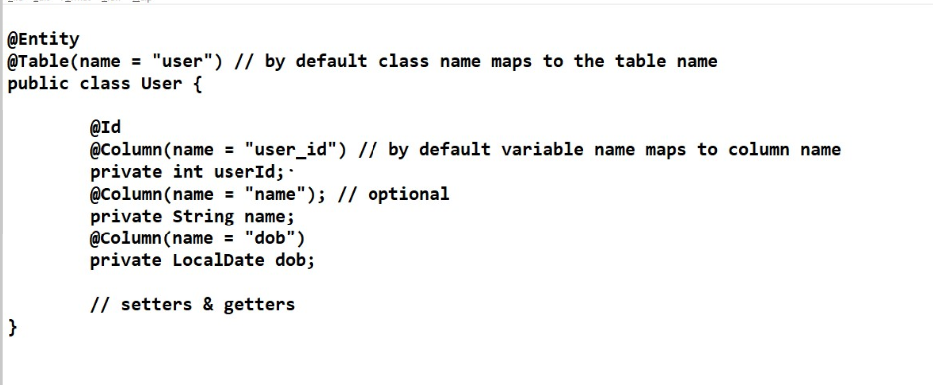
How does ORM Framework work

Entity class: This is a normal java bean class, which will have metadata about the tables & columns like the class is mapped to which table and variables are mapped to which column

When the object of the entity is created then it maps to the table record & their properties maps to table columns.

Configurations files: These are going to have database details like database URL, username, password, driver

Program to perform CRUD Operations: These are some inbuilt classes or methods that help use to perform CRUD operations, they automatically connect to the database using the configurations provided in the configuration file



ORM is a framework which is implemented by many organizations/community

* Hibernate
* Spring
* iBatis
* Toplink

Spring gives us lot of features when we work with ORM like

* Separating the database configurations (application.properties) and the code
* Provides automatic features like establishing connections, supplying the database related dependencies to directly perform the CRUD operations
* Provides some inbuilt Repositories that will have almost all the logics you do in the database like store, update, delete, retrieve, but you can add your own logics also in the Repository
* Automatically takes care closing the database resources, transactions

Repository

These are some interfaces provided by Spring to perform database operations, these interfaces are generics, they don’t have idea about on which table they work, we need to extend them to mention that these repositories works with which entity so that it knows it works with which table.

There are two main repositories spring provides using a library called Spring Data JPA

Spring Data JPA: A library that can perform operations with the database, JPA Stands for Java Persistence API, ORM actually follows every rule given by JPA.

Spring Data JPA will provide a proxy implementation for the repositories so that developers don’t need to implement the repositories

Spring Data JPA provides two repositories

1. CrudRepository<T, ID>
2. JpaRepository<T, ID>

CrudRepository is the top level interface which has crud operations, JpaRepository extends CrudRepository and it has crud operations and other operations like pagination, sorting

Note: Both CrudRepository & JpaRepository are generics type they don’t have idea their methods work on which table

We must extend it to specify they work with which table.

Suppose you have a User which is mapped to user table i.e., @Entity @Table

@Entity @Table  
public class User { …. }

Then we can create an interface that extend either CrudRepository or JpaRepository to specify it must be mapped to which table.

interface UserRepository extends JpaRepository<User, Integer> { }.

List of methods we have in both CrudRepository or JpaRepository

1. save(T t): T
2. delete(T t): void
3. findById(ID id): T
4. findAll(): List<T>

If you observe all the methods take generic type, the interface which extends will specify the operations must be on which entity.

i.e.,

interface UserRepository extends JpaRepository<User, Integer> { }

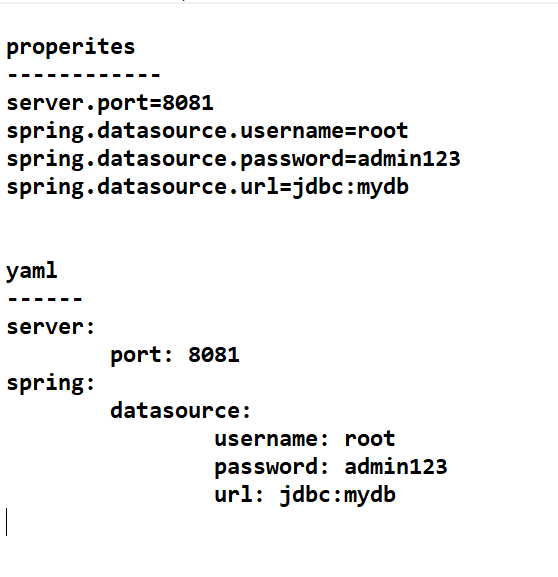
Now all the methods of JpaRepository & CrudRepository changes as below:-

1. save(User user): User
2. delete(User user): void
3. findById(Integer id): User
4. findAll(): List<User>

Note: You don’t have to implement this interface (UserRepository) Spring Data Jpa does it out of the box, we just need to use the interface reference

YAML files:

These are also property files which are used as a replacement to properties file, they use indentations to separate the parent property with sub property so that you can avoid repeating parent properties for multiple configurations

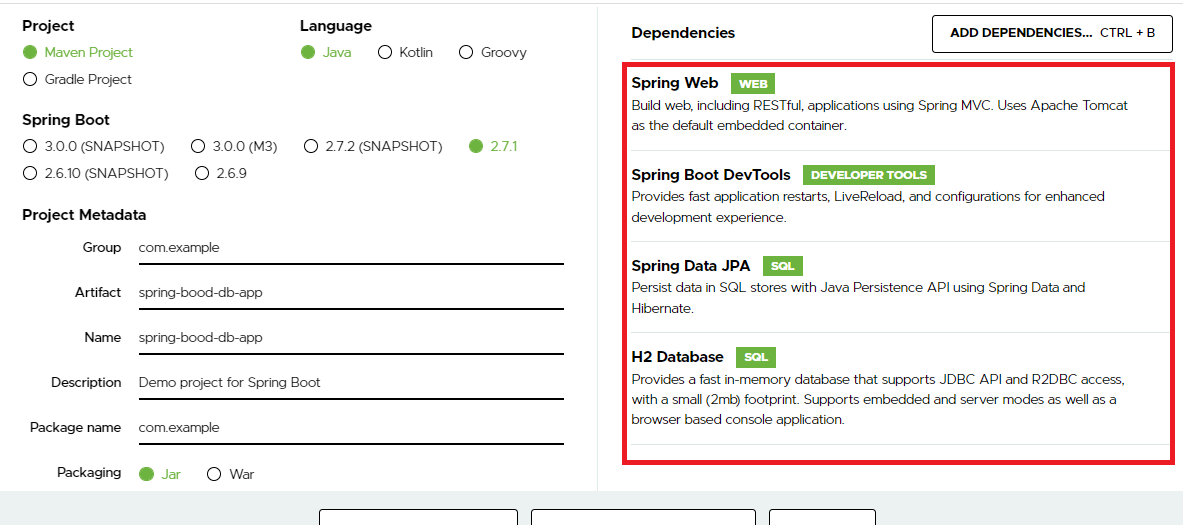


Working with Spring Data JPA

Spring Data Jpa provides you the library that helps you to interact with any relational database like MySQL, Derby, PostgreSQL, Oracle, H2 and so on

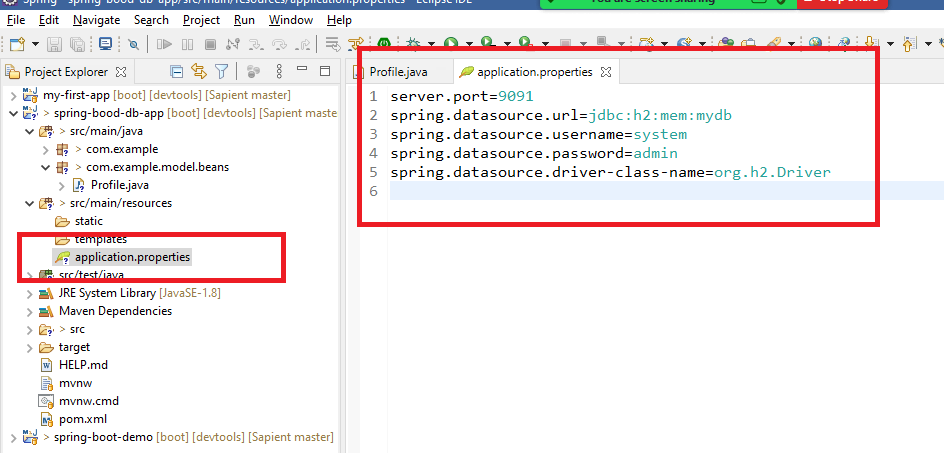
H2 database: It is an in-memory database which you can use to test your database operations without any physical database, at runtime you can still replace it by real database as your code is separated by the configurations.

We can create new project from spring intializr

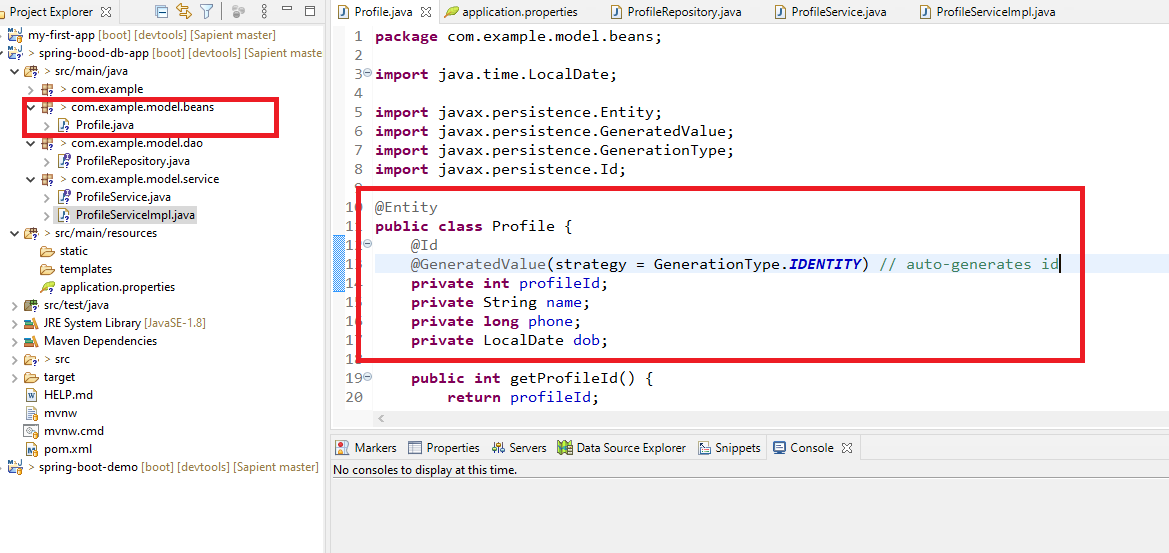


We can write H2 database datasource configurations

application.properties



Profile.java



@GeneratedValue: It auto-generates the primary key value

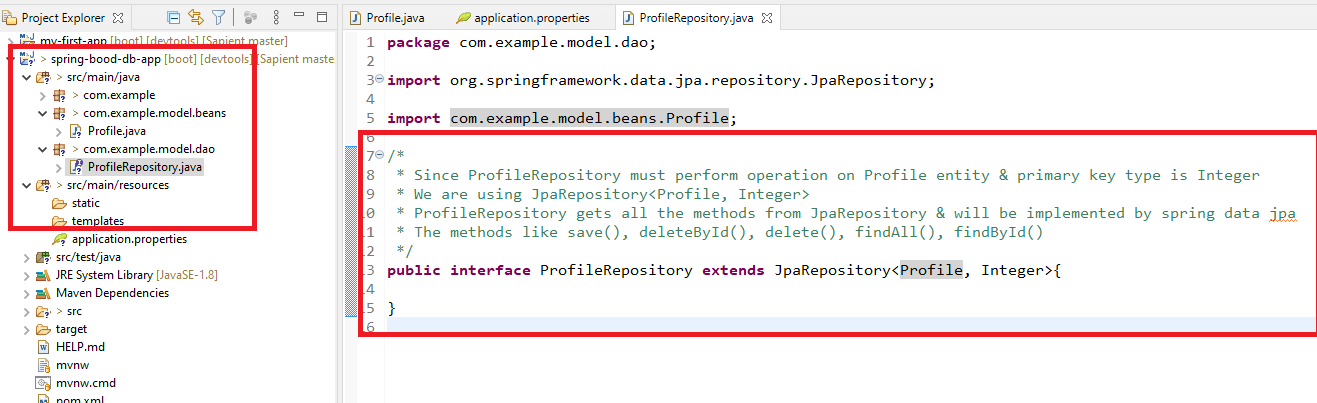
@Entity: Helps you to map the java object to the table, since we are using H2 database we don’t need to specify any @Table, @Column as we don’t have any physical database, but if you replace H2 with physical database, then we must have a table with the name profile & 4 columns profileId (primary key), name, phone & dob

Note: But you can still perform the operations on this in memory database using repository interfaces.

We will create a Repository that extends JpaRepository

The repository must be interface and we are not going to implement as Spring Data JPA does that for us

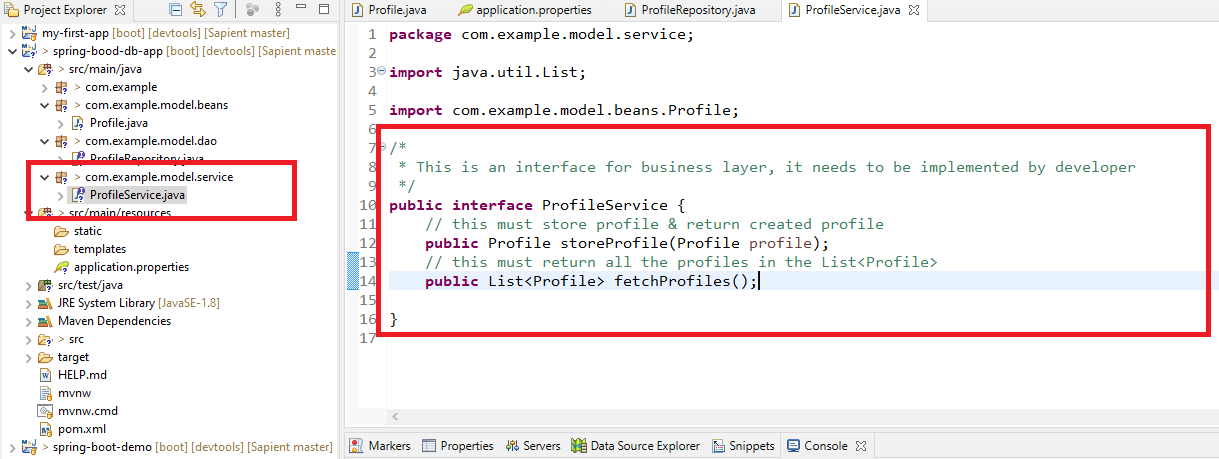
ProfileRepository.java



Here the ProfileRepository will have a proxy implementation that will be registered in the spring container, this can be supplied to the Service layer.

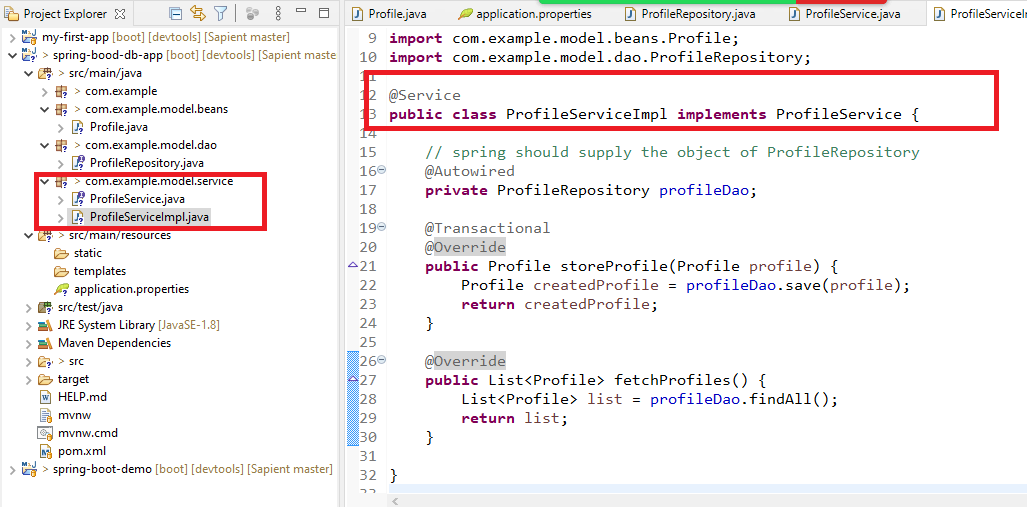
Firstly we will create an interface in the service layer & then implement it.

ProfileService.java



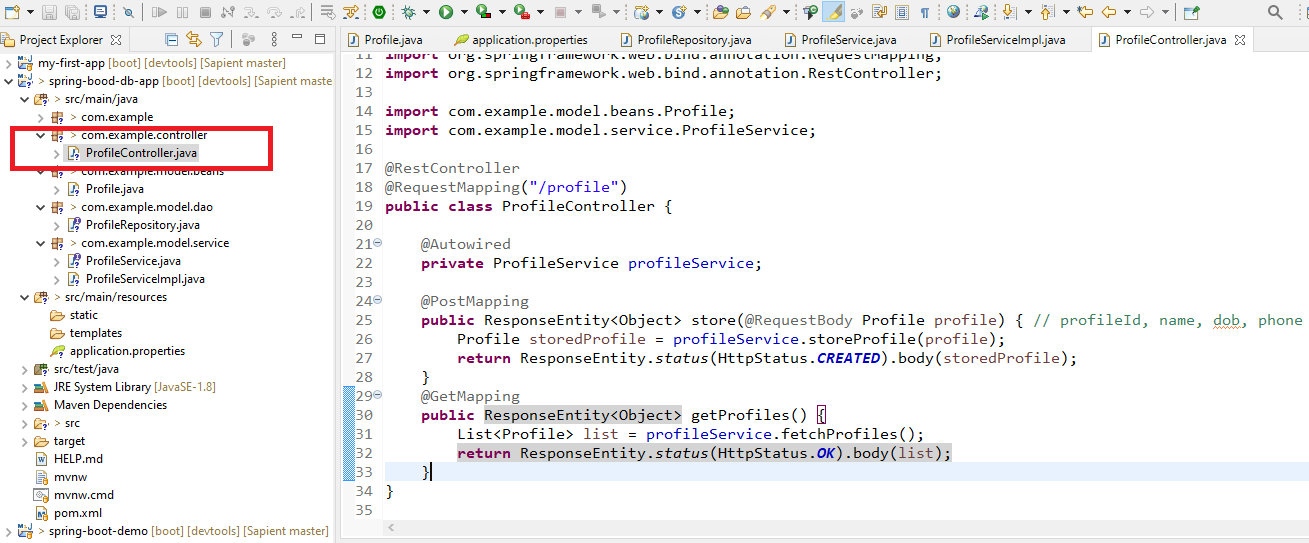
Now we have to implement these two methods in the service layer & connect this layer to the DAO layer

ProfileServiceImpl.java



Now we can connect Service layer to controller through the interface and access the store & fetch methods

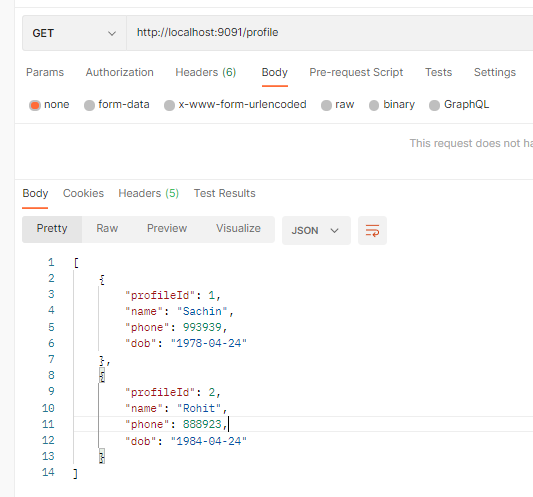
ProfileController.java



Note: This code can keep the data in the real database, but you need to mention their details in application.properties



GET:



Note: You can use MySQL driver in the classpath & provide mysql details in the application.properties so that you can make this code work in MySQL DB, but you must have a table profile with 4 columns which are: profileId, name, phone, dob