**5. Spring Boot**

1. What is an API?

Application Programming Interface, is a set of rules and protocols that allows different software applications to communicate and interact with each other. It defines the methods and data formats that applications can use to request and exchange information or perform specific actions.

APIs act as intermediaries, enabling different software systems to work together seamlessly without needing to understand each other's internal workings. They provide a standardized way for developers to access certain features or functionalities of a software application or service, abstracting away the complexity and exposing only the necessary interfaces.

1. What are http methods?

* **GET:** The GET method is used to retrieve data from the server. When you make a GET request to a resource, you are asking the server to return the data associated with that resource. GET requests should not have any side effects on the server or modify the state of the resource.
* **POST:** The POST method is used to submit data to be processed to the server. It is often used to create new resources on the server or to perform actions that have side effects, such as submitting a form or creating a new record in a database.
* **PUT:** The PUT method is used to update an existing resource or create a new resource if it does not already exist. PUT requests are idempotent, meaning that making the same PUT request multiple times should have the same effect as making it once.
* **DELETE:** The DELETE method is used to delete a specific resource from the server. DELETE requests are also idempotent, so making the same DELETE request multiple times should not have any additional side effects after the first request.
* **PATCH:** The PATCH method is used to partially update a resource. It is similar to the PUT method, but instead of replacing the entire resource, it only updates the specified fields or properties of the resource.
* **HEAD:** The HEAD method is similar to the GET method, but it only requests the headers of the resource without actually retrieving the full content. It is often used to check if a resource exists or to get metadata about the resource.
* **OPTIONS:** The OPTIONS method is used to request information about the communication options available for a resource. It can be used to determine which HTTP methods are supported by the server for a particular resource.

1. Create a spring boot application
2. Create a new controller class
3. Add a new end-point to return and string
4. Test the endpoint with postman
5. Create a Student class (attributes: id, name, date of birth, average)
6. Create a new controller class for students
7. Create a new service class for students
8. Add an endpoint to get list of students
9. Add an endpoint to get a student with id
10. Add new endpoint to create a student
11. Run mongodb as a docker container
12. Insert student received to endpoint created in step 12 to database
13. Extend step 10 and 11 to query data from database
14. Add an endpoint to delete a student with id.
15. Delete the student with id from database
16. What are http status codes

**200 OK**

**201 Created**

**204 No Content**

**400 Bad Request**

**401 Unauthorized**

**403 Forbidden**

**404 Not Found**

1. Briefly explain the meaning of following status codes

200, 201, 301, 400, 401, 403, 404, 405, 500, 501, 502, 503, 504

* **200 OK:** The request was successful, and the server has returned the requested data.
* **201 Created:** The request was successful, a new resource was created as a result, and it is included in the response.
* **301 Moved Permanently:** The requested resource has been permanently moved to a new location, and future requests should use the new URL.
* **400 Bad Request:** The server cannot understand the request due to client error, such as malformed syntax or invalid parameters.
* **401 Unauthorized:** The client needs to provide valid credentials to access the requested resource.
* **403 Forbidden:** The client does not have permission to access the requested resource.
* **404 Not Found:** The requested resource could not be found on the server.
* **405 Method Not Allowed:** The request method (GET, POST, etc.) is not allowed for the requested resource.
* **500 Internal Server Error:** A generic error message indicating that the server encountered an error while processing the request.
* **501 Not Implemented:** The server does not support the functionality required to fulfill the request.
* **502 Bad Gateway:** The server, while acting as a gateway or proxy, received an invalid response from the upstream server it accessed in attempting to fulfill the request.
* **503 Service Unavailable:** The server is temporarily unable to handle the request due to overloading or maintenance of the server. It is typically used for planned maintenance or temporary overloads.
* **504 Gateway Timeout:** The server, while acting as a gateway or proxy, did not receive a timely response from the upstream server or some other auxiliary server it needed to access in order to complete the request.

1. Using docker-compose run spring boot application and mongodb
2. Create new branch “spring-boot-app-v1” and push the project you created
3. Add your codes and answer sheet to a directory named “spring-boot-basic-training-v1” and push it to your training github repository
4. Create a pull request to main branch and assign it to your trainer