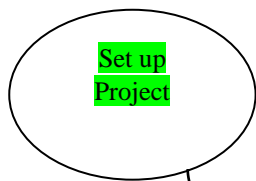


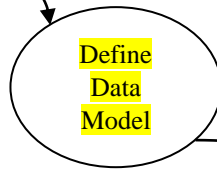
1. Create new Spring Boot project



2. Select dependencies:

- Spring Web
- Spring Data JPA
- Database Driver

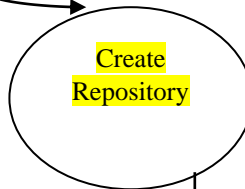
1. Create Entity Class Student.java
2. Annotate with @Entity
3. Define fields:  
id: Long  
name: String  
age: Integer
4. Add annotations: @Id  
@Column



1. Create Repository Interface StudentRepository

2. Extend JpaRepository <Student, Long>

3. Provides built-in CRUD methods



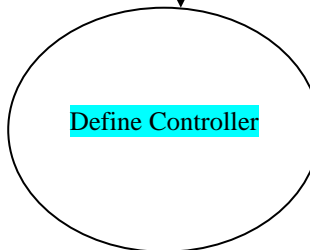
1. Create Controller Class - StudentController

2. Annotate with @RestController

3. Autowire StudentRepository

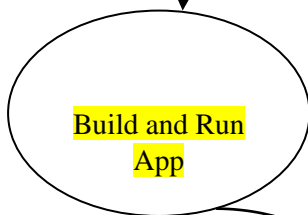
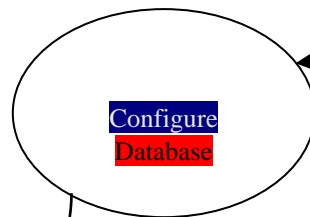
4. Define REST endpoints with annotations: - @GetMapping - @PostMapping - @PutMapping - @DeleteMapping

5. Implement CRUD methods using repository methods



1. Configure database properties in application.properties or application.yml

2. Provide details: - Database URL - Username - Password - Driver class



1. Build the application

- ./mvnw clean package

2. Run the application

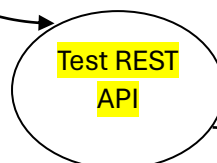
- java -jar target/Spring boot rest api

1. Use tools like Postman, curl, or web browser

2. Send HTTP requests to exposed endpoints:

- GET /students (List all students)
- GET /students/{id} (Get student by ID)
- POST /students (Create new student)
- PUT /students/{id} (Update student)
- DELETE /students/{id} (Delete student)

3. Verify responses



1. Create Dockerfile

2. Build Local Docker Image

3. Test Local Image

