

Student.java

```
package com.kinara.driver.entity;
```

```
import javax.persistence.Entity;
```

```
import javax.persistence.GeneratedValue;
```

```
import javax.persistence.GenerationType;
```

```
import javax.persistence.Id;
```

```
@Entity
```

```
public class Student {
```

```
    @Id
```

```
    @GeneratedValue(strategy = GenerationType.IDENTITY)
```

```
    private Long id;
```

```
    private String name;
```

```
    private int totalMarks;
```

```
    public Long getId() {
```

```
        return id;
```

```
    }
```

```
    public void setId(Long id) {
```

```
        this.id = id;
```

```
    }
```

```
    public String getName() {
```

```
return name;
```

```
}
```

```
public void setName(String name) {
```

```
    this.name = name;
```

```
}
```

```
public int getTotalMarks() {
```

```
    return totalMarks;
```

```
}
```

```
public void setTotalMarks(int totalMarks) {
```

```
    this.totalMarks = totalMarks;
```

```
}
```

```
public Student(Long id, String name, int totalMarks) {
```

```
    super();
```

```
    this.id = id;
```

```
    this.name = name;
```

```
    this.totalMarks = totalMarks;
```

```
}
```

```
public Student() {
```

```
    super();
```

```
}
```

```
}
```

StudentController.java

```
package com.kinara.driver.controller;
```

```
import org.springframework.beans.factory.annotation.Autowired;
```

```
import org.springframework.data.domain.Page;
```

```
import org.springframework.data.domain.Pageable;
```

```
import org.springframework.web.bind.annotation.*;
```

```
@RestController
```

```
@RequestMapping("/api/students")
```

```
public class StudentController {
```

```
private final StudentRepository studentRepository;
```

```
@Autowired
```

```
public StudentController(StudentRepository studentRepository) {
```

```
this.studentRepository = studentRepository;
```

```
}
```

```
@GetMapping
```

```
public Page<Student> getStudents(Pageable pageable) {
```

```
return studentRepository.findAll(pageable);
```

```
}
```

```
@GetMapping("/{id}")
```

```

public Student getIdByPathVariable(Long id) {

return studentRepository.findById(id)

.orElseThrow(() -> new NotFoundException("Student not found with id: " + id));

}

@GetMapping("/filter")

public Page<Student> filterStudents(@RequestParam(required = false) String name,

@RequestParam(required = false) Integer minMarks,

Pageable pageable) {

if (name != null && minMarks != null)

{

return

studentRepository.findByNameContainingIgnoreCaseAndTotalMarksGreaterThanOrEqualTo(name, minMarks, pageable);

}

else if (name != null)

{

return studentRepository.findByNameContainingIgnoreCase(name, pageable);

}

else if (minMarks != null)

{

return studentRepository.findByTotalMarksGreaterThanOrEqualTo(minMarks, pageable);

```

```
}
```

```
else
```

```
{
```

```
return studentRepository.findAll(pageable);
```

```
}
```

```
}
```

```
}
```

StudentRepository.java

```
package com.kinara.driver.repository;
```

```
import org.springframework.data.domain.Page;
```

```
import org.springframework.data.domain.Pageable;
```

```
import org.springframework.data.jpa.repository.JpaRepository;
```

```
import org.springframework.data.jpa.repository.Query;
```

```
public interface StudentRepository extends JpaRepository<Student, Long> {
```

```
    Page<Student> findByNameContainingIgnoreCase(String name, Pageable pageable);
```

```
    @Query("SELECT s FROM Student s WHERE s.totalMarks >= :minMarks")
```

```
    Page<Student> findByTotalMarksGreaterThanOrEqualTo(int minMarks, Pageable  
pageable);
```

```
}
```

SpringBootApplication

```
package com.kinara.driver.repository;
```

```

import org.springframework.data.domain.Page;

import org.springframework.data.domain.Pageable;

import org.springframework.data.jpa.repository.JpaRepository;

import org.springframework.data.jpa.repository.Query;

public interface StudentRepository extends JpaRepository<Student, Long> {

    Page<Student> findByNameContainingIgnoreCase(String name, Pageable pageable);

    @Query("SELECT s FROM Student s WHERE s.totalMarks >= :minMarks")

    Page<Student> findByTotalMarksGreaterThanOrEqualTo(int minMarks, Pageable
pageable);

}

```

Pom.Xml

```

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">

    <modelVersion>4.0.0</modelVersion>

    <groupId>Kinara</groupId>

    <artifactId>Kinara</artifactId>

    <version>0.0.1-SNAPSHOT</version>

    <packaging>war</packaging>

```

```
<name>Kinara</name>
```

```
<!-- FIXME change it to the project's website -->
```

```
<url>http://www.example.com</url>
```

```
<properties>
```

```
<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
```

```
<maven.compiler.source>1.7</maven.compiler.source>
```

```
<maven.compiler.target>1.7</maven.compiler.target>
```

```
</properties>
```

```
<dependencies>
```

```
<dependency>
```

```
<groupId>junit</groupId>
```

```
<artifactId>junit</artifactId>
```

```
<version>4.11</version>
```

```
<scope>test</scope>
```

```
</dependency>
```

```
</dependencies>
```

```
<build>
```

```
<finalName>Kinara</finalName>
```

```
<pluginManagement><!-- lock down plugins versions to avoid using Maven defaults  
(may be moved to parent pom) -->
```

```
<plugins>
```

```
<plugin>
```

```
<artifactId>maven-clean-plugin</artifactId>
```

```
<version>3.1.0</version>
```

```
</plugin>
```

```
<!-- see http://maven.apache.org/ref/current/maven-core/default-bindings.html#Plugin bindings for war packaging -->
```

```
<plugin>
```

```
<artifactId>maven-resources-plugin</artifactId>
```

```
<version>3.0.2</version>
```

```
</plugin>
```

```
<plugin>
```

```
<artifactId>maven-compiler-plugin</artifactId>
```

```
<version>3.8.0</version>
```

```
</plugin>
```

```
<plugin>
```

```
<artifactId>maven-surefire-plugin</artifactId>
```

```
<version>2.22.1</version>
```

```
</plugin>
```

```
<plugin>
```

```
<artifactId>maven-war-plugin</artifactId>
```

```
<version>3.2.2</version>
```



```
</plugin>
```

```
<plugin>
```

```
<artifactId>maven-install-plugin</artifactId>
```

```
<version>2.5.2</version>
```

```
</plugin>
```

```
<plugin>
```

```
<artifactId>maven-deploy-plugin</artifactId>
```

```
<version>2.8.2</version>
```

```
</plugin>
```

```
</plugins>
```

```
</pluginManagement>
```

```
</build>
```

```
</project>
```