

Q10) Create REST APIs with Spring boot.

**Entity Class:**

```
package com.example.demo.entity;

public class JournalEntity {
    private int id;
    private String title;
    private String content;
    public int getId() {
        return id;
    }
    public void setId(int id) {
        this.id = id;
    }
    public String getTitle() {
        return title;
    }
    public void setTitle(String title) {
        this.title = title;
    }
    public String getContent() {
        return content;
    }
    public void setContent(String content) {
        this.content = content;
    }
}
```

**Controller Class:**

```
package com.example.demo.controller;

import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

import org.springframework.web.bind.annotation.DeleteMapping;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.PutMapping;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

import com.example.demo.entity.JournalEntity;
```

```

@RestController
@RequestMapping("/journal")
public class JournalController {

    Map<Integer,JournalEntity> map = new HashMap<>();

    @GetMapping
    public List<JournalEntity> getAll(){
        if(map.containsKey(0)){
            return new ArrayList<>();
        }
        return new ArrayList<>(map.values());
    }

    @PostMapping
    public JournalEntity createNewEntity(@RequestBody JournalEntity
newEntity) {
        map.put(newEntity.getId(),newEntity);
        return map.get(newEntity.getId());
    }

    @PutMapping("id/{id}")
    public String updateEntries(@PathVariable int id, @RequestBody
JournalEntity updateData) {
        if(!map.containsKey(id)){
            return "Entry does n't exists";
        }
        map.put(id,updateData);
        return "Data update Sucessfully";
    }

    @DeleteMapping("id/{id}")
    public String deleteEntries(@PathVariable int id){
        if(!map.containsKey(id)){
            return "Data can't be deleted";
        }
        map.remove(id);
        return "Data delete Sucessfully";
    }
}

```

### **Main Java Class:**

```

package com.example.demo;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

```

```
public class DemoApplication {  
  
    public static void main(String[] args) {  
        SpringApplication.run(DemoApplication.class, args);  
    }  
  
}
```

**Output:**

```
[  
  {  
    "id": 1,  
    "title": "happy",  
    "content": "I am happy"  
  },  
  {  
    "id": 2,  
    "title": "ok",  
    "content": "I am ok"  
  },  
  {  
    "id": 3,  
    "title": "alright",  
    "content": "I am alright"  
  },  
  {  
    "id": 4,  
    "title": "good",  
    "content": "I am good"  
  }  
]
```