

```

package com.dai.webServer.Resources;

import java.net.URI; import java.util.List;

import org.json.simple.JSONObject; import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.ResponseEntity; import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PostMapping; import
org.springframework.web.bind.annotation.RequestBody; import org.springframework.web.bind.annotation.Reque
import org.springframework.web.bind.annotation.RestController; import
org.springframework.web.servlet.support.ServletUriComponentsBuilder;

import com.dai.webServer.Mqtt.Listener; import com.dai.webServer.Objects.Readings;
import com.dai.webServer.Repos.ReadingsRepository;

@RequestMapping("/") @RestController public class ReadingsResouces {

    @Autowired
    private ReadingsRepository leituraRepository;

    @GetMapping("/leitura")
    public List<Readings> retrieveAllLeitura() {
        return leituraRepository.findAll();
    }

    @PostMapping("/leitura")
    public ResponseEntity<Object> createLeitura(@RequestBody Readings leitura) {

        Readings savedLeitura = leituraRepository.save(leitura);

        URI location = ServletUriComponentsBuilder.fromCurrentRequest().path("/{id}")
            .buildAndExpand(savedLeitura.getId()).toUri();

        return ResponseEntity.created(location).build();
    }

    public String criarNaoEstatico(ReadingsResouces o, Readings b) {

        o.createLeitura(b);

        return "done";
    }
}

```

```
public ResponseEntity<Object> novaLeitura(JSONObject leitura) {

    Readings savedLeitura = leituraRepository.save(leitura);

    URI location = ServletUriComponentsBuilder.fromCurrentRequest().path("/{id}")
        .buildAndExpand(savedLeitura.getId()).toUri();
    return ResponseEntity.created(location).build();

}
}
```