Implement a "Rest"Ful Spring Boot Application Using Spring Rest, Spring Security and Spring Cache.

Aim:-

To develop a RESTFUL Spring Boot application using Spring Rest, Spring Security and Spring Cache.

Algorithm:-

Step 1 : Start

Step 2 : Open the Spring Boot application where we are going to perform Spring REST, Spring Security and Spring Cache.

Step 3 : Under Quote.java we are opening a package hello, under public Quote we get values type.

Step 4 : Under Quote.class we get the values from the user.

Step 5 : Then we have application.java, under which we have package hello.

Step 6 : Then we have value.java, where we get values and return the output to the users.

Step 7 : Under the package we import various spring framework.

Step 8 : The framework provides spring REST, Spring Security and Spring cache.

Step 9 : Stop.

Program :-

Quote.java

```
package hello;
import com.fasterxml.jackson.annotation.JsonIgnoreProperties;
@JsonIgnoreProperties(ignoreUnknown = true)
public class Quote {
    private String type;
    private Value value;
    public Quote() {
    }
    public void setType(String type) {
        this.type = type;
    }
    public Value getValue() {
        return value;
    }
    public void setValue(Value value) {
        this.value = value;
    }
    @Override
    public String toString() {
        return "Quote{" +
            "type='" + type + '\'' +
            ", value=" + value +
            '}';
    }
}
```

Value.java

```
package hello;
import com.fasterxml.jackson.annotation.JsonIgnoreProperties;
@JsonIgnoreProperties(ignoreUnknown = true)
public class Value {
    private long id;
    private String quote;
    public Value() {
    }
    public Long getId() {
        return this.id;
    }
    public String getQuote() {
        return this.quote;
    }
    public void setId(Long id) {
        this.id = id;
    }
    public void setQuote(String quote) {
        this.quote = quote;
    }
    @Override
    public String toString() {
        return "Value{" +
            "id=" + id +
            ", quote='" + quote + '\'' +
```

```
        '3';
    }
}
```

Application.java

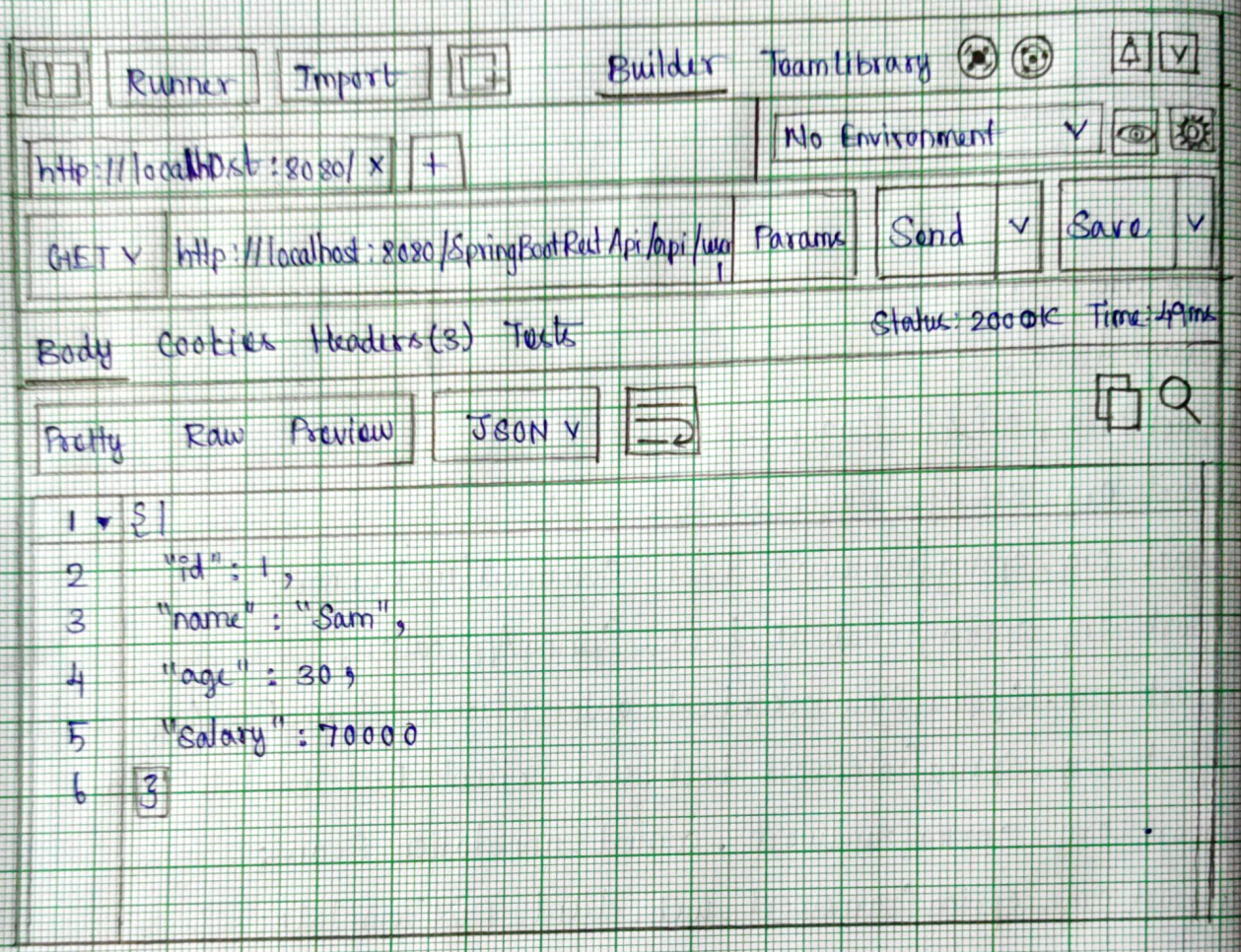
```
package hello;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import org.springframework.boot.CommandLineRunner;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.boot.web.client.RestTemplateBuilder;
import org.springframework.context.annotation.Bean;
import org.springframework.web.client.RestTemplate;
@SpringBootApplication
public class Application {
    private static final Logger log = LoggerFactory.getLogger(Application.class);
    public static void main(String args[]) {
        SpringApplication.run(Application.class);
    }
    @Bean
    public RestTemplate restTemplate(RestTemplateBuilder builder) {
        return builder.build();
    }
    @Bean
    public CommandLineRunner run(RestTemplate restTemplate) throws Exception {
```

```
    return args -> {
        Quote quote = restTemplate.getForObject(
                "http://gturnquist-quoters.cfapps.io/api/random",
Quote.class);
        log.info(quote.toString());
    };
}
}
```

Result:-

Thus the program to develop a RESTful Spring boot application using Spring REST, Spring security and Spring Cache was executed successfully.

Output:-


Runner
Import
Builder
Team Library
No Environment
http://localhost:8080/
GET
http://localhost:8080/SpringBootRestApi/api/
Params
Send
Save
Body
Cookies
Headers (3)
Tests
Status: 200 OK
Time: 49ms
Pretty
Raw
Preview
JSON
{
"id": 1,
"name": "Sam",
"age": 30,
"salary": 70000
}