**Spring Rest:**

REST stands for **RE**presentational **S**tate **T**ransfer. REST is web standards based architecture and uses HTTP Protocol for data communication. It revolves around resource where every component is a resource and a resource is accessed by a common interface using HTTP standard methods.

HTTP methods:

GET : To get a resource from server

POST : To create a new resource

PUT : To update a resource

DELETE : To delete a resource

A REST endpoint provides way to map a URI and HTTP method.

Example:

|  |  |  |
| --- | --- | --- |
| **REST Endpoint** | **HTTP Method** | **Description** |
| /customers | GET | Returns the list of customers |
| /customers/{id} | GET | Returns customer detail for given customer {id} |
| /customers | POST | Creates new customer from the post data |
| /customers/{id} | PUT | Replace the details for given customer {id} |
| /customers/{id} | DELETE | Delete the customer for given customer {id} |

**Annotations used:**

@RestController:

pring 4.0 introduced @RestController, a specialized version of the controller which is a convenience annotation that does nothing more than add the @Controller and @ResponseBody annotations. By annotating the controller class with @RestController annotation, you no longer need to add @ResponseBody to all the request mapping methods.



@PostMapping annotation that maps HTTP post request onto specific handler method.

@PostMapping is equal to **@RequestMapping(value="**"/rest/employee **",method=RequestMethod.POST)**

Example : @PostMapping(value = "/rest/employee")

@PathVariable --- to extract the data from the rest URI and map it to the method argument.

@DeleteMapping--- maps HTTP DELETE request to specific handler method.

ResponseEntity : Represents the entire HTTP response you can control anything that goes into it : status code, headers and body