

REST

REpresentational S tate T ransfer

Used for building web services

It interacts over HTTP

Uses of HTTP methods:

- 1.GET: Retrieve data
- 2.POST: Creates new data
- 3.PUT: updates the existing data
- 4.DELETE: Delete data

Data format: commonly uses JSON

/employees -> represents all employees

/employees/{id} -> represents only one employee

SOME IMP Annotations:

@RestController

@Service

@Controller

@Repository

@Autowired

@RequestBody

@RequestMapping

@GetMapping

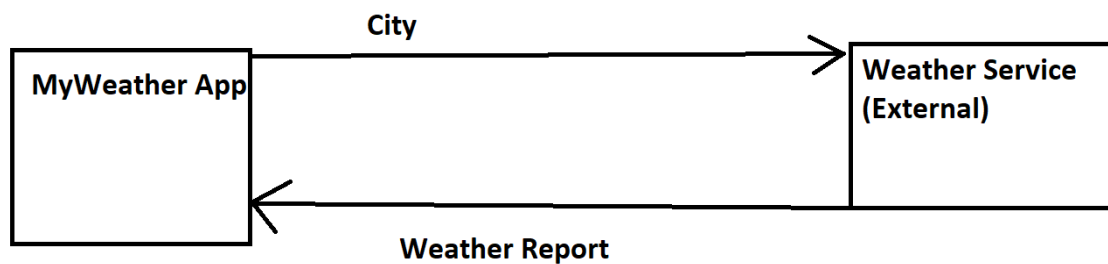
@PostMapping

@PostMapping

@DeleteMapping

Business Problem:

- Build a client app that can provide the weather report for a city
- Need to get that data from an external service



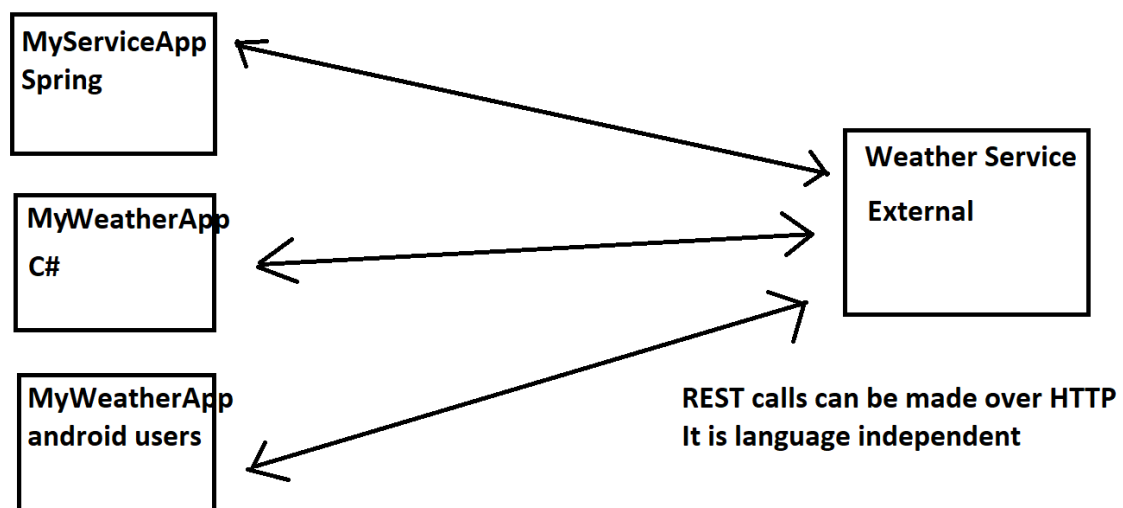
Must have questions:

1. How will we connect to weather service?
 - REST API calls over http
 - Most lightweight approach for interaction between applications
2. Which programming language we are going to use?

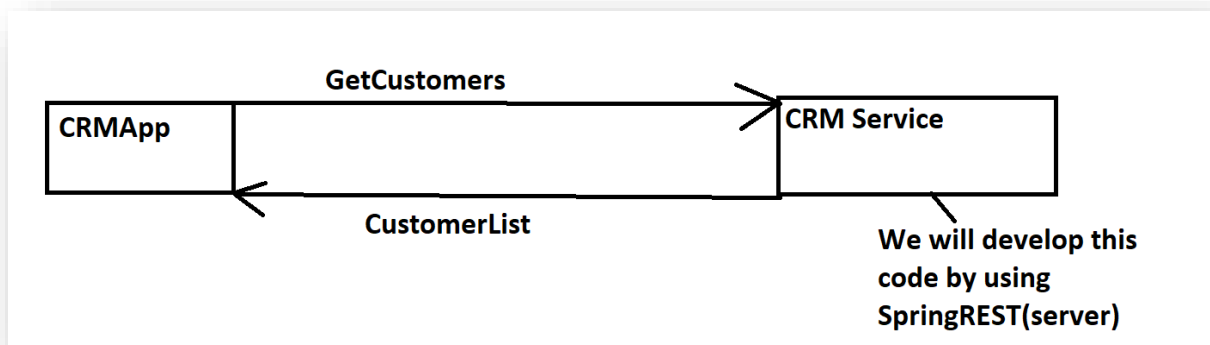
- REST is language independent

3. What is the data format?

- REST can use any data format
- Commonly uses JSON
- JavaScript Object Notation



Customer Relationship Manager(CRM) App



What is JSON?

Lightweight data format

Used for storing and exchanging the data

Language independent.....not only for JS

It is just a plain text data

Ex.

```
{  
    "id" : 1,  
    "firstname" : " Prasad" ,  
    "LastName" : "Jain" ,  
    "active" : true  
}
```

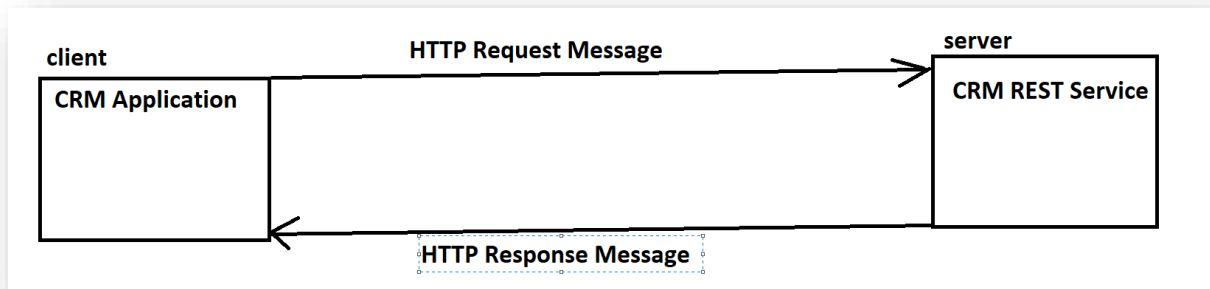
Curly braces define objects in JSON

Object members are name/value pairs

HTTP Methods:

1. POST -> create a new entity
2. GET -> reads a list of entities or a single entity
3. PUT -> Update an existing entity

4.DELETE -> delete an existing entity



HTTP Request Message:

It has 3 parts

- 1.Request line: the HTTP command
- 2.Header Variable: request metadata
- 3.Body of message: (actual payload) contents of the message

HTTP Response Message:

- 1.Response Line: status codes
- 2.Header variables: response metadata
- 3.Message body: contents of message

Http Response - Status Codes

100-199 -> Informational

200-299 -> Successful

300-399 -> Redirectional

400-499 -> client error

500-599 -> server error

Client Tool

We need a client tool to send HTTP requests to the server/REST web service

Curl, Postman