Contents

***1. Core Java***

***2. WebServices and Why WS***

***3. REST APIs and how to Automate***

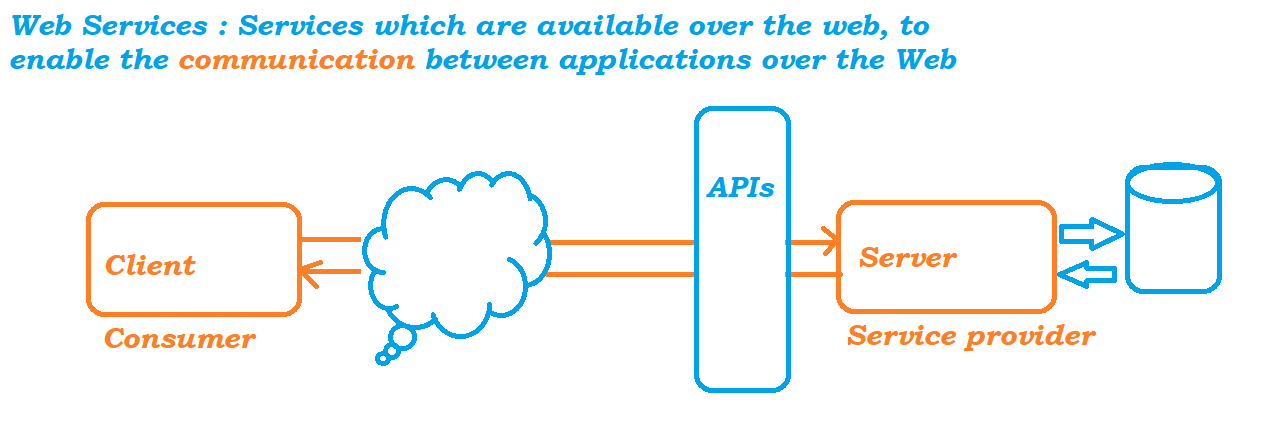
***4. Maven - Build Tool***

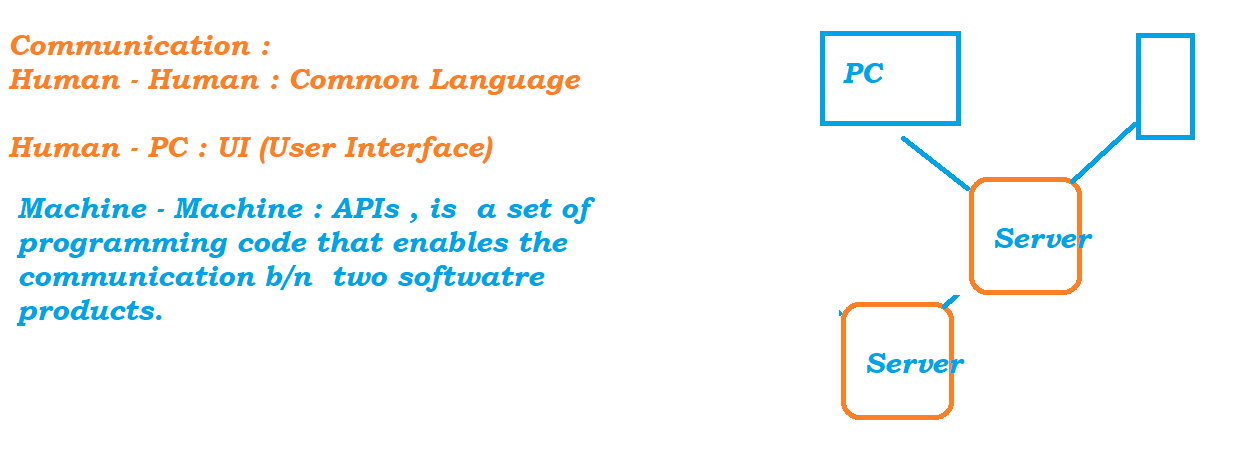
***5. TestNG - Framework***

***6. APIs - Github, Trello, Twitter***

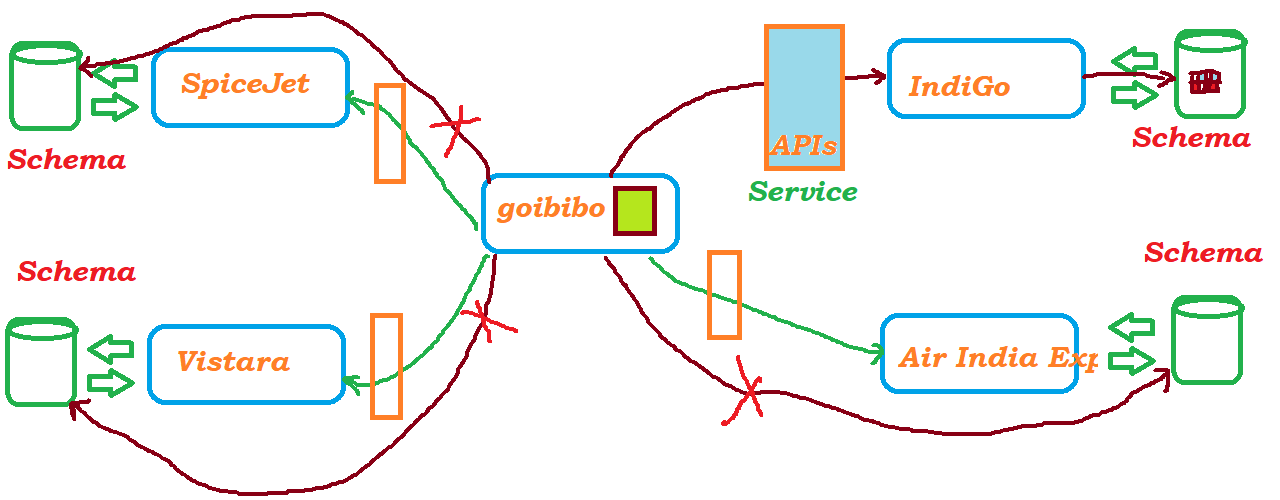
***7. POSTMAN***

***8. Rest Assured Framework***

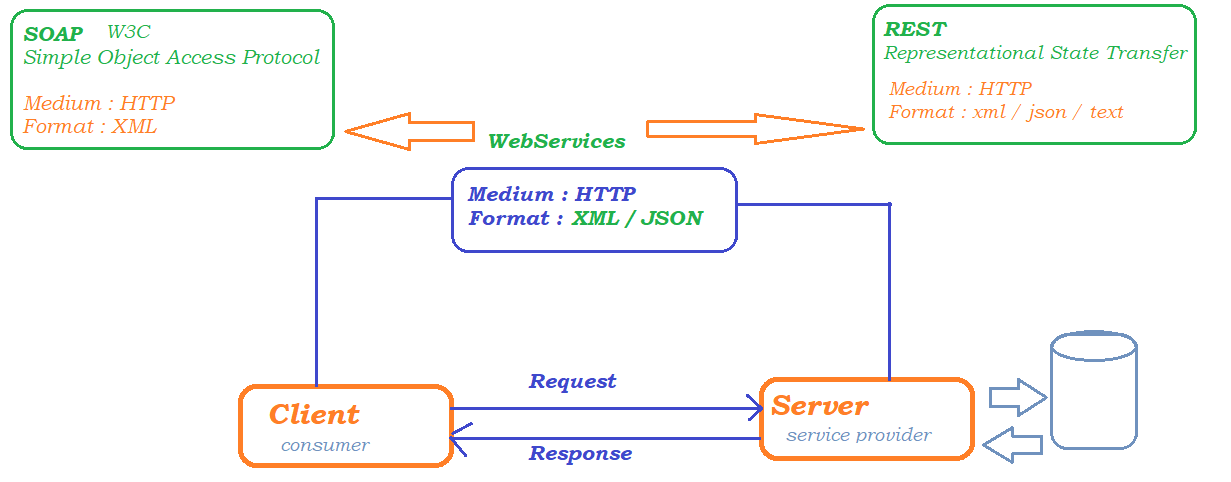
******

******

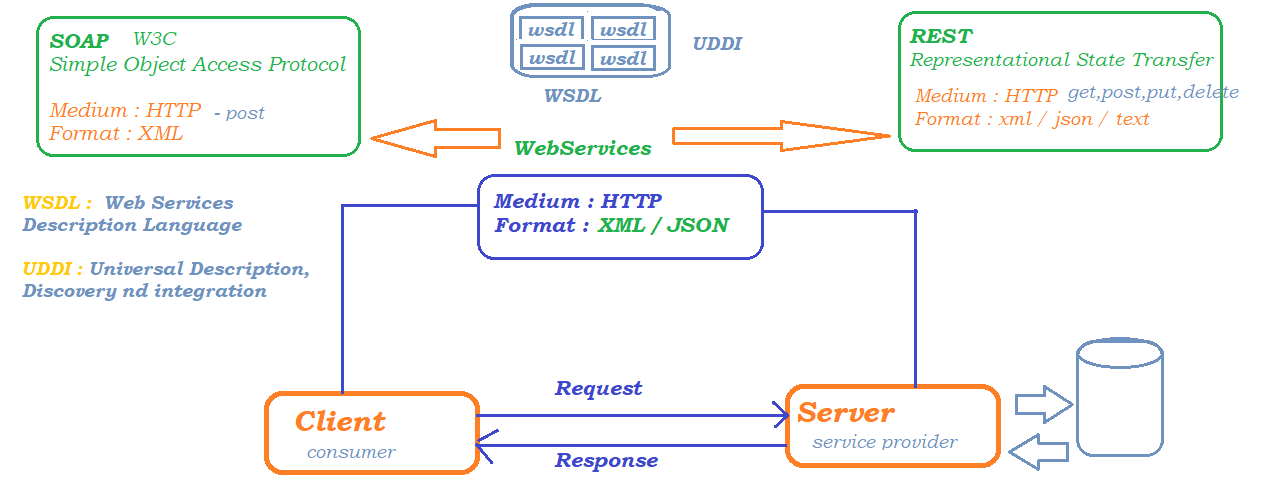
How WS Works ? Need of API in Software

******

Types of Web Services :

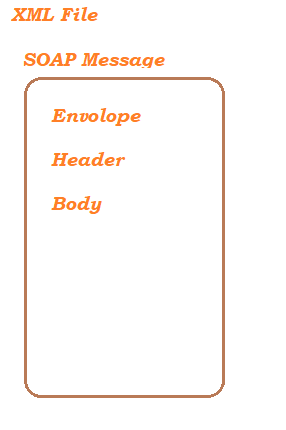


## Detailed Diagram on SOAP



## Sample SOAP Message :

<http://webservices.oorsprong.org/websamples.countryinfo/CountryInfoService.wso?op=CapitalCity>



RESTful Web Service

*REST : WebService that communicates or Exchange the messages b/n 2 Application using REST Architecture*

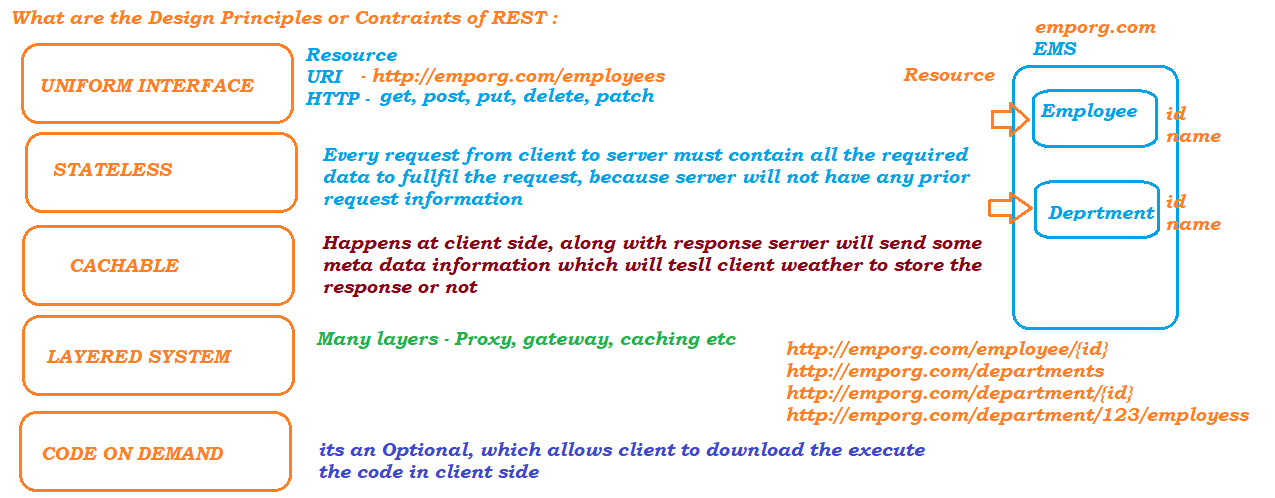
*OR Rest Principles then we call the Service as RESTful WebServies*

*- REST is nothing but Representational State Transfer its an Architecture style*

*- REST is not a protocol, there are no standard Rules, or no central body to define / control the rules*

*- REST is just a design principle*

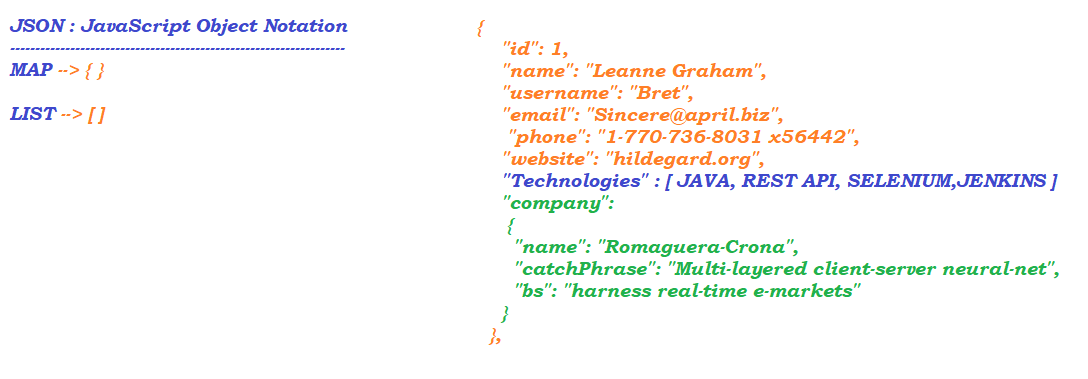
# What are the Design Principles or Constraints of REST :



Example of REST API :

<https://jsonplaceholder.typicode.com/>

JSON



PostMan

Its a third party software used to Test the REST Api’s. POSTMAN can be download and installed on a machine or it can be used as a browser plugin

STEPS to use PostMan

1. Download and install <https://www.postman.com/downloads/>

2. Double click on the postman icon

3. Once the application is launched you have to register with POSTMAN, you have to signup (you can use your gmail account to login)

# RestAPI Elements :

* **HTTP Request :**

Its a package of information requested by client to server

* + **Request Line -** specifies the method token ( GET, PUT, POST, DELETE, PATCH..)
  + **Request Header – *Optional*,** is used to store additional information while sending a request
    - **EX** : trying to access face book home page – **0 headers**
    - trying to perform some operation on facebook ,like send request or send message to friend – **additional headers**
  + **Request Body – Optional,** itis used to send additional information along with HTTP Request
  + **Authorization – Optional,** if API needs **Authorization** to fulfill the request then we have to pass the **token** or **key** provided by the Application developer
  + **Pre-Request Script, Optional,** Before sending a request if we want to execute some piece of code
* **HTTP Response :**

Its a package of information sent by server for the request made by client

* **Response Line - GET**
* **Response Headers –** contains meta data like, date, server who processed the request etc
* **Optional Body –** Optional, based on the API
* **Status Code –** APIs are validated through status code
* **Resource** – Any information stored in server and requested by client
* **Resource Identifier –** to identify the resource uniquely , its a complete URL in WebService we call it as URI
* **Representation –** Actual data returned by the server – HTML, XML, JSON
* **Representation Metadata –** Extra information sent by the server for every request made, date, server who processed the request etc
* **URI ::**
  + **scheme://domain/pathparameter**
  + **scheme://domain/queryparameter**

**scheme : http / https/ ftp ...**

**domain: name where the application is hosted or accessed**

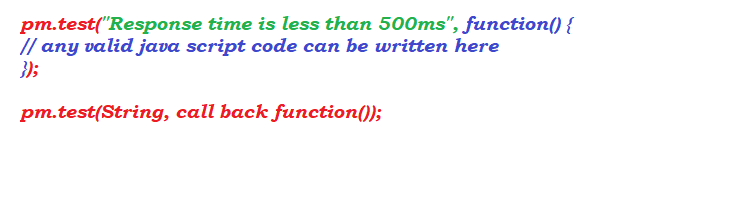
**pathparameter:** separated by / it is the actual location of the resource

**queryparameter:** separated by ? it is the data sent to server

# Using output of one API in another API Call

In a parameter Value use the global variable value in a double flower bracket EX : {{Global Data}}

**Syntax of TEST in postMan ::**



Trello API

* Register to Trello : <https://trello.com/>
* Create a sample Board -> List -> Card
* Go to API Documentation <https://developer.atlassian.com/cloud/trello/rest/api-group-actions/#api-group-actions>
* To use the APIs from trello we need to get authentication token from the Service provider
* Steps to get ID and Token
  + Click on Guides on top centre of <https://developer.atlassian.com/cloud/trello/>
  + Scroll down and click on trello REST APIs
  + click on API Introduction
  + click on new Pop-ups
  + Enter all the mandatory details
  + Click on API Key from the left panel,
  + Click on generate new API Key
  + Copy the newly created key and secrete key to notepad
  + On the same page click on token
  + Click on Allow and copy the token to notepad
* 