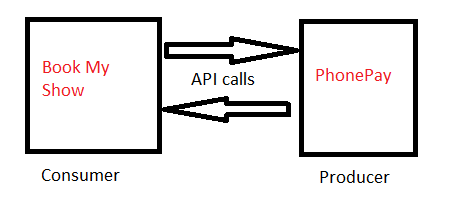
**REST API-INTRODUCTION**

**End user🡪Gets all required information**

**Developer🡪Designs app to access data in database via APIs**

**API🡪[Application Programming Interface] connects with database to retrieve required information**

**Database🡺Offers third party services**

****

**APIs are a set of functions and procedures which allow for the creation of applications that access data and features of other applications and services.**

**Purpose:Interoperability**

|  |  |
| --- | --- |
| **SOAP** | **REST** |
| **Simple object access protocol** | **Representational State Transfer** |
| **XML based message protocol** | **Architectural style** |
| **WSDL required** | **No such restriction** |
| **RPC are used to get services** | **We use url paths to get services** |
| **Does not return human readable results** | **Returns JSON results which are in human readable form** |
| **Heavy weight** | **Light weight** |
| **More bandwidht** | **Less Band width** |
| **More secure** | **Less secure** |
| **Uses HTTP,FTP,SMTP etc** | **Mostly HTTP** |
| **Less performance** | **More performance** |
| **No limit on data length** | **Limit on data length** |

**CRUD operations and HTTP verbs**

**1.GET🡪Getting information/resource🡺Retrieve**

**2.POST🡪Create a new resource🡪Create**

**3.PUT🡪To update resource🡪Update**

**4.PATCH🡪Partial updation-🡪Update**

**5DELETE🡪Delete a resource🡪Delete**

**DUMPS AND LOADS**

**Python provides a buit in package called json that is used to work with JSON data.And this package includes the methods like dumps and loads**

**Python dictionary🡪JSON [Javascript object notation]🡺dumps()**

**JSON🡪Python dictionary🡪loads()**

**Ex1:**

**import json**

**d={'name':'shreenath','marks':90} #dictionary**

**j=json.dumps(d)**

**print(j)**

**print(type(j))**

**o/p**

**{"name": "shreenath", "marks": 90}**

**<class 'str'>**

**Ex2:**

**import json**

**j='{"name":"shreenath","marks":90}'**

**d=json.loads(j)**

**print(d)**

**print(type(d))**

**o/p**

**{'name': 'shreenath', 'marks': 90}**

**<class 'dict'>**

**django-admin startproject withoutrest**

**cd withoutrest**

**py manage.py startapp myApp**

**In settings.py add the application**

**views.py**

**from django.shortcuts import render**

**from django.http import HttpResponse,JsonResponse**

**import json**

**def emp\_data\_view(request):**

**emp\_data={'eno':100,'ename':'Shreenath','esal':60000}**

**response='<h1>Employee Num:{}<br> Employee Name:{} <br> Employee Sal:{}</h1>'.format(emp\_data['eno'],emp\_data['ename'],emp\_data['esal'])**

**return HttpResponse(response)**

**def emp\_json\_data\_view1(request):**

**emp\_data={'eno':200,'ename':'Shreya','esal':70000}**

**json\_data=json.dumps(emp\_data)**

**return HttpResponse(json\_data,content\_type='application/json')**

**def emp\_json\_data\_view2(request):**

**emp\_data={'eno':300,'ename':'Shreeram','esal':80000}**

**return JsonResponse(emp\_data)**

**urls.py**

**from django.contrib import admin**

**from django.urls import path**

**from myApp.views import \***

**urlpatterns = [**

**path('admin/', admin.site.urls),**

**path('response1/',emp\_data\_view),**

**path('response2/',emp\_json\_data\_view1),**

**path('response3/',emp\_json\_data\_view2)**

**]**

**Sending request from command prompt**

**pip install httpie**

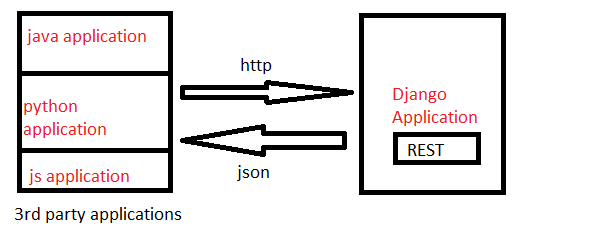
**cmd prompt screen1: run djangio server**

**cmd prompt screen 2:** **cd djangoapplications**

**cd withoutrest**

**http** [**http://127.0.0.1:8000/3ndpoint/**](http://127.0.0.1:8000/3ndpoint/)

**Communication of Django with partner applications**

****

**To send http request from a python application we use a module called**

**requests.To install🡪pip install requests**