**Rest Full Web Services & Micro services:**

**REST API & Micro Services:**

1) Type of Application.

2) What is distributed application?

3) Why we need to go for distributed applications

4) Distributed Technologies

5) Provider Development

6) Testing Using Postman

7) Swagger

8) Consumer Development

9) Exception Handling

10) Micro Services

11) Service Registry

12) Admin Server

13) Zipkin Server

14) API Gateway

15) Ribbon for lbr

16) Feign Client

17) Circuit Breaker

18) Kafka

19) Redis

20) Config Server

21) Security

**Types of Application:**

1. Standalone Application( Only one user can access at a time)
2. Web Application( Multiple users can access at a time)
3. Distribution Application (Web Services).

**What is Distribution Application?**

1. If one application communicate another application then it is called Distribution application.

Ex: 1. Make my Trip Application----->IRTC Application

2. Passport ------------> aadhar

3. gpay ----------------> banking

**Why to develop distribution application?**

=> for code reusability

=> for loosely coupling.

**Distribution Application Architecture:**

=> In distribution applications, 2 actions will be available

1) Provider: the app which is giving to service to other application

Ex: IRTC.

2) Consumer: the app which is accessing the service to other application.

Ex: MakeMyTrip, Yatra …

**What is Interoperability?**

=> Not language specification

=> Not platform specification

=> Irrespective of platform and language if two apps are communicating then those are called as interoperability applications.

Java -------------> .net

Java--------------> python

Note: To achieve intereoperability we will use xml/json format to transfer data from one application to another application.

=> XML & JSON formats are universal, all languages will understand these formats.

Note: To achieve intereoperability we will use xml/json format to transfer data from one application to another application.

=> XML & JSON formats are universal, all languages will understand these formats.

**Distribution Technologies:**

1) CORBA

2) RMI

3) EJB

4) SOAP Web services (Outdated)

- JAX-RPC

- JAX-WS

5) Restful Services (Trending)

- JAX-RS (outdated)

- Spring REST (trending)

**JSON:**

=> Json stands for **Java Script Object Notation**.

=> In Distributed applications we will use JSON as a mediator between Consumer & Provider.

=> JSON is intereoperable

=> JSON is very light weight

=> JSON will represent data in key-value format.

Syntax:

{

"Id": 101,

"Name”: "Ravi",

"Gender": "Male"

}

=> To convert java object to json data and json data to java object we will use "Jackson" API.

=> Jackson is a third party library.\

=> Jackson API provided "Object Mapper" class to perform conversions

Jackson API

Java obj <---------------------> json data

ObjectMapper mapper = new ObjectMapper ( );

writeValue(object); // convert java obj to json

readValue (String file); // convert json to java obj

**Java with Jackson API Example:**

<dependency>

<groupId>com.fasterxml.jackson.core</groupId>

<artifactId>jackson-databind</artifactId>

<version>2.16.1</version>

</dependency>

public class Customer {

private Integer cid;

private String cname;

private String cemail;

// setters & getters

}