1.JSON

JSON stands for JAVASCRIPT OBJECT NOTATION(JSON):--

* JavaScript Object Notation (JSON) is a standard text-based format for representing structured data based on JavaScript object syntax.
* It is commonly used for transmitting data in web applications (e.g., sending some data from the server to the client, so it can be displayed on a web page, or vice versa).
* JSON is a text format for storing and transporting data.
* JSON is "self-describing" and easy to understand.
* JSON is a lightweight data-interchange format. JSON is plain text written in JavaScript object notation
* JSON is used to send data between computers. JSON is language independent.
* This example is a JSON string:

'{"name":"John", "age":30, "car":null}'

* It defines an object with 3 properties:
* Name
* Age
* car
* The property has a value.
* If you parse the JSON string with a JavaScript program, you can access the data as an object:

let personName = obj.name;

* let personAge = obj.age;

2.JSON Array

* JSON is a lightweight data-interchange format. JSON is plain text written in JavaScript object notation
* JSON is used to send data between computers JSON is language independent
* Arrays in JSON are almost the same as arrays in JavaScript
* In JSON, array values must be of type string, number, object, array, boolean or null.
* In JavaScript, array values can be all of the above, plus any other valid JavaScript expression, including functions, dates, and undefined.
* You can create a JavaScript array from a literal:

Example

myArray = ["Ford", "BMW", "Fiat"];

3.Controller and RestController

* The @Controller is a common annotation which is used to mark a class as Spring MVC Controller The @RestController is a special controller used in RESTFul web services and the equivalent of @Controller + @ResponseBody.
* The @RestController is relatively new, added only on Spring 4.0 but @Controller is an old annotation, exists since Spring started supporting annotation, and officially it was added on Spring 2.5 version.
* The @Controller annotation indicates that the class is a “Controller” e.g. a web controller while the @RestController annotation indicates that the class is a controller where @RequestMapping methods assume @ResponseBody semantics by default i.e. servicing REST API.
* The @Controller is a specialization of @Component annotation while @RestController is a specialization of @Controller annotation. It is actually a convenience controller annotated with @Controller and @ResponseBody

4.Soap and Restful Webservices

SOAP (Simple Object Access Protocol) is a standards-based web services access protocol that has been around for a long time. REST (Representational State Transfer) is another standard, made in response to SOAP's shortcomings. It seeks to fix the problems with SOAP and provide a simpler method of accessing web services.

Soap webservices:

* SOAP stands for Simple Object Access Protocol.
* SOAP is a protocol.
* SOAP can't use REST because it is a protocol.
* SOAP uses services interface to expose the business logic.
* JAX-WS is the java API for SOAP web services.
* SOAP defines standard to be strictly followed.
* SOAP requires more bandwidth and resource than REST.
* SOAP defines its own security.
* SOAP permits XML data format only.
* SOAP is less preferred than REST

Restful Webservices:

* REST stands for Representational State Transfer
* REST is an architectural style.
* REST can use SOAP web services because it is a concept and can use any protocol like HTTP, SOAP.
* REST uses URI to expose business logic.
* JAX-RS is the java API for RESTful web services.
* REST does not define too much standards like SOAP.
* REST requires less bandwidth and resource than SOAP.
* RESTful web services inherits security measures from the underlying transport.
* REST permits different data format such as Plain text, HTML, XML, JSON etc.
* REST more preferred than SOAP.

5.What is the difference between Web Application and WebService Application ?

Web Services

* Web services are a type of API, which must be accessed through a network connection.
* All Web services are APIs.
* It provides supports only for the HTTP protocol.
* Web service supports only XML.
* Web Services can be hosted on IIS.
* It is not open source, however can be devoured by any customer that comprehends xml.
* It is open source and also ships with .NET framework
* It doesn’t have lightweight design, needs a SOAP convention to send or receive data over the system.
* Webservice is used for REST,SOAP and XML-RPC for communication.

Web Application

* APIs are application interfaces, implying that one application can communicate with another application in a standardized manner.
* APIs are not web services.
* It provides support for the HTTP/s protocol: URL Request/Response Headers, and so on.
* API supports XML and JSON.
* Web API can be hosted only on IIS and self.
* It has a light-weight architecture furthermore, useful for gadgets which have constrained transmission capacity like smart phone.
* API is used for any style of communication

6.Whats is DDL?

A data definition language (DDL) is a computer language used to create and modify the structure of database objects in a database. These database objects include views, schemas, tables, indexes. ddl. auto is a hibernate configuration property. It is used to validate and exports schema DDL to the database when the SessionFactory is created. If we want to make use of it, we should have to pass the appropriate values to the hibernate.

7. JPARepository, PagingAndSortingRepository and CRUDRepositoy?

* CrudRepository mainly provides CRUD functions. PagingAndSortingRepository provides methods to do pagination and sorting records.
* JpaRepository provides some JPA-related methods such as flushing the persistence context and deleting records in a batch.
* Because of the inheritance mentioned above, JpaRepository will have all the functions of CrudRepository and PagingAndSortingRepository. So if you don't need the repository to have the functions provided by JpaRepository and PagingAndSortingRepository , use CrudRepository

8. what are mappings one-to-one, one-to-many, many-to-one, many-to-many?

* One-to-one relationships associate one record in one table with a single record in the other table.
* One-to-many relationships associate one record in one table with many records in the other table. ... It is not possible to have two foreign keys for each table as it will be impossible to create records.
* Many-to-many: Multiple records in one table are related to multiple records in another table.
* Many to one:multiple record in one table are related to single records in another table.

9.what are relations in parent child table is-a, uses-a, has-a

Is-A Relationship in Java

In Java, an Is-A relationship depends on inheritance. Further inheritance is of two types, class inheritance and interface inheritance. It is used for code reusability in Java. For example, a Potato is a vegetable, a Bus is a vehicle, a Bulb is an electronic device and so on. One of the properties of inheritance is that inheritance is unidirectional in nature. Like we can say that a house is a building. But not all buildings are houses. We can easily determine an Is-A relationship in Java. When there is an extends or implement keyword in the class declaration in Java, then the specific class is said to be following the Is-A relationship.

Has-A Relationship in Java

In Java, a Has-A relationship is also known as composition. It is also used for code reusability in Java. In Java, a Has-A relationship simply means that an instance of one class has a reference to an instance of another class or an other instance of the same class. For example, a car has an engine, a dog has a tail and so on. In Java, there is no such keyword that implements a Has-A relationship. But we mostly use new keywords to implement a Has-A relationship in Java.

10.what is transient in jpa

@Transient annotation in JPA or Hibernate is used to indicate that a field is not to be persisted or ignore fields to save in the database. ... It is used to annotate a property or field of an entity class, mapped superclass, or embeddable class.

Transient is a variables modifier used in serialization. At the time of serialization, if we don't want to save value of a particular variable in a file, then we use transient keyword. When JVM comes across transient keyword, it ignores original value of the variable and save default value of that variable data type.

What is a response entity?

ResponseEntity represents the whole HTTP response: status code, headers, and body. As a result, we can use it to fully configure the HTTP response. If we want to use it, we have to return it from the endpoint; Spring takes care of the rest. ResponseEntity is a generic type.

Response Body

The REST API sends a response header and response body in JSON format with information about the success or failure of the REST API call.

The response header can contain the following information:

server

content-type

content-language

content-length

date

connection

transfer-encoding

severity

errorCode

description

logFile

* The response body can contain information about the output and additional output ports. If the output type was file, the response body contains a path to the output. If the output type was buffer, the response body contains the buffer contents. The response body can also contain a success or failure code and message
* The response body consists of the resource data requested by the client. In our example, we requested the book's data, and the response body consists of the different books present in the database along with their information.