**Date : 20/10/2020**

**Spring Boot 7AM**

**Mr. RAGHU**

**-----------------------------------------------------**

\*)Joins:-

This concept is given by database, to fetch data using SELECT (single SELECT query)

from multiple connected tables.

SQL Joins:-

a. INNER JOIN | JOIN

Fetching data from both tables which are connected with each other.

b. OUTER JOIN

LEFT OUTER JOIN | LEFT JOIN

Fetching all left side table rows and only connected right side table rows.

RIGHT OUTER JOIN | RIGHT JOIN

Fetching all right side table rows and only connected left side table rows.

FULL OUTER JOIN | FULL JOIN

Fetching all connected and non-connected rows from both tables.

c. Cross Join (mxn)

d. Equal Join (m.idType=n.idType)

e. Self Join (m.id=m.fid)

..etc

SQL Syntax:

SELECT <T1>.<column>, <T2>.<column>

FROM <Table#1> <T1>

[JOIN TYPE]

<Table#2> <T2>

ON <T1>.<PK> = <T2>.<FK>

WHERE <CONDITION>;

**Ex:**

SELECT V.VNAME,C.CODE, C.BID\_AMT

FROM VENDOR V

INNER JOIN

CONTRACT C

ON V.VID = C.VID\_FK

WHERE C.CODE='C1';

=> SQL Joins works for any multiplicity.(same syntax, only result gets modified).

=> Based on output required we need to choose Join Type.

--------------------------------------------------------------------------------------

**Joins using Spring Boot Data JPA**

HQL/JPQL Syntax:-

SELECT <p>.<variable>,<c>.<variable>

FROM <ParentClass> <p>

[JOIN TYPE]

<p>.<hasAVariable> as <c>

WHERE <condition>;

Ex: B---<>A

class A {

int id;

String code;

}

class B {

int id;

String name;

A oa;//HAS-A

}

--Ex HQL Joins:-

SELECT ob.name,oa.code

FROM B ob

INNER JOIN

ob.oa as oa

WHERE oa.id=10;

========================================================================