**API End Points and Communication**

**Day 7: 7 March 28**

**Spring with ORM tool ie JPA**

*SQL*

*Select \* from employee, \* means all column and employee is a table name. sql is not a case sensitive.*

*Using sql if we want to retrieve particular column information may be 1 column or more than one column*

*select name from employee; name is column*

*select name, salary from employee; name and salary is column name*

*JPQL*

*select emp from Employee emp Employee is entity class name, case sensitive and emp is object name.*

*select emp.name from Employee emp; here emp is object and name is variable of Employee class*

*select emp.name, emp.salary from Employee emp; here emp is object and name and salary are variable of Employee class*

*JPA relationship*

*4 types of relationship*

*One to many @OneToMany*

*One to one @OneToOne*

*Many to one @ManyToOne*

*Many to many @ManyToMany*

*Database level one to many relationship*

*Trainer table*

*TId(PK) Tname tech*

*1 Raj Java*

*2 Steven Python*

*@Entity*

*Class Trainer*

*@Id*

*Tid 🡪*

*TName*

*Tech*

*@OneToMany*

*@JoinColumn(name=”tid”)-🡪link to FK*

*List<Student> listOfStd;*

*Student table*

*Sid(PK) SName Age tid(FK)*

*100 Leena 21 1*

*101 Veena 22 1*

*102 Meena 23 2*

*@Entity*

*Class Student*

*@Id*

*sid*

sname

age

tid

we create the table trainer as well as student

create table trainer(tid int primary key,tname varchar(30), tech varchar(30));

create table student(sid int primary key,sname varchar(30), age int, tid int, foreign key(tid) references trainer(tid));

inner join using sql

select t.tname,t.tech,s.sname from trainer t inner join student s on t.tid=s.tid;

it retrieve common record present in both the tables.

inner join using jpql

select t.tname,t.tech,s.name from Trainer t inner join Student s on t.tid=s.tid;