**SPRING BOOT**

**JPA TABLE RELATION**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*ONE TO ONE MAPPING\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UNI directional using OneToOne direction:-

OWNING SIDE:-

Annotation:-

@OneToOne(cascade = cascadeType.All) = cascade used kupeleka data za referenced table (child) ktk owing table

@JoinColumn(name = "fk\_column name", referencedColumnName = "referenced colum")

private Entity entity; =this is reference side entity.

REFERENCING SIDE:-

Annotation:-

@OneToOne(mappedBy = Entity)

private Entity entity; =this is owing side entity.

Fetch:-

Egar = it contain all table's columns of owing side and referencing side columns

Lazy = it cotain only table's column in owing side, until you specificaly ask for referencing column(\* but some time

you have to remove toString method By @ToString(exlude = "referenced\_table\_name"))

BI directional relational = inafanya table moja lenye mchanganyiko wa attribute za table zote mbili.

@OneToOne(mapedBy = "referenced table name ")

private Entity entity;

\*\*@OneToOne(cascade = cascadeType.ALL,

Fetch = fetchType.EGAR,

Optional = false )

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*JPA\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

OneToMany relation

SECURITY JWT

1. Tunatengenza table mbili ya USER & ROLE than tuna zi break kupata table ya tatu USER\_ROLE
2. Tuna tengeza APIs mbili za ku GET kuangalia uthentication & authorization.
3. Tuna add two dependency SECURITY & JSONWEBTOKEN(JWT)