CADT
IDT
វិទ្យាស្ថានបច្ចេកវិទ្យាឌីជីថល
Institute of Digital Technology
DEPARTMENT OF COMPUTER SCIENCE

Database Systems

05 Data Definition Language (DDL)


Kak Soky, PhD

soky.kak@cadt.edu.kh

085683680

Constraints

SQL allows you to define constraints on columns and tables. Constraints give you as much control over the data in your tables as you wish. If a user attempts to store data in a column that would violate a constraint, an error is raised.

Constraints Types:

- **NOT NULL:** prohibits a database value from being null.
- **Unique:** prohibits multiple rows from having the same value in the same column or combination of columns but allows some values to be null.
- **Primary key:** combines a NOT NULL constraint and a unique constraint in a single declaration. It prohibits multiple rows from having the same value in the same column or combination of columns and prohibits values from being null.
- **Check:** requires a value in the database to comply with a specified condition.
- **Foreign key:** requires values in one table to match values in another table.

You can define constraints syntactically in two ways:

- As a part of the definition of an individual column or attribute. This is called **inline** specification.
- As a part of the table definition. This is called **out-of-line** specification.

Notes:

- ✓ NOT NULL constraints must be declared inline. All other constraints can be declared either inline or out of line.
- ✓ The constraint on a combination of columns must be declared out of line.
- ✓ You cannot designate the same column or combination of columns as both a primary key and a unique key.

In Line Specification
[Constraint const_name] const_type [const_specifications] ***Note:*** In this specification const_type doesn't specified for foreign key constraint.

Out Of Line Specification
[Constraint const_name] const_type (column[s]) [const_specifications]

Constraint const_name specifies a name for the constraint. If you omit this identifier, then Oracle Database generates a name.

Constraint name clarifies error messages and allows you to refer to the constraint when you need to change it.

Not Null Constraint

It must be declared in line.

In Line Specification
[Constraint const_name] Not Null

Unique Constraint

In Line Specification
[Constraint const_name] Unique

Out of Line Specification
[Constraint const_name] Unique (Column[s])

Primary Key Constraint

In Line Specification
[Constraint const_name] Primary Key

Out of Line Specification
[Constraint const_name] Primary Key (Column[s])

Check Constraint

In Line & Out Of Line Specifications
[Constraint const_name] ckeck (Boolean Expression)

Foreign Key Constraint

A foreign key constraint specifies that the values in a column (or a group of columns) must match the values appearing in some row of another table. This maintains the referential integrity between two related tables.

When Deleting or updating a referenced row, Oracle allows you to handle that as well. There are two options:

- **CASCADE** if you want Oracle to remove/update dependent foreign key values.
- **SET NULL** if you want Oracle to convert dependent foreign key values to NULL.

In Line Specification
[Constraint const_name] references ReferencedTable (ReferencedCol) [ON Delete set Null | CASCADE] [ON UPDATE set NULL | CASCADE]

Out Of Line Specification
[Constraint const_name] foreign key (Column[s]) references ReferencedTable (ReferencedCol[s]) [ON Delete set Null | CASCADE] [ON UPDATE set NULL | CASCADE]

If you omit ON DELETE or ON UPDATE clause, then Oracle does not allow you to delete or update referenced key values in the parent table that have dependent rows in the child table.

Note: Foreign key column data type must be as same as referenced column.

Example:

```
Create table EMPLOYEE (
SSN varchar2(9) Not Null constraint PK Primary Key,
FName varchar2(10) constraint FN_NN Not Null,
LName varchar2(10),
Salary number (6,2) check (Salary > 2000),
MgrSSN varchar2(9) constraint fk references EMPLOYEE (SSN),
Hired_Date Date,
DNo INTEGER,
Constraint fk2 foreign key (DNo) references DEPARTMENT (DNUM)
);
```

Modifying Tables

Alter Table command allow you to alter the definition, or structure, of the table, such you can:

- Add columns.
- Remove columns.
- Add constraints.
- Remove constraints.
- Change default values.
- Change column data types.
- Rename columns.
- Rename tables.

Add Column

ALTER TABLE TName

ADD ColName DataType [Default value] [Constraints]**;**

Add Columns

```
ALTER TABLE TName
ADD(
Col1Name DataType [Default value] [Constraints],
Col2Name DataType [Default value] [Constraints]
… );
```

Change Column Data Type

```
ALTER TABLE TName
MODIFY  ColName newDataType;
```

Note: You can change the data type of any column if all rows of the column contain nulls.

Specify Column Default Value

```
ALTER TABLE TName
MODIFY  ColName DEFAULT DefValue;
```

To discontinue previously specified default values, so that they are no longer automatically inserted into newly added rows, replace the values with NULL.

Remove Default Value

```
ALTER TABLE TName
MODIFY  ColName DEFAULT NULL;
```

Rename Column

```
ALTER TABLE TName
Rename Column oldName to  newName;
```

Remove Column

ALTER TABLE TName

Drop Column ColName [CASCADE]**;**

CASCADE: if you want to drop everything that depends on the column.

Add Out Of Line Constraint

ALTER TABLE TName

Add OutOfLineConstraint **;**

To add a not-null constraint, which cannot be written as out of line constraint, use this syntax:

Add Not Null Constraint

ALTER TABLE TName

Modify ColName **Not Null;**

Rename Constraint

ALTER TABLE TName

RENAME CONSTRAINT OldName **TO** NewName;

Remove Constraint

ALTER TABLE TName

DROP CONSTRAINT name [**CASCADE**]**;**

Remove Not Null Constraint

ALTER TABLE TName

MODIFY ColName **NULL;**

ReName Table

ALTER TABLE TName

RENAME TO NewName**;**

Remove Table

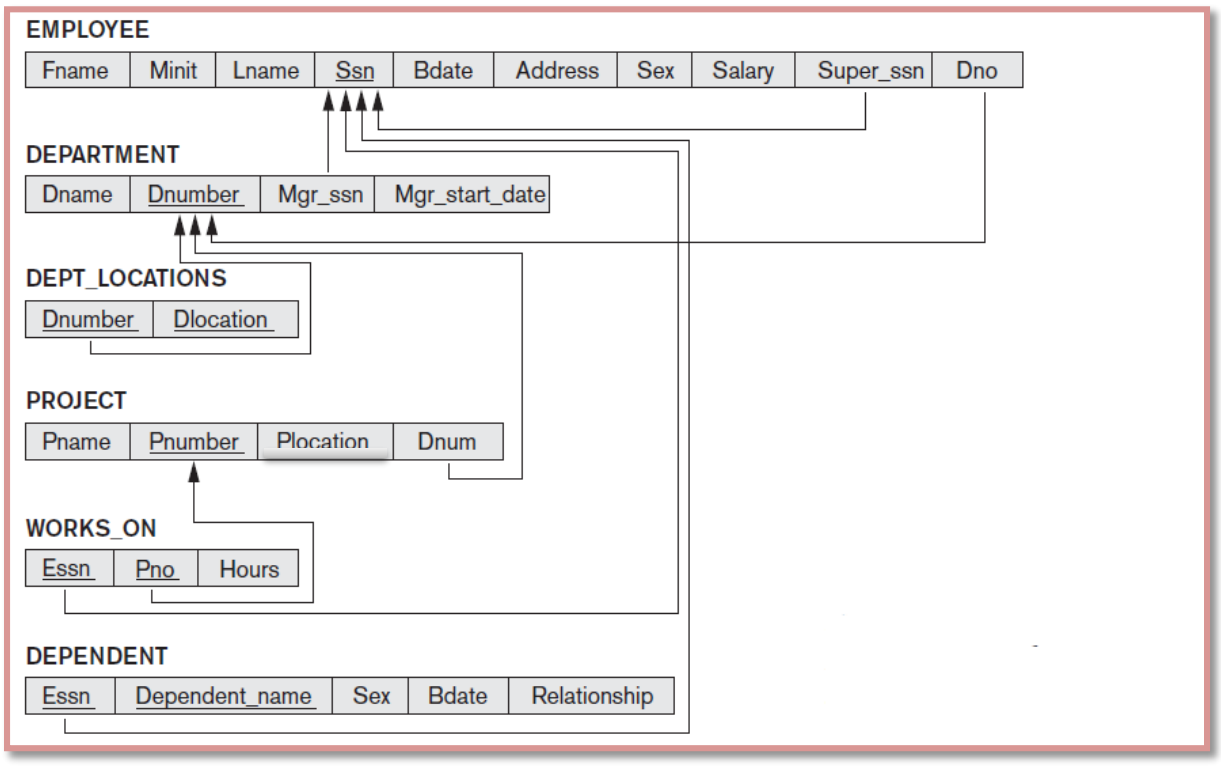
DROP TABLE TName [**CASCADE**]**;**

Notes:

- SQL Statements are executed sequentially.
- To run a statement, put the cursor on the statement and press Ctrl+Enter, or click on  icon.
- To run more than one statement sequentially, select them then click on  icon.
- You can run all statements in the worksheet, put the cursor on the first statement and press F5. Or click on  icon.

Example:

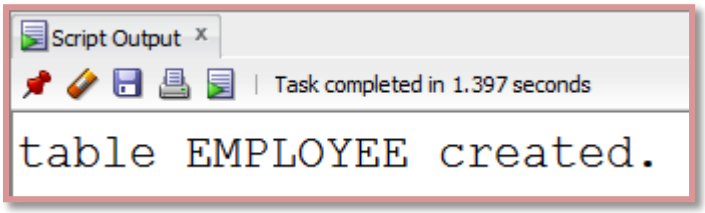
You have the following COMPANY Schema, create it on Oracle.



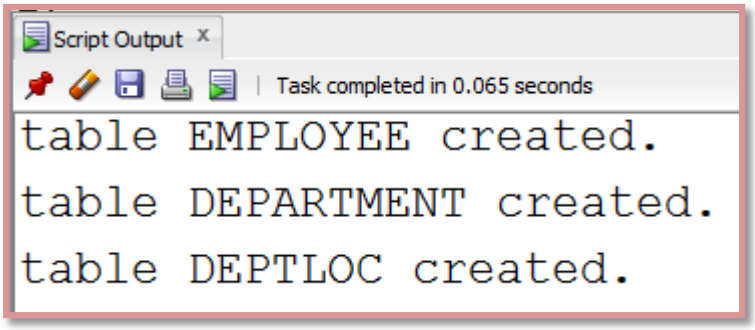
Solution:

```
create table EMPLOYEE(
Fname varchar2(10) Not Null,
Minit varchar2(10),
Lname varchar2(10),
Ssn varchar2(9) Not Null constraint pk PRIMARY KEY,
Bdate Date,
Address varchar2(20),
Sex varchar2(6),
Salary number default 2000 check (Salary >=2000),
Super_ssn varchar2(9) references EMPLOYEE(Ssn),
Dno integer
);
```

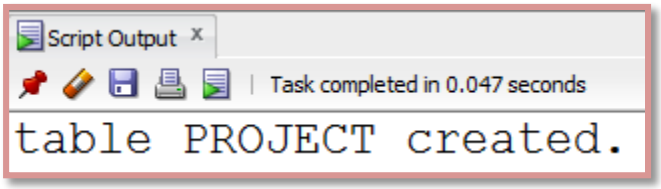
Note: Dno is a foreign key references the Department table but we haven't created DEPARTMENT table yet, so we can't add foreign constraint here, and we need to alter EMPLOYEE table to add the foreign constraint after DEPARTMENT creating.



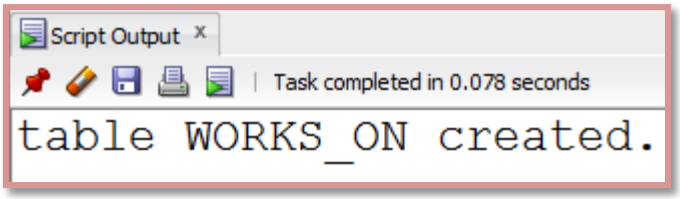
```
create table DEPARTMENT(
Dname varchar2(10) constraint unq unique,
Dnumber integer PRIMARY KEY,
Mgr_ssn varchar2(9),
Mgr_start_date Date,
constraint fk1 foreign key (Mgr_ssn) references EMPLOYEE (Ssn)
);

create table DEPTlOC(
Dnum integer constraint fk2 references DEPARTMENT(Dnumber),
Dlocation varchar2(10),
PRIMARY KEY (Dnum,Dlocation)
);
```



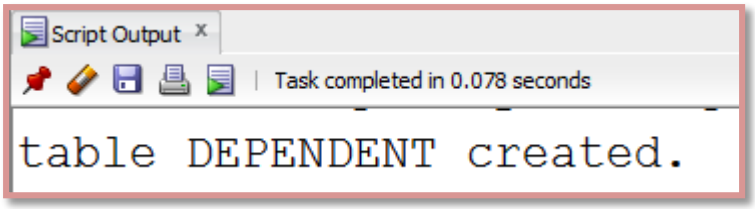

```
create table PROJECT(
Pname varchar2(10),
Pnumber int not null PRIMARY KEY,
Plocation varchar2(15),
Dnum integer constraint fk3 references DEPARTMENT(Dnumber),
Constraint Pro_uniq UNIQUE (Pname)
);
```



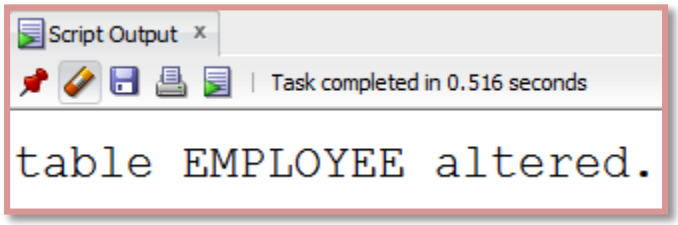
```
create table WORKS_ON(
Essn varchar2(9)references EMPLOYEE(Ssn),
Pno int constraint fk4 references PROJECT (Pnumber),
Hours numeric(5,2) Not Null,
Constraint PK PRIMARY KEY (Essn,Pno)
);
```



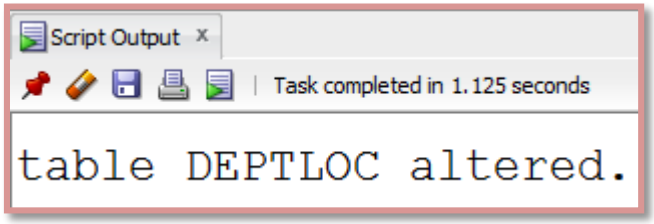

```
create table DEPENDENT(
Essn varchar2(9) references EMPLOYEE(Ssn),
Dependent_name varchar2(15),
Sex varchar2(6),
Bdate varchar2(4),
Relationship varchar2(10),
PRIMARY KEY (Essn,Dependent_name)
);
```



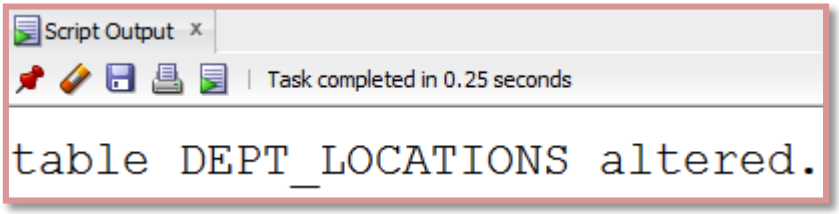
```
alter table EMPLOYEE
add constraint fk5 foreign key (Dno) references DEPARTMENT (Dnumber);
```



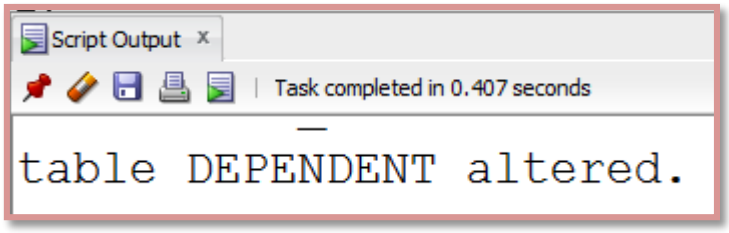

```
alter table DEPTlOC
Rename to DEPT_LOCATIONS;
```



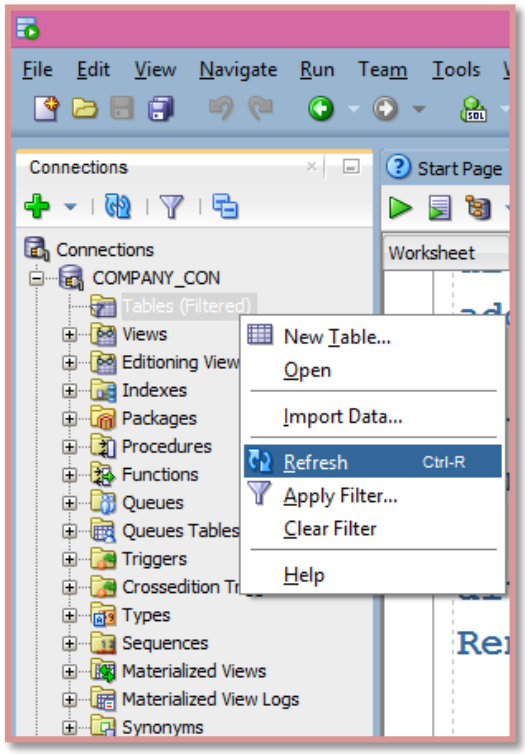

```
alter table DEPT_LOCATIONS
Rename Column Dnum to Dnumber;
```

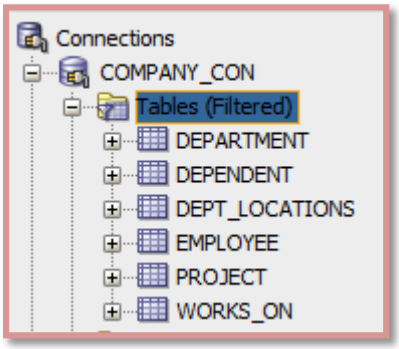



```
alter table DEPENDENT
modify Bdate Date;
```



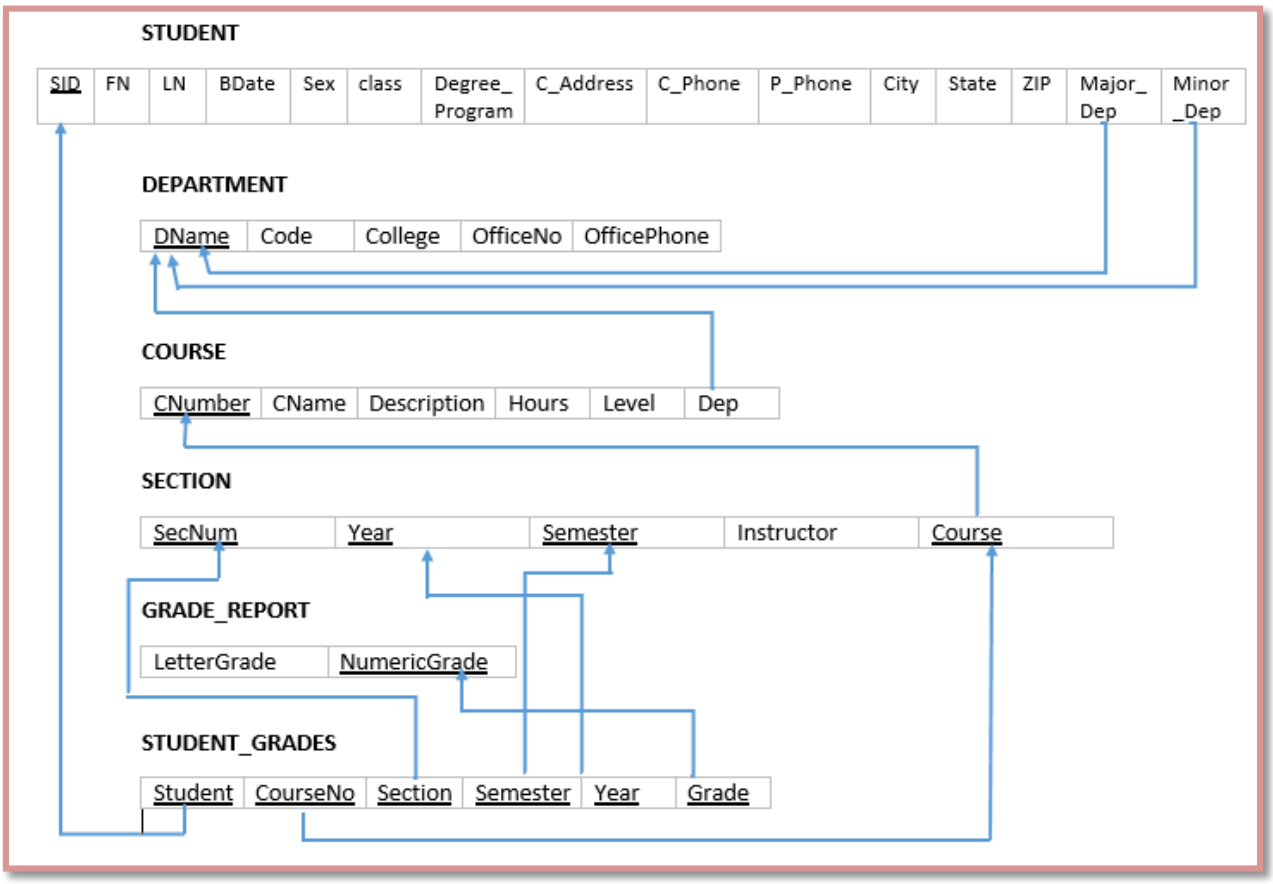
To view the tables in the schema, R-Click on Tables >> Refresh.



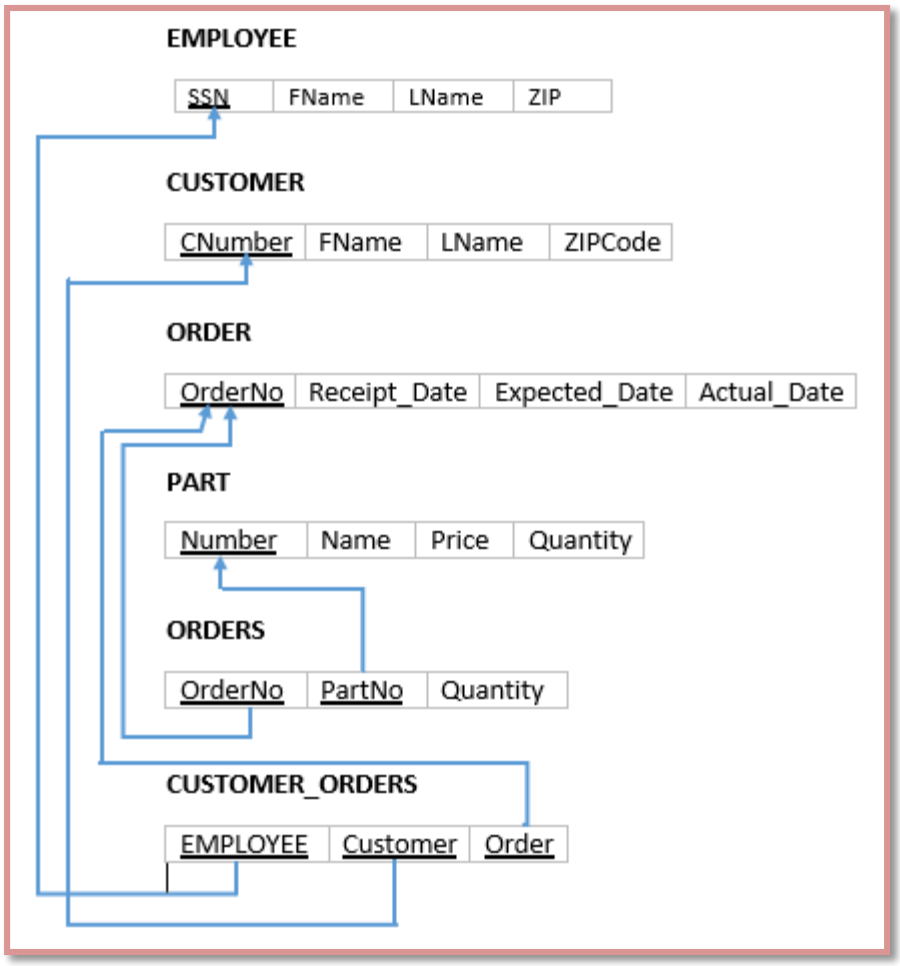


Exercises

1. Create the following University schema.



2. Create Mail Order Schema.



3. Create the following Sale Management System schema.

Sales_table (Sales_number, Sales_date, Customer_code)
Sales_detail_table (Sales_number, Product_code, Quantity)

Product_table (Product_code, Product_name,Unit_Price)
Sales_staff_table (Sales_staff_code, Sales_staff_name, Phone_number, E-mail)
Customer_table(Customer_Code, Customer_name, Customer_address, Phone_number, Sales_staff_code)

*Note: Underline is a Primary Key and Double underline is a Foreign key