RELATIONAL DATABASES

IMPLEMENTING CONSTRAINTS

OBJECTIVES

- DEFINE AND GIVE AN EXAMPLE OF A PRIMARY KEY, FOREIGN KEY AND UNIQUE KEY.
- DEMONSTRATE THE CREATION OF THESE CONSTRAINTS AT COLUMN OR TABLE LEVEL.

PRIMARY KEY CONSTRAINTS

- A PRIMARY KEY CONSTRAINT IS A RULE THAT THE VALUES IN ONE COLUMN OR COMBINATION OF COLUMNS MUST UNIQUELY IDENTIFY EACH ROW IN A TABLE.
- NO PRIMARY KEY VALUE CAN APPEAR IN MORE THAN ONE ROW IN A TABLE.
- TO SATISFY THE PRIMARY KEY CONSTRAINT BOTH OF THE FOLLOWING MUST BE TRUE:
    - NO COLUMN THAT IS PART OF THE PRIMARY KEY CAN CONTAIN A NULL.
    - A TABLE CAN ONLY HAVE ONE PRIMARY KEY.
- A PRIMARY KEY CONSTRAINTS CAN BE DEFINED AT THE COLUMN OR TABLE LEVEL.
- IF IT IS A COMPOSITE KEY IT MUST BE DEFINED AT THE TABLE LEVEL.

PRIMARY KEY CONSTRAINTS

```
CREATE TABLE employees
( employee_number INT AUTO_INCREMENT PRIMARY KEY,
  first_name VARCHAR(25),
 last_name VARCHAR(25),
 dob DATE,
 email VARCHAR(35),
 phone INT,
department INT
);
```

```
CREATE TABLE departments
( dept_number INT AUTO_INCREMENT,
dept_name VARCHAR(25),
dept_location VARCHAR(25),
 established DATE,
 phone INT,
PRIMARY KEY (dept_number)
);
```

PRIMARY KEY CONSTRAINTS

```
CREATE TABLE employees
( employee_number INT,
  first_name VARCHAR(25),
 last_name VARCHAR(25),
 dob DATE,
 email VARCHAR(35),
 phone INT,
 department INT,
 PRIMARY KEY (first_name, last_name)
);
```

PRIMARY KEY CONSTRAINTS

- WHEN YOU DEFINE THE PRIMARY KEY AT THE COLUMN LEVEL YOU USE THE KEYWORDS PRIMARY KEY AFTER THE COLUMN NAME AND DATA TYPE (AND AUTO_INCREMENT).
- WHEN YOU DEFINE THE PRIMARY KEY AT THE TABLE LEVEL YOU USE THE KEYWORDS PRIMARY KEY AFTER YOU HAVE DEFINED ALL COLUMNS.
- WHEN YOU DEFINE A COMPOSITE PRIMARY KEY YOU MUST DO SO AT THE TABLE LEVEL.

UNIQUE CONSTRAINTS

```
CREATE TABLE employees
( employee_number INT,
  first_name VARCHAR(25),
 last_name VARCHAR(25),
 dob DATE,
 email VARCHAR(35) UNIQUE,
 phone INT,
 department INT,
 PRIMARY KEY (first_name, last_name)
);
```

```
CREATE TABLE employees
( employee_number INT,
  first_name VARCHAR(25),
 last_name VARCHAR(25),
 dob DATE,
 email VARCHAR(35),
 phone INT,
 department INT,
 PRIMARY KEY (first_name, last_name),
 UNIQUE (phone,dob)
);
```

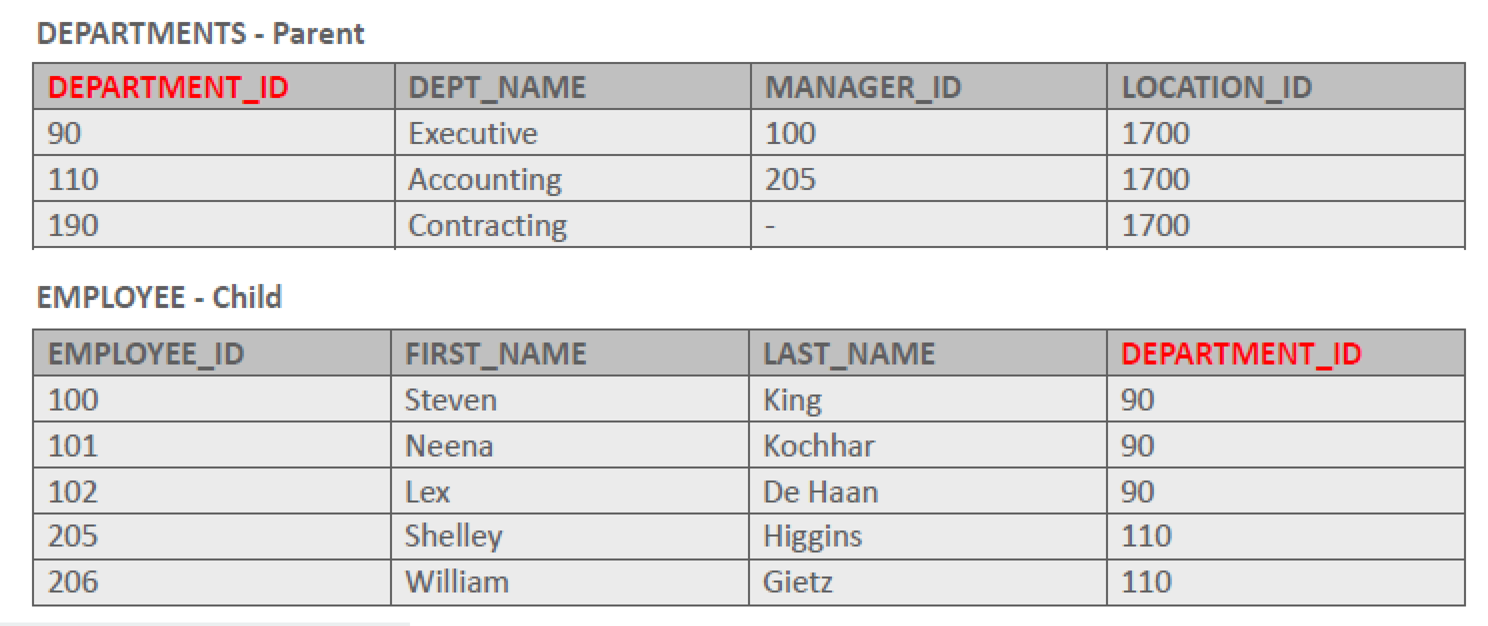
NOT NULL CONSTRAINT

```
CREATE TABLE employees
( employee_number INT,
  first_name VARCHAR(25) NOT NULL,
 last_name VARCHAR(25),
 dob DATE,
 email VARCHAR(35) UNIQUE,
 phone INT,
 department INT,
 PRIMARY KEY (first_name, last_name)
);
```

FOREIGN KEY CONSTRAINTS

- FOREIGN KEY CONSTRAINTS ARE ALSO CALLED ‘REFERENTIAL INTEGRITY’ CONSTRAINTS.
- THEY DESIGNATE A COLUMN OR COMBINATION OF COLUMNS AS A FOREIGN KEY.
- THEY LINK BACK TO THE PRIMARY KEY IN ANOTHER TABLE.
- THE TABLE CONTAINING THE FOREIGN KEY IS CALLED THE ‘CHILD’ TABLE AND THE TABLE IT REFERENCES IS CALLED THE ‘PARENT’ TABLE.

FOREIGN KEY CONSTRAINTS



DEPARTMENTS - Parent

DEPARTMENT_ID	DEPT_NAME	MANAGER_ID	LOCATION_ID
90	Executive	100	1700
110	Accounting	205	1700
190	Contracting	-	1700

EMPLOYEE - Child

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	DEPARTMENT_ID
100	Steven	King	90
101	Neena	Kochhar	90
102	Lex	De Haan	90
205	Shelley	Higgins	110
206	William	Gietz	110

- TO SATISFY A REFERENTIAL INTEGRITY CONSTRAINT A FOREIGN KEY VALUE MUST MATCH AN EXISTING VALUE IN THE PARENT TABLE OR BE NULL
- A PRIMARY KEY VALUE CAN EXIST WITHOUT A CORRESPONDING FOREIGN KEY BUT NOT VISA VERSA.

FOREIGN KEY CONSTRAINTS

- BEFORE YOU DEFINE A REFERENTIAL INTEGRITY CONSTRAINT IN THE CHILD TABLE, THE REFERENCED PRIMARY KEY CONSTRAINT IN THE PARENT TABLE MUST ALREADY EXIST.
- THIS MEANS WHEN YOU START TO CREATE YOUR TABLES YOU MUST CREATE EACH PARENT TABLE BEFORE YOU CREATE ANY ASSOCIATED CHILD TABLES.
- DON’T FORGET TO SET YOUR ON DELETE OPTION WHEN CREATING FOREIGN KEYS.

FOREIGN KEY CONSTRAINTS

```
CREATE TABLE employees
( employee_number INT AUTO_INCREMENT PRIMARY KEY,
  first_name VARCHAR(25),
 last_name VARCHAR(25),
 dob DATE,
 phone INT,
department INT,
FOREIGN KEY fk_dept(department) REFERENCES departments(department_number)
ON UPDATE CASCADE
ON DELETE RESTRICT
);
```