Data Definition and Constraints

The CREATE statement in SQL has a few important constraints that are highlighted below.

NOT NULL

The **NOT NULL** constraint indicates that the column cannot contain a null value.

Here is the syntax for adding a NOT NULL constraint to the CREATE statement:

```
CREATE TABLE IF NOT EXISTS customer_transactions (
    customer_id int NOT NULL,
    store_id int,
    spent numeric
);
```

You can add **NOT NULL** constraints to more than one column. Usually this occurs when you have a **COMPOSITE KEY**, which will be discussed further below.

Here is the syntax for it:

```
CREATE TABLE IF NOT EXISTS customer_transactions (
    customer_id int NOT NULL,
    store_id int NOT NULL,
    spent numeric
);
```

UNIQUE

The **UNIQUE** constraint is used to specify that the data across all the rows in one column are unique within the table. The **UNIQUE** constraint can also be used for multiple columns, so that the combination of the values across those columns will be unique within the table. In this latter case, the values within 1 column do not need to be unique.

Let's look at an example.

```
CREATE TABLE IF NOT EXISTS customer_transactions (
    customer_id int NOT NULL UNIQUE,
    store_id int NOT NULL UNIQUE,
    spent numeric
);
```

Another way to write a **UNIQUE** constraint is to add a table constraint using commas to separate the columns.

```
CREATE TABLE IF NOT EXISTS customer_transactions (
    customer_id int NOT NULL,
    store_id int NOT NULL,
    spent numeric,
    UNIQUE (customer_id, store_id, spent)
);
```

PRIMARY KEY

The **PRIMARY KEY** constraint is defined on a single column, and every table should contain a primary key. The values in this column uniquely identify the rows in the table. If a group of columns are defined as a primary key, they are called a **composite key**. That means the combination of values in these columns will uniquely identify the rows in the table. By default, the **PRIMARY KEY** constraint has the unique and not null constraint built into it.

Let's look at the following example:

```
CREATE TABLE IF NOT EXISTS store (
    store_id int PRIMARY KEY,
    store_location_city text,
    store_location_state text
);
```

Here is an example for a group of columns serving as **composite key**.

```
CREATE TABLE IF NOT EXISTS customer_transactions (
    customer_id int,
    store_id int,
    spent numeric,
    PRIMARY KEY (customer_id, store_id)
);
```

To read more about these constraints, check out the **PostgreSQL documentation**.