# Chapter 16: Database Administration in SQL

- SQL tables, like R data frames, consist of columns that are all of the same length but may be of different data types.
- Often we place additional constraints on columns of SQL tables to ensure *data integrity*.
- Some of the following keywords for database creation and administration are specific to MySQL.
- These keywords are not used in every SQL implementation.

## Creating New SQL Databases

- In MySQL, the command `SHOW DATABASES;` will show the available databases.
- `CREATE DATABASE` will create a new database, for example:

```
CREATE DATABASE airlines;
```

- `USE airlines;` will connect specifically to the `airlines` database.
- `SHOW TABLES;` will show the tables available in the current database.

## Creating a Table

- In MySQL, the `CREATE TABLE` command will create a new table.
- We can specify:
  1. the table name
  2. the name of the column (field)
  3. the data type for each column
  4. the width of each field
  5. and characteristics like whether `NULL` values are allowed for that column and what the default entry should be for that column.
- Once a table is created, we can modify it with `ALTER TABLE`.

## Example Code for Creating a Table

```
CREATE TABLE ‘airports‘ (
‘faa‘ varchar(3) NOT NULL DEFAULT ’’,
‘name‘ varchar(255) DEFAULT NULL,
‘lat‘ decimal(10,7) DEFAULT NULL,
‘lon‘ decimal(10,7) DEFAULT NULL,
‘alt‘ int(11) DEFAULT NULL,
‘tz‘ smallint(4) DEFAULT NULL,
‘dst‘ char(1) DEFAULT NULL,
‘city‘ varchar(255) DEFAULT NULL,
‘country‘ varchar(255) DEFAULT NULL,
PRIMARY KEY (‘faa‘)

ALTER TABLE airports CHANGE tz tz smallint(2) DEFAULT 0;
```

## Keys: Primary Key Columns

- *Keys* are special columns in SQL tables.
- A *primary key* column is a column that uniquely identifies each row.
- It is often something like an ID number/label, and it may be used as the key column when joining tables.
- A primary key column cannot have a `NULL` value, and every value in the primary key column must be *unique*.
- If you try to enter a row into a table which would result in a duplicate value in the `PRIMARY KEY` column, you will get an error and the insertion will be prevented (data integrity issue).

## Keys: Other Key Columns

- `UNIQUE KEY` columns are similar, but these can have `NULL` values.
- A `FOREIGN KEY` column is a column in one table that is connected to a `PRIMARY KEY` column in another table.
- The `SHOW KEYS` command will show the key columns in a table.

## Indices in Tables

- Use of an *index* or *indices* can help speed up operations like sorting or joining tables.
- These indices are built in advance and keep track of which records contain certain values.
- Indices are stored on disk, not in memory.
- They are available in SQL but not in R.

## Changing Data in SQL Tables

- The `UPDATE` command allows you to change values in a table over many rows at once.
- Since `UPDATE` can change many rows, be careful — there is no “undo” operation.
- Example code to change the airport name associated with an FAA code:

-
  ```
  UPDATE airports
  SET name = 'Washington National'
  WHERE faa = 'DCA';
  ```

## Adding Data Records to an SQL Table

- The `INSERT INTO` command is a quick way to insert a small number of rows into an existing SQL table.
- The `VALUES` clause specifies the variable values for the new row.
- For example, this inputs a new row with specified values for *two* variables (all other variables in this row will have the default data values):

-
  ```
  INSERT INTO airports (faa, name)
  VALUES ('SJU', 'Luis Munoz Marin International
  Airport');
  ```

## Adding Data Records to an SQL Table

- If we want to insert a large number of rows from a file, it's easier to use `LOAD DATA` than `INSERT`.
- See examples on the presidents' blood pressure table.