# CREATE DATABASE

- Syntax:
  CREATE DATABASE *database;*
  - *database* is the name of the database to create
- Create a new database based on your PennKey username.
  CREATE DATABASE <pennkey>_example;
  #Brandon Krakowsky would run CREATE DATABASE lbrandon_example;

# CREATE TABLE LIKE

- Syntax:
  CREATE TABLE *new_table* LIKE *old_table;*
  - *new_table* is the name of the new table
  - *old_table* is the name of the old table
- Create a “Patient” table in your new database to store patient records. This table should have the same structure as the “Patient” table in the “example” database.
  CREATE TABLE Patient LIKE example.Patient;

*Note this is not a supported function in SQLite

# UPDATE

- Syntax:
  UPDATE *table*
  SET *column* = *new_value*
  WHERE *condition*;
  - *column* is the name of the column to update
  - *condition* is a condition to be satisfied
- Change your new doctor's title to "DO".

```
UPDATE Doctor
SET d_title = 'DO'
WHERE d_id = 9;
```

# UPDATE

- Add the birthdate of 12/19/1981 for new patient “Elliot Graham”.

```
UPDATE Patient
SET bdate = '1981-12-19’
WHERE p_firstname = 'Elliot’
AND p_lastname = 'Graham';
```

# UPDATE

- Schedule an appointment for patient “Elliot Graham” on 02/16/2016 at 10AM with your new doctor.

```
UPDATE Appointment AS a
SET a.p_id = 12
WHERE a.d_id = 9
AND a.a_date = '2016-02-16 10:00:00’;
```

# UPDATE

- Schedule an appointment for patient “Elliot Graham” on 02/16/2016 at 10AM with your new doctor.

```
UPDATE Appointment AS a
SET a.p_id = 12
WHERE a.d_id = 9
AND a.a_date = '2016-02-16 10:00:00’;
```

- Schedule an appointment for patient “Brenda Cohann” on 02/16/2016 at 11AM with your new doctor.

```
UPDATE Appointment AS a
SET a.p_id = (SELECT p.p_id FROM Patient p WHERE p.p_firstname = 'Brenda'
AND p.p_lastname = 'Cohann’)
WHERE a.d_id = (SELECT d.d_id FROM Doctor d WHERE d.d_firstname = 'Brandon'
AND d.d_lastname = 'Krakowsky’)
AND a.a_date = '2016-02-16 11:00:00';
```

# DELETE FROM

- Syntax:
  DELETE FROM *table*
  WHERE *condition*;
  - You always want a **WHERE** condition when deleting records!
- Delete patient “Todd Wells”, who does not have any scheduled appointments.

```
DELETE FROM Patient
WHERE p_firstname = 'Todd’
AND p_lastname = 'Wells';
```

# DELETE FROM

- Delete the doctors who do not have any scheduled appointments.
```
DELETE FROM Doctor
WHERE d_id NOT IN (SELECT a.d_id FROM Appointment a);
```
- Cancel the appointment on 02/16/2016 at 10am with Elliot Graham.
```
DELETE a FROM Appointment a
JOIN Patient p
ON a.p_id = p.p_id
WHERE a.a_date = '2016-02-16 10:00:00'
AND (p.p_firstname = 'Elliot' AND p.p_lastname = 'Graham');
```
#When deleting using a JOIN, you must add a reference to the table(s) between the DELETE and the FROM

# ALTER TABLE – ADD Column

- Syntax:
  ALTER TABLE *table*
  ADD *column_name  data_type*  [*column_constraint*];
- Add a column to store the name of the medical school for each doctor.
  ALTER TABLE Doctor
  ADD COLUMN medical_school varchar(100) DEFAULT NULL;
- Add columns to store the residency and board certification date for each doctor.
  ALTER TABLE Doctor
  ADD COLUMN residency varchar(100) DEFAULT NULL,
  ADD COLUMN board_certification_date date;

# ALTER TABLE – ADD Column

- Add a column to store the office id for each appointment.
  ```
  ALTER TABLE Appointment
  ADD COLUMN office_id int; #This will be our foreign key pointing to the “o_id” column in the “Office” table
  ```
- Set the office id for *all* appointments to 1.
  ```
  UPDATE Appointment
  SET office_id = 1; #This will point to the default (only) office location record in the “Office” table
  ```
- Add a column to store the address for each patient.
  ```
  ALTER TABLE Patient
  ADD COLUMN p_address varchar(200) DEFAULT NULL;
  ```

# ALTER TABLE – DROP COLUMN

- Syntax:
  ALTER TABLE *table*
  DROP COLUMN *column_name*;
- Drop the column to store the residency for each doctor.
  ALTER TABLE Doctor
  DROP COLUMN residency;

# ALTER TABLE – MODIFY COLUMN

- Syntax:
  ALTER TABLE *table*
  MODIFY COLUMN *column_name  data_type  [column_constraint]*;
- Modify the patient address column so that it stores a larger string and is between the lastname column and bdate column.
  ALTER TABLE Patient
  MODIFY COLUMN p_address varchar(200) DEFAULT NULL AFTER p_lastname;

# ALTER TABLE – CHANGE COLUMN

- Syntax:
  ALTER TABLE *table*
  CHANGE COLUMN *old_name  new_name  data_type  [column_constraint]*;
- Rename the doctor “specialty” column to “d_specialty”.
  ALTER TABLE Doctor
  CHANGE COLUMN specialty d_specialty varchar(50);

# ALTER TABLE – RENAME TO

- Syntax:
  ALTER TABLE *table_name*
  RENAME TO *new_table_name*;
- Rename the Doctor table to “doc”.
  ALTER TABLE Doctor
  RENAME TO doc;
- Rename the Patient table to “pt”.
  ALTER TABLE Patient
  RENAME TO pt;
- Rename the Appointment table to “appt”.
  ALTER TABLE Appointment
  RENAME TO appt;

# ALTER TABLE – RENAME TO

- Rename them back!

```
ALTER TABLE doc
RENAME TO Doctor;

ALTER TABLE pt
RENAME TO Patient;

ALTER TABLE appt
RENAME TO Appointment;
```

# ALTER TABLE – ADD PRIMARY KEY

- Syntax:
  ALTER TABLE *table*
  ADD PRIMARY KEY (*column*);
- Make the “p_id” column the primary key in the Patient table.
  ALTER TABLE Patient
  ADD PRIMARY KEY (p_id);
- Also, let’s modify the “p_id” column so that it’s value is created automatically every time a new patient is inserted.
  ALTER TABLE Patient
  MODIFY COLUMN p_id int AUTO_INCREMENT; #The p_id column will auto increment every time a new record is inserted
- Insert a new patient record without the “p_id” and see what happens!

# ALTER TABLE – DROP PRIMARY KEY

- Syntax:
  ALTER TABLE *table*
  DROP PRIMARY KEY;
- Drop the primary key for the Patient table.
  ALTER TABLE Patient
  DROP PRIMARY KEY;
- Why does this fail?
  - We need first to modify the “p_id” column so that it’s value is *not* set to auto increment
    ALTER TABLE Patient
    MODIFY COLUMN p_id int;
  - Then we can drop the primary key!

# DROP TABLE

- Syntax:
  DROP TABLE *table;*

# Creating, Updating, & Deleting Data Summary

| Command Desc. | Basic Syntax Structure | Example |
|---|---|---|
| Create a table in a database | CREATE TABLE *table* (<br>*column_name data_type*<br>[*column_constraint*],<br>... ,<br>*column_name data_type*<br>[*column_constraint*],<br>[*table_constraint*]); | CREATE TABLE Doctor (<br>d_id int DEFAULT '0',<br>d_firstname varchar(75) DEFAULT NULL,<br>d_lastname varchar (75) DEFAULT NULL,<br>d_title varchar(10) DEFAULT NULL,<br>specialty varchar (50) DEFAULT NULL<br>); |
| Copy the structure of a table | CREATE TABLE *new_table* LIKE *old_table*; | CREATE TABLE Patient LIKE example.Patient; |
| Copy records into a table | INSERT INTO *new_table* (*new_table_col1*, ..., *new_table_col_n*)<br>(SELECT *old_table_col1*, ..., *old_table_col_n* FROM *old_table*); | INSERT INTO Doctor (d_id, d_firstname, d_lastname, d_title, specialty)<br>(SELECT d_id, d_firstname, d_lastname, d_title, specialty FROM example.Doctor); |
| Copy the structure of a table and all the records | CREATE TABLE *new_table*<br>SELECT * FROM *old_table*; | CREATE TABLE Appointment<br>SELECT * FROM example.Appointment; |
| Insert records into a table | INSERT INTO *table* (*column1*, ..., *column_n*)<br>VALUES (*value1*, ..., *value_n*); | INSERT INTO Doctor (d_id, d_firstname, d_lastname, d_title, specialty)<br>VALUES (9, 'Brandon', 'Krakowsky', 'MD', 'Dermatology'); |

# Creating, Updating, & Deleting Data Summary

| Command Desc. | Basic Syntax Structure | Example |
|---|---|---|
| Update records in a table | UPDATE *table*<br>SET *column* = *new_value*<br>WHERE *condition*; | UPDATE Doctor<br>SET d_title = 'DO'<br>WHERE d_id = 9; |
| Delete records from a table | DELETE FROM *table*<br>WHERE *condition*; | DELETE FROM Patient<br>WHERE p_firstname = 'Todd'<br>AND p_lastname = 'Wells'; |
| Alter a table in a database<br>Add a column | ALTER TABLE *table*<br>ADD *column_name data_type*<br>[*column_constraint*]; | ALTER TABLE Doctor<br>ADD COLUMN medical_school<br>varchar(100) DEFAULT NULL; |
| Alter a table in a database<br>Drop a column | ALTER TABLE *table*<br>DROP COLUMN *column_name*; | ALTER TABLE Doctor<br>DROP COLUMN residency; |
| Alter a table in a database<br>Modify a column's data type or move a column | ALTER TABLE *table*<br>MODIFY COLUMN *column_name data_type*<br>[*column_constraint*]; | ALTER TABLE Patient<br>MODIFY COLUMN p_address<br>varchar(200) DEFAULT NULL AFTER<br>p_lastname; |

# Creating, Updating, & Deleting Data Summary

| Command Desc. | Basic Syntax Structure | Example |
|---|---|---|
| Alter a table in a database<br>Change a column's name | ALTER TABLE *table*<br>CHANGE COLUMN *old_name new_name data_type* [*column_constraint*]; | ALTER TABLE Doctor<br>CHANGE COLUMN specialty d_specialty varchar(50); |
| Alter a table in a database<br>Rename a table | ALTER TABLE *table_name*<br>RENAME TO *new_table_name*; | ALTER TABLE Doctor<br>RENAME TO doc; |
| Alter a table in a database<br>Add a primary key | ALTER TABLE *table*<br>ADD PRIMARY KEY (*column*); | ALTER TABLE Patient<br>ADD PRIMARY KEY (p_id) |
| Alter a table in a database<br>Drop a primary key | ALTER TABLE *table*<br>DROP PRIMARY KEY; | ALTER TABLE Patient<br>DROP PRIMARY KEY; |
| Delete a table in a database | DELETE TABLE *table*; | DROP TABLE Patient; |