Objective: Use SQL queries to extract and analyze data from a database.

Tools: SQLite

Dataset: American Community Survey 1-Year Data for 2015

Procedure:

1. Schema

```
PS C:\sqlite3> sqlite3 acs-1-year-2015.sqlite
SQLite version 3.50.0 2025-05-29 14:26:00
Enter ".help" for usage hints.
sqlite> .schema
CREATE TABLE states (
    year INTEGER ,
    name TEXT ,
    geo_id TEXT ,
    total_population INTEGER ,
    white INTEGER ,
    black INTEGER ,
    hispanic INTEGER ,
    asian INTEGER ,
    american_indian INTEGER ,
    pacific_islander INTEGER ,
    other_race INTEGER ,
    median_age FLOAT ,
    total_households INTEGER ,
    owner_occupied_homes_median_value INTEGER ,
    per_capita_income INTEGER ,
    median_household_income INTEGER ,
    below_poverty_line INTEGER,
    foreign_born_population INTEGER,
    state TEXT
);
CREATE TABLE congressional_districts (
    year INTEGER ,
    name TEXT ,
    geo_id TEXT ,
    total_population INTEGER ,
    white INTEGER ,
    black INTEGER ,
    hispanic INTEGER ,
    asian INTEGER ,
    american_indian INTEGER ,
    pacific_islander INTEGER ,
    other_race INTEGER ,
    median_age FLOAT ,
    total_households INTEGER ,
    owner_occupied_homes_median_value INTEGER ,
    per_capita_income INTEGER ,
    median_household_income INTEGER ,
    below_poverty_line INTEGER,
    foreign_born_population INTEGER,
    state TEXT,
    congressional_district TEXT
);
CREATE TABLE places (
    year INTEGER ,
    name TEXT ,
    geo_id TEXT ,
    total_population INTEGER ,
    white INTEGER ,
    black INTEGER ,
    hispanic INTEGER ,
    asian INTEGER ,
    american_indian INTEGER ,
    pacific_islander INTEGER ,
    other_race INTEGER ,
    median_age FLOAT ,
    total_households INTEGER ,
    owner_occupied_homes_median_value INTEGER ,
    per_capita_income INTEGER ,
    median_household_income INTEGER ,
    below_poverty_line INTEGER,
    foreign_born_population INTEGER,
    state TEXT,
    place TEXT
);
CREATE INDEX "state_on_states" ON states(state);
CREATE INDEX "state_cd_on_cdistricts" ON congressional_districts(state, congressional_district);
CREATE INDEX "state_on_places" ON places(state);
CREATE INDEX "name_on_states" ON states(name);
CREATE INDEX "name_on_cdistricts" ON congressional_districts(name);
CREATE INDEX "name_on_places" ON places(name);
```

2. SELECT, WHERE, ORDER BY, GROUP BY

```
sqlite> SELECT name, total_population
   ...> FROM states
   ...> WHERE total_population > 10000000
   ...> ORDER BY total_population DESC;
California|39144818
Texas|27469114
Florida|20271272
New York|19795791
Illinois|12859995
Pennsylvania|12802503
Ohio|11613423
Georgia|10214860
North Carolina|10042802
sqlite> SELECT name, AVG(median_household_income) AS avg_income
   ...> FROM states
   ...> GROUP BY name
   ...> ORDER BY avg_income DESC;
Maryland|75847.0
District of Columbia|75628.0
Hawaii|73486.0
Alaska|73355.0
New Jersey|72222.0
Connecticut|71346.0
Massachusetts|70628.0
New Hampshire|70303.0
Virginia|66262.0
```

3. JOINS (INNER, LEFT, RIGHT)

```
sqlite> SELECT s.name AS state_name, cd.congressional_district, cd.total_population
   ...> FROM states s
   ...> INNER JOIN congressional_districts cd ON s.state = cd.state;
Alabama|01|706302
Alabama|02|686622
Alabama|03|703986
Alabama|04|684685
Alabama|05|708972
Alabama|06|700691
Alabama|07|667721
Alaska|00|738432
Arizona|01|759663
Arizona|02|713631
Arizona|03|761488
Arizona|04|739374
```

```
sqlite> SELECT s.name AS state_name, cd.congressional_district, cd.total_population
   ...> FROM states s
   ...> LEFT JOIN congressional_districts cd ON s.state = cd.state;
Alabama|01|706302
Alabama|02|686622
Alabama|03|703986
Alabama|04|684685
Alabama|05|708972
Alabama|06|700691
Alabama|07|667721
Alaska|00|738432
Arizona|01|759663
Arizona|02|713631
```

```
sqlite> SELECT cd.name AS district_name, s.name AS state_name
   ...> FROM congressional_districts cd
   ...> LEFT JOIN states s ON cd.state = s.state;
Congressional District 1 (114th Congress), Alabama|Alabama
Congressional District 2 (114th Congress), Alabama|Alabama
Congressional District 3 (114th Congress), Alabama|Alabama
Congressional District 4 (114th Congress), Alabama|Alabama
Congressional District 5 (114th Congress), Alabama|Alabama
Congressional District 6 (114th Congress), Alabama|Alabama
Congressional District 7 (114th Congress), Alabama|Alabama
Congressional District (at Large) (114th Congress), Alaska|Alaska
Congressional District 1 (114th Congress), Arizona|Arizona
```

4. Write subqueries

```
sqlite> SELECT name, median_household_income
   ...> FROM states
   ...> WHERE median_household_income > (
(x1...>     SELECT AVG(median_household_income) FROM states
(x1...> );
Alaska|73355
California|64500
Colorado|63909
Connecticut|71346
Delaware|61255
District of Columbia|75628
Hawaii|73486
Illinois|59588
Maryland|75847
Massachusetts|70628
Minnesota|63488
New Hampshire|70303
New Jersey|72222
New York|60850
```

```
sqlite> SELECT name, total_population
   ...> FROM places
   ...> WHERE total_population IS NOT NULL
   ...> ORDER BY total_population DESC
   ...> LIMIT 5;
New York city, New York|8550405
Los Angeles city, California|3971896
Chicago city, Illinois|2720556
Houston city, Texas|2298628
Philadelphia city, Pennsylvania|1567442
```

5. Aggregate functions (SUM, AVG)

```
sqlite> SELECT SUM(total_population) AS total_us_population FROM states;
324893003
sqlite> SELECT name, AVG(median_age) AS avg_age
   ...> FROM states
   ...> GROUP BY name;
Alabama|38.7
Alaska|33.3
Arizona|37.4
Arkansas|37.9
California|36.2
Colorado|36.4
Connecticut|40.6
Delaware|39.7
District of Columbia|33.8
Florida|41.8
Georgia|36.4
Hawaii|37.7
Idaho|35.8
Illinois|37.7
```

6. Create Views for Analysis

```
sqlite> CREATE VIEW state_population_income AS
   ...> SELECT name, total_population, median_household_income, per_capita_income
   ...> FROM states;
sqlite> CREATE VIEW district_poverty_analysis AS
   ...> SELECT state, congressional_district, below_poverty_line, total_population,
   ...>        CAST(below_poverty_line AS FLOAT) / total_population AS poverty_rate
   ...> FROM congressional_districts;
```

7. Optimize queries with indexes

```
sqlite> SELECT name, tbl_name, sql
   ...> FROM sqlite_master
   ...> WHERE type = 'index'
   ...> ORDER BY tbl_name;
state_cd_on_cdistricts|congressional_districts|CREATE INDEX "state_cd_on_cdistricts" ON congressional_districts(state, congressional_district)
name_on_cdistricts|congressional_districts|CREATE INDEX "name_on_cdistricts" ON congressional_districts(name)
state_on_places|places|CREATE INDEX "state_on_places" ON places(state)
name_on_places|places|CREATE INDEX "name_on_places" ON places(name)
name_on_states|states|CREATE INDEX "name_on_states" ON states(name)
state_on_states|states|CREATE INDEX state_on_states ON states(state)
```