**Data Science**

**IP6 Queries**

1. SELECT COUNT(DISTINCT(crime\_type) AS unique\_crimes

FROM incident\_reports; (16)

1. SELECT crime\_type, COUNT(crime\_type) AS num\_crimes

FROM incident\_reports

GROUP BY crime\_type

ORDER BY crime\_type;

1. SELECT count(DATEDIFF(date\_reported, date\_occured)) AS date\_diff

FROM incident\_reports

WHERE DATEDIFF(date\_reported, date\_occured) = 0;

1. SELECT date\_reported, date\_occured, crime\_type, YEAR(date\_reported)-YEAR(date\_occured) AS date\_difference

FROM incident\_reports

WHERE YEAR(date\_reported)-YEAR(date\_occured)>1

ORDER BY YEAR(date\_reported)-YEAR(date\_occured) DESC;

1. SELECT YEAR(date\_occured) AS year, COUNT(incident\_number) as num\_incidents

FROM incident\_reports

WHERE YEAR(date\_occured) > 2011

GROUP BY YEAR(date\_occured)

ORDER BY YEAR(date\_occured) DESC;

1. SELECT \*

FROM incident\_reports

WHERE crime\_type = “robbery”;

1. SELECT lmpd\_division, incident\_number, date\_occured

FROM incident\_reports

WHERE crime\_type = “robbery”

ORDER BY lmpd\_division, date\_occured;

1. SELECT date\_occured, crime\_type

FROM incident\_reports

WHERE zip\_code = “40202”

ORDER BY crime\_type, date\_occured;

1. SELECT zip\_code, COUNT(incident\_number) AS “num\_thefts”

FROM incident\_reports

WHERE crime\_type = “vehicle break-in/theft”

GROUP BY zip\_code

ORDER BY COUNT(incident\_number) DESC;

44 results

40202 has highest number of thefts

1. SELECT COUNT(DISTINCT(city))

FROM incident\_reports;

234 different cities

1. SELECT city, COUNT(incident\_number)

FROM incident\_reports

GROUP BY city

ORDER BY COUNT(incident\_number) DESC;

“LVIL” had the second highest number of incidents. This is weird because it’s not really a city name, it’s probably shorthand for Louisville.

1. SELECT uor\_desc, crime\_type

FROM incident\_reports

WHERE crime\_type != “other”

ORDER BY uor\_desc, crime\_type;

The Uniform Offense Reporting code classifies each crime type in further detail. So each UOR description will fall under the same crime type, but each crime type may have several UOR codes within it.

1. SELECT COUNT(DISTINCT(lmpd\_beat))

FROM incident\_reports;

1. SELECT COUNT(DISTINCT(offense\_code))

FROM nibrs\_codes;

Checked ^

1. SELECT DISTINCT(nibrs\_codes.offense\_code), incident\_reports.nibrs\_code

FROM nibrs\_codes

LEFT JOIN incident\_reports ON incident\_reports.nibrs\_code = nibrs\_codes.offense\_code

1. SELECT incident\_reports.date\_occured, incident\_reports.block\_address, incident\_reports.zip\_code, nibrs\_codes.offense\_description

FROM incident\_reports

JOIN nibrs\_codes ON incident\_reports.nibrs\_code = nibrs\_codes.offense\_code

WHERE incident\_reports.nibrs\_code IN (“240”, “250”, “270”, “280”)

GROUP BY incident\_reports.block\_address;

1. SELECT zip\_code, offense\_against

FROM incident\_reports, nibrs\_codes

WHERE incident\_reports.nibrs\_code = nibrs\_codes.offense\_code AND LENGTH(zip\_code) = 5 AND zip\_code != “999”

GROUP BY zip\_code;

1. SELECT offense\_against, COUNT(offense\_against)

FROM nibrs\_codes

GROUP BY offense\_against

ORDER BY offense\_against;

1. SELECT zip\_code, COUNT(crime\_type) AS “num\_fraud”

FROM incident\_reports

WHERE crime\_type = “fraud”

GROUP BY zip\_code

HAVING COUNT(crime\_type) > 1000;

1. SELECT lmpd\_division, crime\_type, offense\_description, date\_occured

FROM incident\_reports, nibrs\_codes

WHERE incident\_reports.nibrs\_code = nibrs\_codes.offense\_code AND crime\_type = “burglary”

GROUP BY lmpd\_division;