Please write an application preferrably in python, that calls the USGS API and store the result in a relational database of your choice.

https://earthquake.usgs.gov/fdsnws/event/1/

1) Please query all events that have occurred during year 2017

2) Read a JSON response from the API

3) Design the database objects required to store the result in a relational fashion.

4) Store the response in those objects

5) Provide query/analysis to give biggest earthquake of 2017

6) Provide query/analysis to give most probable hour of the day for the earthquakes bucketed by the range of magnitude (0-1,1-2,2-3,3-4,4-5,5-6,>6 For border values in the bucket, include them in the bucket where the value is a lower limit so for 1 include it in 1-2 bucket)

For question 1-4 provide following

Code for the integration



DB schema



For question 5 and 6 provide following

Queries for the analysis

Any interesting visualization (using any open source product or trial version of any product) for these questions. Just attach screenshot.

**STEP 1:** a temp table to get all the frequency per mag level per hour of day. Store it as a intermediate table.

CREATE TABLE earthquake\_frequency AS

SELECT

mag,

hour\_of\_day,

num\_of\_earthquake

FROM (

SELECT

STRFTIME("%H", DATETIME(time/1000.0, 'unixepoch', 'localtime')) AS hour\_of\_day,

CASE WHEN mag < 1 THEN '0-1'

WHEN mag >= 1 and mag < 2 THEN '1-2'

WHEN mag >= 2 and mag < 3 THEN '2-3'

WHEN mag >= 3 and mag < 4 THEN '3-4'

WHEN mag >= 4 and mag < 5 THEN '4-5'

WHEN mag >= 5 and mag < 6 THEN '5-6'

WHEN mag >= 6 THEN '>6'

END AS mag,

COUNT(1) AS num\_of\_earthquake

FROM earthquakes

WHERE mag >= 0

GROUP BY STRFTIME("%H", DATETIME(time/1000.0, 'unixepoch', 'localtime')),

CASE WHEN mag < 1 THEN '0-1'

WHEN mag >= 1 and mag < 2 THEN '1-2'

WHEN mag >= 2 and mag < 3 THEN '2-3'

WHEN mag >= 3 and mag < 4 THEN '3-4'

WHEN mag >= 4 and mag < 5 THEN '4-5'

WHEN mag >= 5 and mag < 6 THEN '5-6'

WHEN mag >= 6 THEN '>6'

END

) aa

ORDER BY mag, num\_of\_earthquake DESC;

**STEP2:** INNERJoin earthquake\_frequency with itself by subquery to take each MAX(num\_of\_earthquake) to find the hour\_of\_day for each mag level with highest frequency.

SELECT aa.mag,

aa.hour\_of\_day AS hour\_of\_highest\_possibility

FROM earthquake\_frequency aa

JOIN (SELECT mag, MAX(num\_of\_earthquake) max\_num FROM earthquake\_frequency GROUP BY mag) bb

ON aa.mag = bb.mag AND aa.num\_of\_earthquake = bb.max\_num

GROUP BY aa.mag, aa.hour\_of\_day

ORDER BY aa.mag ASC;

Visualization used seaborn heatmap, due to time limitation, I would not choose any JS based tool, like pyecharts (Echarts). It can perform very professional visualization as well.

