Reminder: Quiz 3, which is based on the following, is in-class Monday, March 2.

The cloud service provider you should use for quiz 3 is MicroSoft Azure.

As described in class, the subject matter for this quiz is to measure performance in a few different scenarios

of SQL DBs on a cloud provider (Azure).

Measuring and Improving Performance Relational DB, SQL (Cloud)

----------------------------------------

Task: You will get world earthquake data, import into SQL and with a web interface

allow users to find out (query) interesting information about those earthquakes.

You will measure performance, and then improve performance.

  Your assignment is to measure performance on SQL tables: creating, querying,

  modifying data (tuples).

  Starting with (fairly) large, well structured data at:

[https://earthquake.usgs.gov/earthquakes/feed/v1.0/csv.php](https://nam05.safelinks.protection.outlook.com/?url=https%3A%2F%2Fearthquake.usgs.gov%2Fearthquakes%2Ffeed%2Fv1.0%2Fcsv.php&data=02%7C01%7Cneelambhimjibha.makwana%40mavs.uta.edu%7C43e8a533bec043c6ed6908d7bdb4c928%7C5cdc5b43d7be4caa8173729e3b0a62d9%7C0%7C0%7C637186456482977904&sdata=XxydPD8gpsBSyJh7N4MhZlK0LzeDcrwCnd%2Bq6fdau0M%3D&reserved=0)

  (all earthquakes for the last 30 days)

  If earthquakes make you nervous, equally large (or larger) data exists on:

[https://www2.census.gov/programs-surveys/popest/datasets/](https://nam05.safelinks.protection.outlook.com/?url=https%3A%2F%2Fwww2.census.gov%2Fprograms-surveys%2Fpopest%2Fdatasets%2F&data=02%7C01%7Cneelambhimjibha.makwana%40mavs.uta.edu%7C43e8a533bec043c6ed6908d7bdb4c928%7C5cdc5b43d7be4caa8173729e3b0a62d9%7C0%7C0%7C637186456482987897&sdata=0SW7N19uG%2FeZeLoTClfsRYrAGIB2R4fL5jVQ4%2BEZmE8%3D&reserved=0)

  descriptions on: [https://www2.census.gov/](https://nam05.safelinks.protection.outlook.com/?url=https%3A%2F%2Fwww2.census.gov%2F&data=02%7C01%7Cneelambhimjibha.makwana%40mavs.uta.edu%7C43e8a533bec043c6ed6908d7bdb4c928%7C5cdc5b43d7be4caa8173729e3b0a62d9%7C0%7C0%7C637186456482987897&sdata=IhHunI4jfulis0TE2798unlTTNbYds8fwfeIfKAzTo4%3D&reserved=0)

  Create a SQL table, calculate time to create the table (and indexes).

  Allow a user to specify on a web interface:

1. A number of random queries (from 1 up to 1000 queries of random tuples in

     the dataset), the "random" values might be magnitude (for example random values between

    0.2 and 10), or location, or other attributes in that data.

1. A restricted set of queries, similar to previous (1.) but where selection is

     restricted (ie only occurring in CA, or within N<100 km of a specified

     lat,long location. These will (typically) return a much smaller result set.

     Or: a time range, or a magnitude range.

1. Measure time expended to perform these queries.
2. Show results. (on the web page)

  Users of this service will interact with your performance service through web

  page interfaces, all processing and web service hosting is (of course) cloud

  based.

You will use some type of RDB SQL to store and retrieve earthquake information.

And (of course) a friendly web UI.

You should handle conditions such as: missing data (fields, attributes), and similar.

1. Then, installing and using either memcache or Redis repeat steps 1 through 4.
2. And show results on a web interface (page).