Vehicle data lake

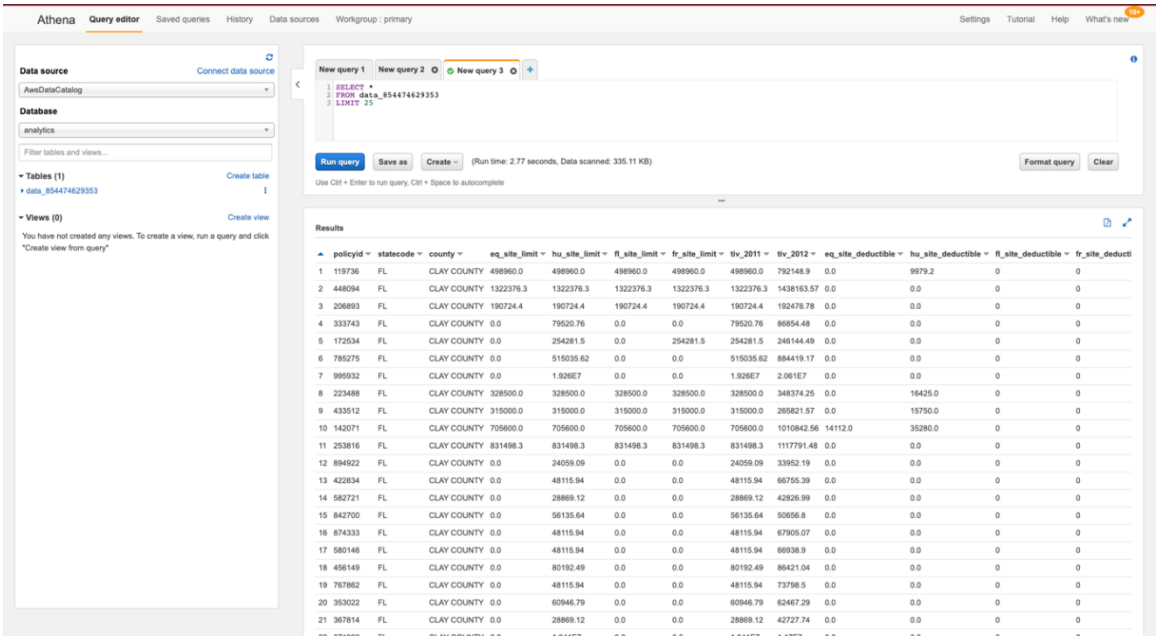
As a data engineer, you transform data into a useful format for analysis. Therefore, you have to provide a platform that can be easily used by a data scientists or other stakeholders, e.g. citizen data scientist in another department. It is important to ingest as many relevant data sources as possible and make the data accessible for analysis to create the maximum value for the MAN as a data-driven company.

The goal of this exercise is to check your architecture skills and cloud knowledge to fulfil the work of a data engineer. It should not take more than an hour to implement it. If it takes longer you maybe overengineer the solution. You are allowed to choose every cloud provider you want or you are familiar with. Please use the KISS principal and use as much services of the cloud provider as possible to fulfil the task.

The task is simple create a data lake solution, import the given data to it and make it possible to query the data. On the one hand side you have to deliver an architecture for a data lake solution and on the other hand side you have to implement the data lake solution. Our preferred cloud provider is Amazon Webservices (AWS) but you can use any other cloud provider too. AWS provides an implementation guide for data lake solution (https://docs.aws.amazon.com/solutions/latest/data-lake-solution/overview.html). You should use the idea of the implementation guide and simplify it a bit that you can stay in the one-hour time frame. It should be possible to deploy everything with the AWS Free Tier (https://aws.amazon.com/free/)

Picture 1 - The whole data lake setup should be done a infrastructure as code

From the functional perspective the user interface for data scientists should like this:



Picture 2 - Query Editor in AWS Athena that can be used to query the data

Task

Build a small **data lake** in any cloud you want:

1) Therefore, choose a **storage platform** that provides an optimal foundation for our data lake because of its virtually unlimited scalability. Setup the storage platform with DevOps principals like **infrastructure as a code**. The cloud automation framework is free of choice, too.
2) **Upload the data** file csv that contains the exercise zip file to the storage platform.
3) Index the data and create a meta data catalogue that make it possible to **query the data with SQL**
4) Create an aggregation data set inside of a new data storage area containing number of policies and highest eq_site_limit inside each county, with the corresponding policyID of this eq_site_limit. The application that is writing the aggregation data set should be deployed inside AWS. Index this data and make it queryable via SQL.

Requirements

- You create a small sketch of the data lake architecture.
- You create a reference implementation of our data lake architecture in a cloud provider of your choice.
- You have setup the architecture with a cloud automation tool of our choice.
- You are able to demonstrate and explain us the setup.

Nice to have

- The step 3 and 4 are automated – in case of a new file on step 3, there will be created a new aggregation on step 4.
- You ingest data from a data stream to the data lake. (e.g. moving data of vehicles from the LA Metro https://developer.metro.net/api/)

- Your project is deployed anywhere, where we can access but keep it in a private environment (give access to norbert-csaba.gergely@man.eu and bernd.zuther@man.eu).
- Usage of any CI/CD tool to test, build and deploy your project (Gitlab, Circle CI, Travis or any of your preference).