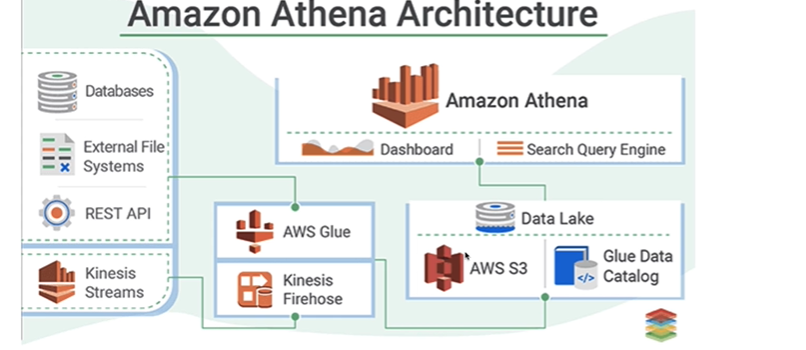
<https://www.youtube.com/watch?v=tykcCf-Zz1M>

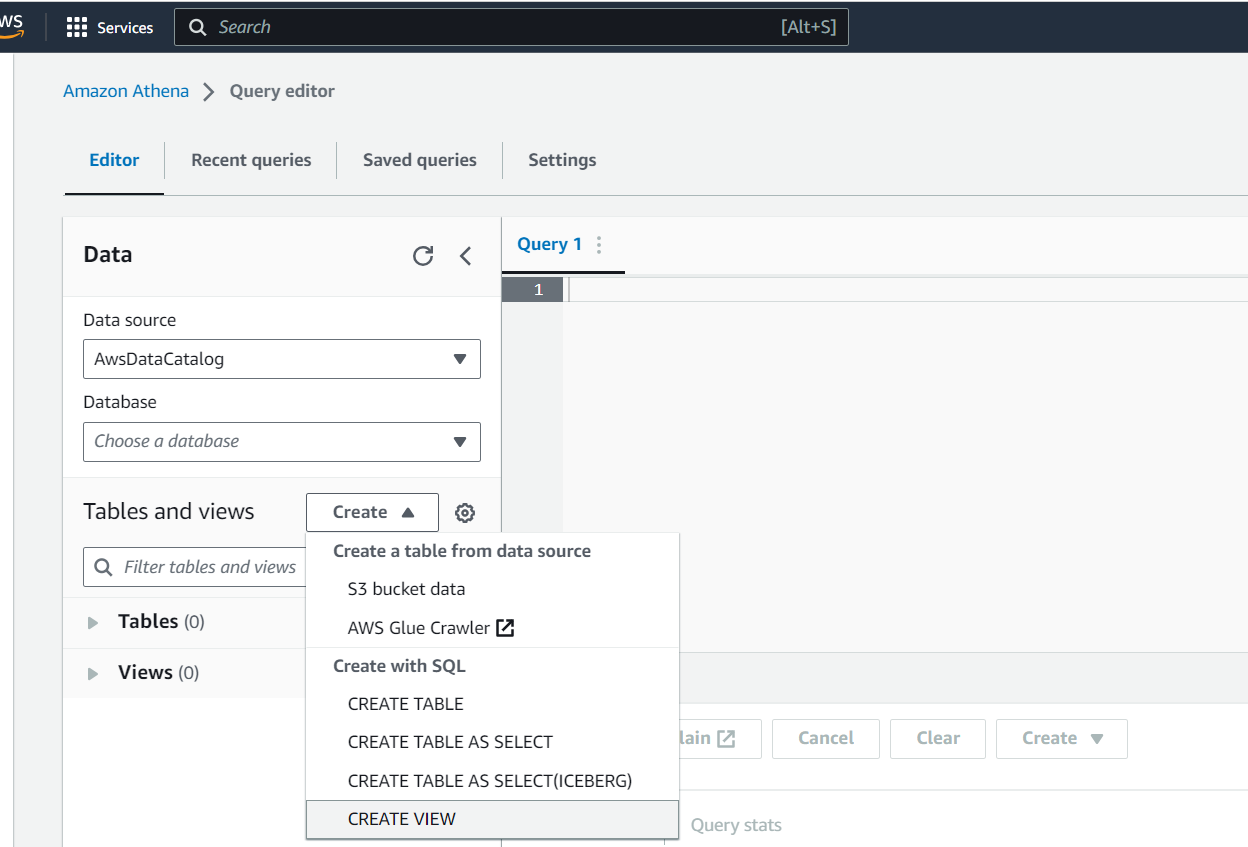


**Athena:**

****

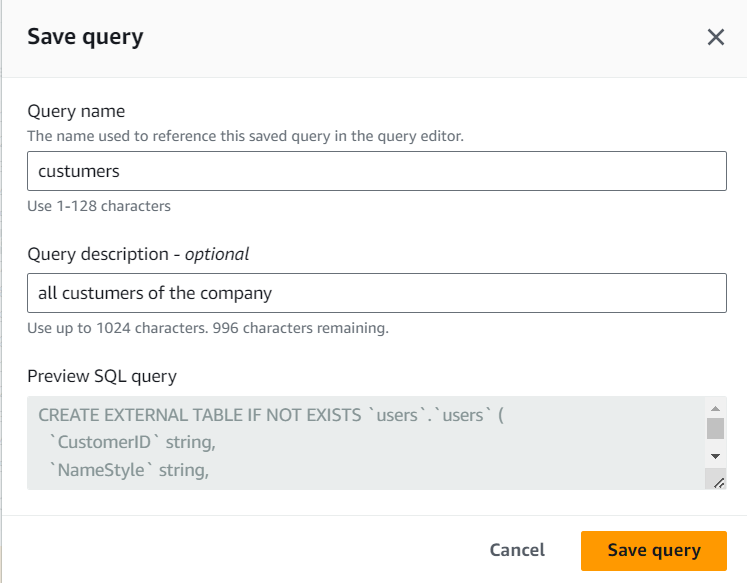
**Athena é mais barato para apenas fazer consultas no buckets**

Para criar base a partir do S3 é possivel fazer ou por point click ou com código.

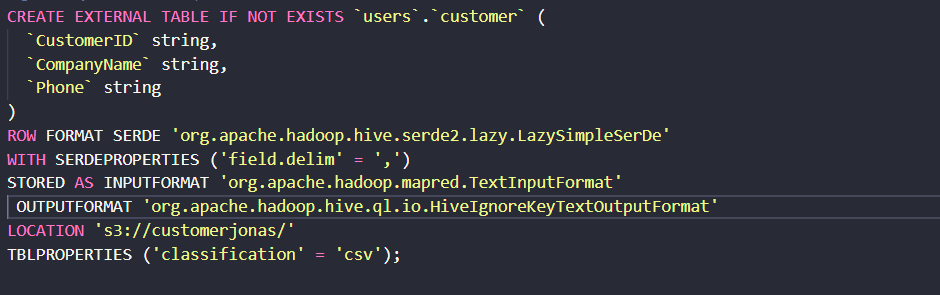


É preciso definir o nome da base, o database, qual é o formato do arquivo (csv, parquet,..) e incluir o nome das variaveis com o schema. O Athena irá criar um código.





O usual é criar os códigos e salvar.



CREATE EXTERNAL TABLE IF NOT EXISTS `users`.`customer` (

  `CustomerID` string,

  `CompanyName` string,

  `Phone` string

)

ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe'

WITH SERDEPROPERTIES ('field.delim' = ',')

STORED AS INPUTFORMAT 'org.apache.hadoop.mapred.TextInputFormat'

 OUTPUTFORMAT 'org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat'

LOCATION 's3://customerjonas/'

TBLPROPERTIES ('classification' = 'csv');



Como a AWS cobra por volume processado, transforme a tabela em parquet

create table "users"."customers\_parque"

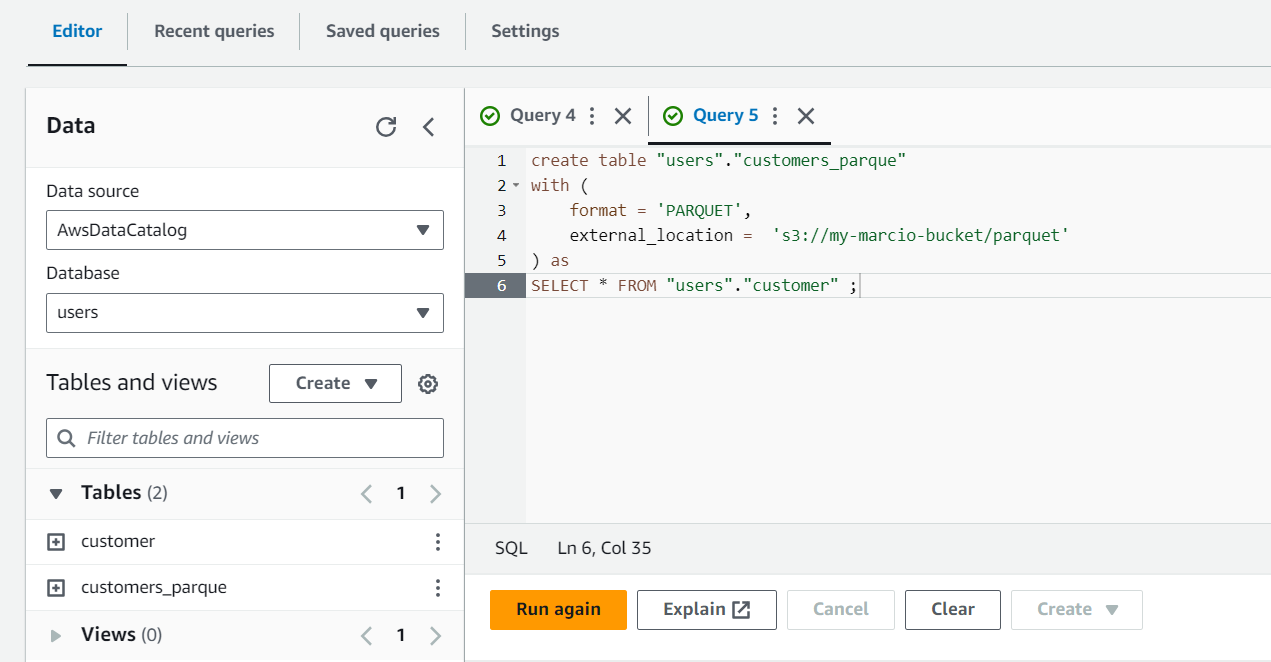
with (

    format = 'PARQUET',

    external\_location =  's3://my-marcio-bucket/parquet'

) as

SELECT \* FROM "users"."customer" ;



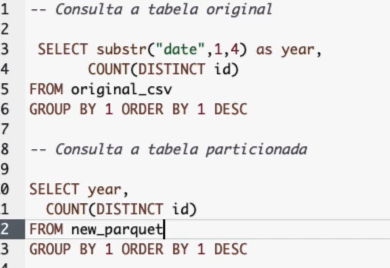
**tabelas particionadas**

Crie tabelas particionadas para separar o arquivo e diminuir o volume de consulta e consecutivamente o custo

Exemplo



Consulta: agora é possível consultar o folder e ler tudo ou a apenas uma partição





Inserindo mais dados



É possível gerenciar um limete de gastos por grupo de trabalho através do **workgroups.**   
o limite é feito por volume d e processamento e é possível criar alertas de quantidade já usado.

**AWS Glue**

