In [1]:

```
//Make a connection to sql server
//Connect to data storage and read the file into dataframe
//Convert df into temp table
//Convert temp table into hive table
//Write hive table to sql server

val user = "sqladmin1"
val pwd = "Tspl@12345"
val sqlhost = "mysqldbser.database.windows.net"
val sqlport = 1433
val sqldb = "mySampleDB"
```

```
Starting Spark application
```

| ID | YARN Application ID | Kind | State | Spark UI |
|---|---|---|---|---|
| 1 | application_1522302367312_0006 | spark | idle | Link (http://hn1-hdclus.pord2uzbbzqefnbet4ebrof3yc.cx.inte |

```
SparkSession available as 'spark'.
sqldb: String = mySampleDB
```

In [37]:

```
import java.util.Properties
val sqlurl = s"jdbc:sqlserver://${sqlhost}:${sqlport};database=${sqldb};encryp
t=true;trustServerCertificate=false;hostNameInCertificate=*.database.windows.n
et;loginTimeout=60;"
//print(sqlurl)
val sqlproperties = new Properties()

sqlproperties.put("user",s"${user}")
sqlproperties.put("password",s"${pwd}")
```

```
res74: Object = null
```

In [42]:

```
val custschema = spark.read.option("header",true).csv("wasb://hdcluster@4rrlm5
7xr2ozs.blob.core.windows.net/example/data/customerschema.csv").schema
print(custschema)
val custDF = spark.read.format("csv").schema(custschema).option("header",false
).option("inferschema",false).load("wasb://hdcluster@4rrlm57xr2ozs.blob.core.w
indows.net/example/data/customer.csv")
```

```
custDF: org.apache.spark.sql.DataFrame = [custid: string, custname: string
... 3 more fields]
```

In [43]:

```
custDF.createOrReplaceTempView("custtemp")
spark.sql("create table cust_hive8 as select * from custtemp")
spark.table("cust_hive8").write.mode("overwrite").jdbc(sqlurl,"customer",sqlpr
operties)
print("Data inserted")
```

```
Data inserted
```

In [ ]: