



Hive UDF

...

User Defined Functions in Hive

IMPORTANT

Copyright Infringement and Illegal Content Sharing Notice

All course content designs, video, audio, text, graphics, logos, images are Copyright© and are protected by India and international copyright laws. All rights reserved.

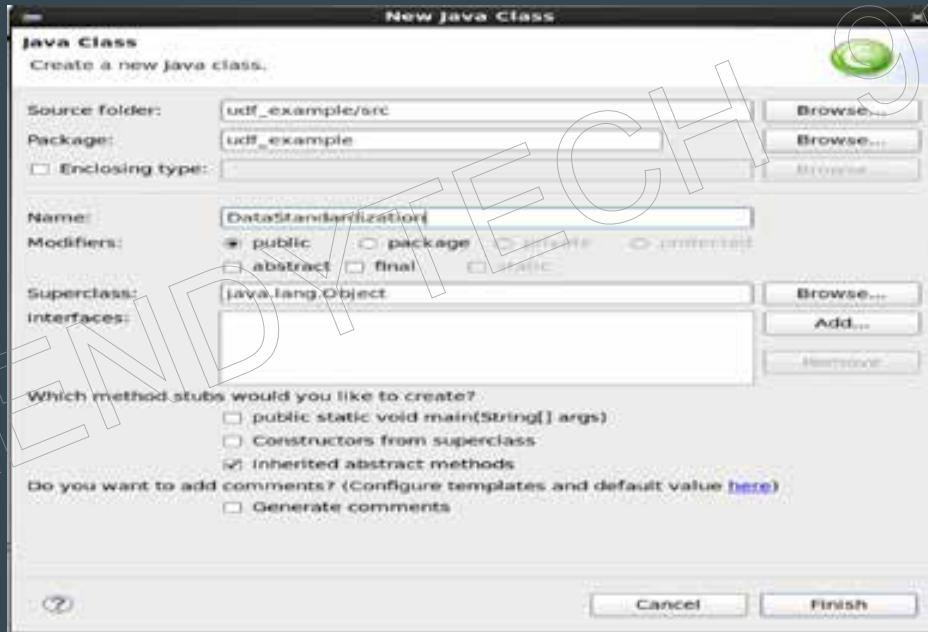
Permission to download the contents (wherever applicable) for the sole purpose of individual reading and preparing yourself to crack the interview only. Any other use of study materials – including reproduction, modification, distribution, republishing, transmission, display – without the prior written permission of Author is strictly prohibited.

Trendytech Insights legal team, along with thousands of our students, actively searches the Internet for copyright infringements. Violators subject to prosecution.



Create a UDF to convert String to Upper case

Open eclipse & Create a new java Project (udf_example)
Inside that create a New Class

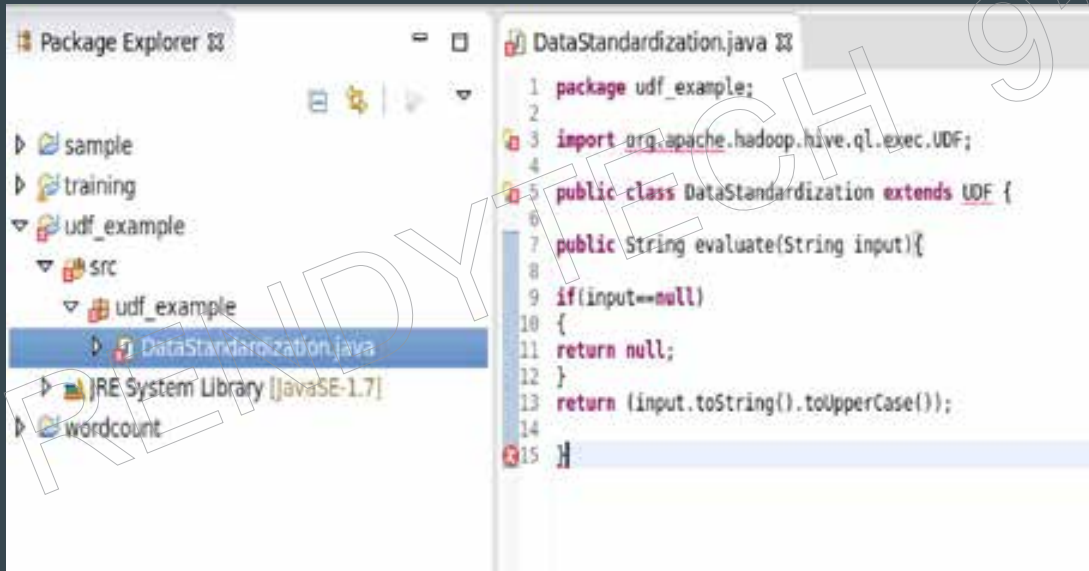


Package name:
udf_example

Class Name:
DataStandardization

What was the pain point?

Copy the given code in this DataStandardization.java file



The screenshot shows an IDE with the Package Explorer on the left and the DataStandardization.java file open in the editor. The Package Explorer shows a project structure with 'sample', 'training', 'udf_example', and 'src' folders. The 'src' folder is expanded, showing 'udf_example' and 'DataStandardization.java'. The editor shows the following code:

```
1 package udf_example;
2
3 import org.apache.hadoop.hive.ql.exec.UDF;
4
5 public class DataStandardization extends UDF {
6
7     public String evaluate(String input){
8
9         if(input==null)
10         {
11             return null;
12         }
13         return (input.toString().toUpperCase());
14     }
15 }
```

You will see few errors because some of the relevant jars are missing



Below is the Code

```
package udf_example;

import org.apache.hadoop.hive.ql.exec.UDF;

public class DataStandardization extends UDF {

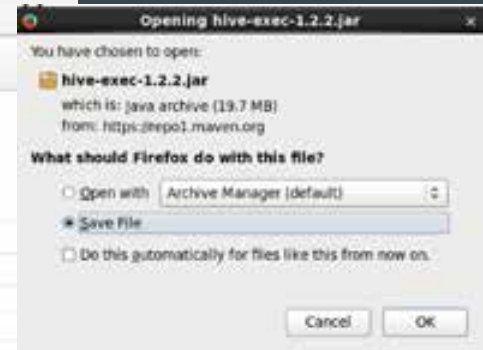
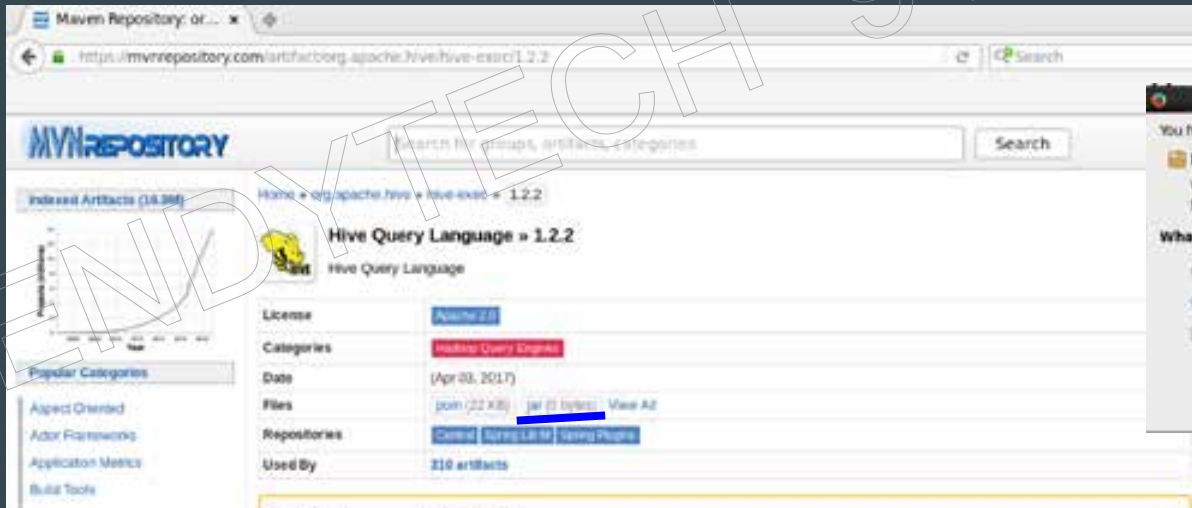
    public String evaluate(String input){

        if(input==null)
        {
            return null;
        }
        return (input.toString().toUpperCase());
    }
}
```

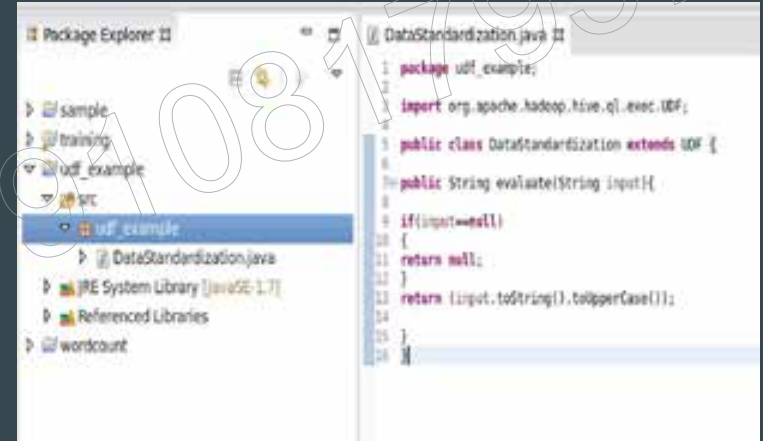
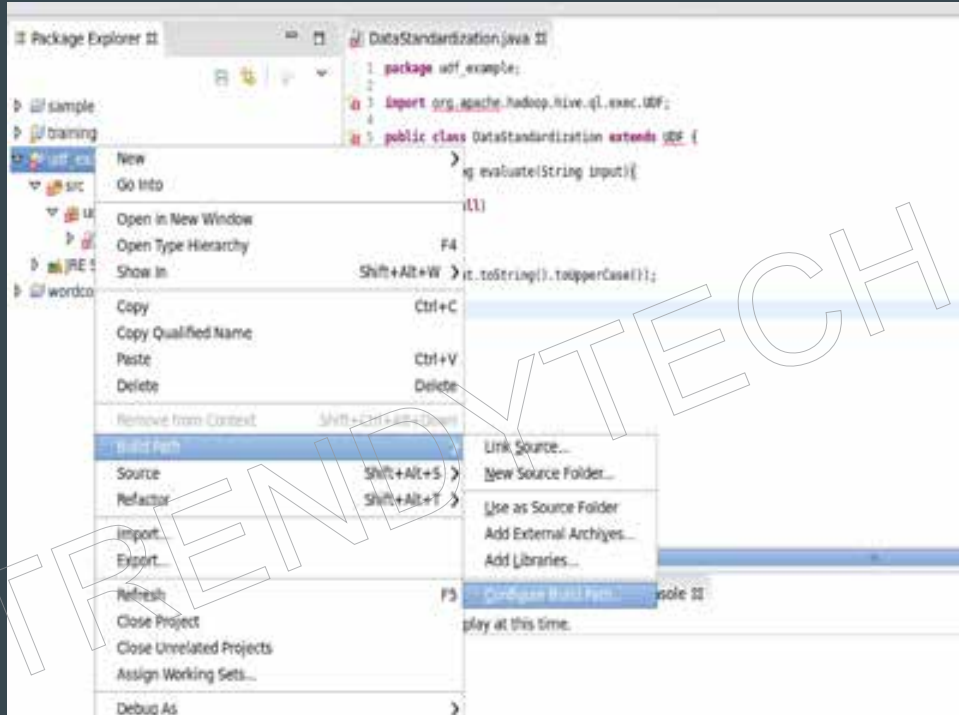
Download the necessary jar

Link to download the jar:

<https://mvnrepository.com/artifact/org.apache.hive/hive-exec/1.2.2>



Add the Jar hive-exec-1.2.2.jar



After adding the jar, you can see the errors are gone.



Let us create a Hive Table & Load Data

use trendytech;

create table if not exists sample_table(name string,count int) row format delimited fields terminated by ',' lines terminated by '\n' stored as textfile;

load data local inpath '/home/cloudera/Downloads/sample_data.txt' overwrite into table sample_table;

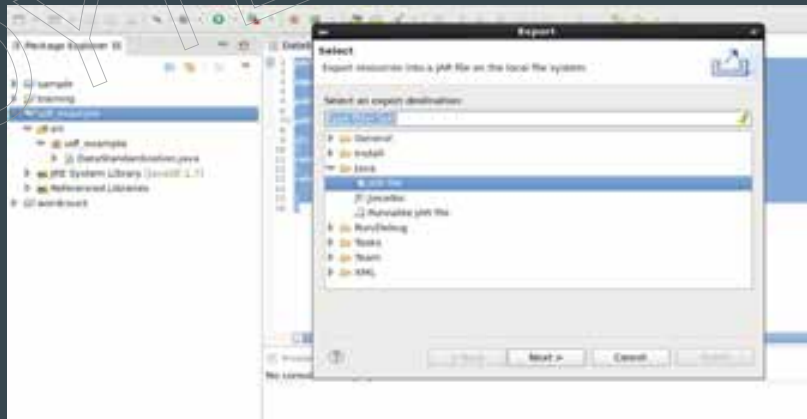
```
hive> create table if not exists sample_table(name string,count int) row format delimited fields terminated by ',' lines terminated by '\n' stored as textfile;
OK
Time taken: 0.328 seconds
hive> load data local inpath '/home/cloudera/Downloads/sample_data.txt' overwrite into table sample_table;
Loading data to table trendytech.sample_table
Table trendytech.sample_table stats: [numFiles=1, numRows=0, totalSize=153, rawDataSize=0]
OK
Time taken: 1.517 seconds
hive> select * from sample_table;
OK
John      1500
Albert    1500
Mark      1000
Frank     1150
Loopa     1100
Lui       1300
John      1300
John       900
Lesia     1500
Lesia     900
Pars      800
Leo       700
Leo       1500
Lock      650
Bhut      800
Lio       500
Time taken: 0.543 seconds, Fetched: 16 row(s)
```




export the jar file for the java project

Right click project -> export -> jar file

With this give the desktop path and jar will be created on Desktop.





Add the Jar & Create temporary function

ADD JAR /home/cloudera/Desktop/udf_example.jar;

create temporary function standardize as
'udf_example.DataStandardization';

```
File Edit View Search Terminal Help
hive> ADD JAR /home/cloudera/Desktop/udf_example.jar;
Added [/home/cloudera/Desktop/udf_example.jar] to class path
Added resources: [/home/cloudera/Desktop/udf_example.jar]
hive> █

-----
hive> create temporary function standardize as 'udf_example.DataStandardization';
OK
Time taken: 0.003 seconds
hive> █
```



Now we are all set, try calling this UDF on a column in a table

select standardize(name) from sample_table;

```
File Edit View Search Terminal Help
hive> select standardize(name) from sample_table;
OK
JOHN
ALBERT
MARK
FRANK
LOOPA
LUI
JOHN
JOHN
LESA
LESA
PARS
LEO
LEO
LOCK
BHUT
LIO
Time taken: 0.205 seconds, Fetched: 16 row(s)
hive> █
```



What's the issue?

These are temporary things. This jar & function is valid only for current session. That means if you come out of hive and go back in, then you won't be able to call this function.



Create a permanent function

For this we need to have our jar in HDFS

Let us transfer the jar to HDFS

`hadoop fs -put udf_example.jar /user/cloudera`

```
[cloudera@quickstart Desktop]$ hadoop fs -put udf_example.jar /user/cloudera  
[cloudera@quickstart Desktop]$ █
```



Create a permanent function

Now in Hive create a permanent function

```
CREATE FUNCTION standardize_permanent AS  
'udf_example.DataStandardization' using JAR  
'hdfs://localhost:8020/user/cloudera/udf_example.jar';
```

```
hive> CREATE FUNCTION standardize_permanent AS 'udf_example.DataStandardization' using JAR 'hdfs://lo  
calhost:8020/user/cloudera/udf_example.jar';  
converting to local hdfs://localhost:8020/user/cloudera/udf_example.jar  
Added [/tmp/dcee3d91-a797-48fb-8dc8-c598c7629416_resources/udf_example.jar] to class path  
Added resources: [hdfs://localhost:8020/user/cloudera/udf_example.jar]  
OK  
Time taken: 0.075 seconds  
hive> █
```



Try calling the function now.

```
select standardize_permanent(name) from  
sample_table;
```

```
hive> select standardize_permanent(name) from sample_table;  
OK  
JOHN  
ALBERT  
MARK  
FRANK  
LOOPA  
LUI  
JOHN  
JOHN  
LESA  
LESA  
PARS  
LEO  
LEO  
LOCK  
BHUT  
LIO  
Time taken: 0.085 seconds, Fetched: 16 row(s)  
hive> █
```



We learnt to Create a Hive UDF

Happy Learning!!!



5 Star Google Rated
Big Data Course

LEARN FROM THE EXPERT



9108179578

Call for more details



Follow US

Trainer Mr. Sumit Mittal

Phone 9108179578

Email trendytech.sumit@gmail.com

Website <https://trendytech.in/courses/big-data-online-training/>

LinkedIn <https://www.linkedin.com/in/bigdatabysumit/>

Twitter @BigdataBySumit

Instagram bigdatabysumit

Facebook <https://www.facebook.com/trendytech.in/>

Youtube TrendyTech