Pig UDF (User Defined Functions)

To specify custom processing, Pig provides support for user-defined functions (UDFs). Thus, Pig allows us to create our own functions. Currently, Pig UDFs can be implemented using the following programming languages: -

- Java
- Python
- Jython
- JavaScript
- Ruby
- Groovy

Among all the languages, Pig provides the most extensive support for Java functions. However, limited support is provided to languages like Python, Jython, JavaScript, Ruby, and Groovy.

Example of Pig UDF

In Pig,

- All UDFs must extend "org.apache.pig.EvalFunc"
- All functions must override the "exec" method.

Let's see an example of a simple EVAL Function to convert the provided string to uppercase.

UPPER.java

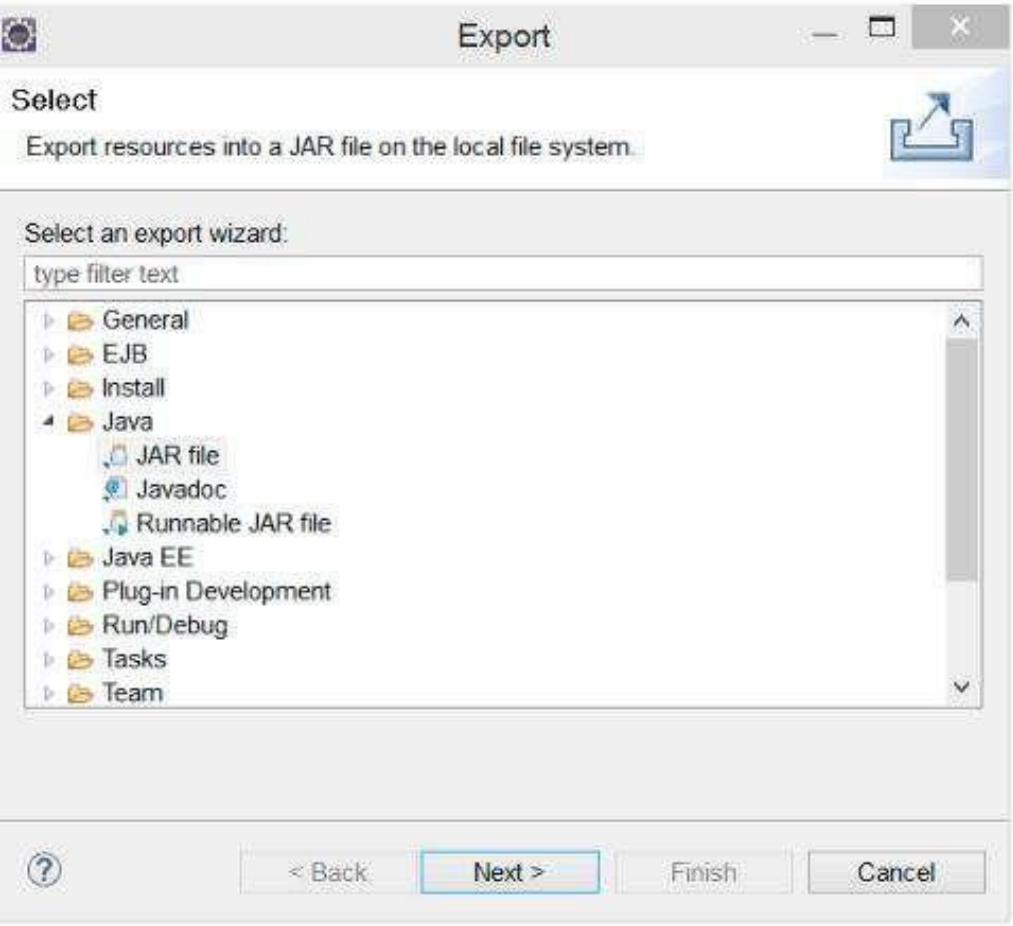
```
package com.hadoop;

import java.io.IOException;

import org.apache.pig.EvalFunc;
import org.apache.pig.data.Tuple;

public class TestUpper extends EvalFunc<String>  {
   public String exec(Tuple input) throws IOException {
      if (input == null || input.size() == 0)
      return null;
      try{
            String str = (String)input.get(0);
      return str.toUpperCase();
      }catch(Exception e){
      throw new IOException("Caught exception processing input row ", e);
          }
        }
}
```

- Create the jar file and export it into the specific directory. For that ,right click on project - **Export** - **Java** - **JAR file** - **Next**.



- Now, provide a specific name to the jar file and save it in a local system directory.

- Create a text file in your local machine and insert the list of tuples.

```
$ nano pigsample
```

- Upload the text files on HDFS in the specific directory.

```
$ hdfs dfs -put pigexample /pigexample
```

- Create a pig file in your local machine and write the script.

```
$ nano pscript.pig
```

- Now, run the script in the terminal to get the output.

```
$pig pscript.pig
```