

 免费电子书

学习

apache-pig

Free unaffiliated eBook created from
Stack Overflow contributors.

#apache-pig

.....	1
1: apache-pig	2
.....	2
Examples.....	2
.....	2
Linux.....	2
Pig.....	3
.....	4
2:	5
Examples.....	5
.....	5
3:	7
Examples.....	7
.....	7
ElasticSearch.....	7
.....	9

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [apache-pig](#)

It is an unofficial and free apache-pig ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official apache-pig.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

1: apache-pig

apache-pig。

apache-pig。 apache-pig。

Examples

Linux

r0.16.0

Apache-PigUnixWindows。

- Hadoop 0.23.X1.X2.X
- Java 1.6JAVA_HOMEJava
- Python 2.7Python UDF
- Ant 1.8

Pig

<http://pig.apache.org/releases.html#Downloadpig>

```
mkdir Pig
cd Downloads/
tar zxvf pig-(latest-version).tar.gz
tar zxvf pig-(latest-version).tar.gz
mv pig-(latest-version).tar.gz/* /home/Pig/
```

Apache Pig。

.bashrc

```
vim ~/.bashrc
```

.bashrc -

```
export PIG_HOME = /home/Pig
export PATH = PATH:/home/Pig/bin
```

bashrc

```
. ~/.bashrc
```

```
pig -version
```

Pig

```
pig -h
```

pig

```
pig -x local
```

Hadoop

Hadoop1.x2.xHADOOP_HOME

.bashrcpigHadoop

```
export PIG_CLASSPATH = $HADOOP_HOME/conf
```

bin / pigjar java -cp pig.jar Pig

PIG3

- pig -x local ...

- **Mapreduce**

```
pig -x mapreduce ...  
    (or)  
pig ...
```

- **Tez**

```
pig -x tez ...
```

PigGrunt shell shellPig Latin

```
$ pig -x <mode> <enter>  
grunt>
```

Mode

Pig pig.pig

```
$ pig -x <mode> <script.pig>  
grunt>
```

Mode

Pig

```
Mary had a little lamb
its fleece was white as snow
and everywhere that Mary went
the lamb was sure to go.
```

```
-- Load input from the file named Mary, and call the single
-- field in the record 'line'.
input = load 'mary' as (line);

-- TOKENIZE splits the line into a field for each word.
-- flatten will take the collection of records returned by
-- TOKENIZE and produce a separate record for each one, calling the single
-- field in the record word.
words = foreach input generate flatten(TOKENIZE(line)) as word;

-- Now group them together by each word.
grp = group words by word;

-- Count them.
cntd = foreach grp generate group, COUNT(words);

-- Print out the results.
dump cntd;
```

```
Mary,2
had,1
a,1
little,1
lamb,2
its,1
fleece,1
was,2
white,1
as,1
snow,1
and,1
everywhere,1
that,1
went,1
the,1
sure,1
to,1
go,1
```

PigHadoop。 Pig Latin。 Pig Latin。 PigApache。 Apache - 。

[apache-pig https://riptutorial.com/zh-CN/apache-pig/topic/3244/apache-pig](https://riptutorial.com/zh-CN/apache-pig/topic/3244/apache-pig)

2:

Examples

4

1. A / B/
2. /
3. /
4. /

```
test    mobile    order_module    click
prod    web      order_module    view
prod    mobile    order_module    click
```

- 1.
- 2.
3. **prodorder_module**

◦ ◦ ◦

PIGCUBE◦

```
example = LOAD './cube.example' AS (product:chararray, client:chararray, module:chararray,
action:chararray);

cubed_data = CUBE example BY CUBE(product, client, module, action);

final_data = FOREACH cubed_data GENERATE $0, COUNT_STAR($1);

dump final_data;
```

◦ - ◦ ◦

```
((prod,web,order_module,view),1)
((prod,web,order_module,),1)
((prod,web,,view),1)
((prod,web,,),1)
((prod,mobile,order_module,click),1)
((prod,mobile,order_module,),1)
((prod,mobile,,click),1)
((prod,mobile,,),1)
((prod,,order_module,view),1)
((prod,,order_module,click),1)
((prod,,order_module,),2)
((prod,,,view),1)
((prod,,,click),1)
((prod,,,),2)
((test,mobile,order_module,click),1)
((test,mobile,order_module,),1)
((test,mobile,,click),1)
((test,mobile,,),1)
```

```
((test,,order_module,click),1)
((test,,order_module,),1)
((test,,,click),1)
((test,,,),1)
((,web,order_module,view),1)
((,web,order_module,),1)
((,web,,view),1)
((,web,,,),1)
((,mobile,order_module,click),2)
((,mobile,order_module,),2)
((,mobile,,click),2)
((,mobile,,,),2)
((,,order_module,view),1)
((,,order_module,click),2)
((,,order_module,),3)
((,,,view),1)
((,,,click),2)
((,,,),3)
```

<https://riptutorial.com/zh-CN/apache-pig/topic/6120/>

3:

Examples

HDFS ◦ CSV ◦

```
ABT,20160106,42.310001,42.98,42.209999,42.560001,5906000
BAC,20160201,14.05,14.09,13.8,13.96,105739400
CAS,20160129,1.9,1.97,1.83,1.84,34500
DCA,20160129,3.46,3.54,3.46,3.51,84600
ECL,20160114,103.480003,105.400002,102.480003,104.82,1485000
FAF,20160201,34.040001,34.82,33.939999,34.639999,1222600
TYL,20160201,156.070007,159.550003,155.690002,158.259995,177100
UTL,20160201,38.610001,39.889999,38.57,39.27,119500
VTR,20160128,54.09,54.73,53.549999,53.790001,2441300
WWE,20160201,17.629999,18,17.27,17.799999,734100
XRX,20160104,10.41,10.43,10.13,10.3,9122600
YUM,20160104,71.32,72.25,70.639999,72.209999,3466300
ZTR,20160104,12.1,12.14,11.98,12.11,60200
```

1 LOAD

```
stocks = load '/user/pig/stock.txt' using PigStorage(',') as
          (sym:chararray, date:int, open:float, high:float, low:float,
           close:float, vol:int);
```

ElasticSearch

◦

ElasticSearch `elasticsearch` `elasticsearch-hadoop` ◦

`jar` `elasticsearch-hadoop-<version>.jar` `pig` ◦ - - `commons-httpclient-<version>.jar` ◦

`shell` `pig` `grunt` `pig` ◦

```
REGISTER /path/to/jars/commons-httpclient-<version>.jar;
REGISTER /path/to/jars/elasticsearch-hadoop-<version>.jar;
```

◦

◦

```
DATA = LOAD 'my_index/log' USING org.elasticsearch.hadoop.pig.EsStorage (
  'es.nodes=https://server1:port1,https://server2:port2,https://server3:port3',
  'es.query=?q=*',
  'es.net.ssl=true',
  'es.net.http.auth.user=user',
  'es.net.http.auth.pass=pass',
  'es.net.ssl.keystore.type=JKS',
```

```
'es.net.ssl.truststore.location=file:///path/to/truststore.jks',  
'es.net.ssl.truststore.pass=pass');
```

◦

- es.nodes **ElasticSearch** ◦ ◦
- es.query **ElasticSearch** ◦ **DSL** **DSL** es.read.source.filter ◦ **DSL** 'es.query = { "query":{
"match_all":{ } } }'
- es.net.ssl=true es.net.http.auth.user es.net.http.auth.passes es.net.http.auth.pass ◦
- es.net.ssl.keystore.type ◦ es.net.ssl.truststore.location file:///es.net.ssl.truststore.pass
◦
- es.read.source.filter=field1, field2, field3 **ElasticSearch** ◦
- es.output.json=true **JSON** ◦ false **CSV** ◦

<https://riptutorial.com/zh-CN/apache-pig/topic/7257/>

S. No		Contributors
1	apache-pig	Bhavesh , Brian Armstrong , Community , DhiwaTdG , Mzzzzzzz , pratiklodha
2		Wenzhong
3		Andrea Romagnoli , DhiwaTdG