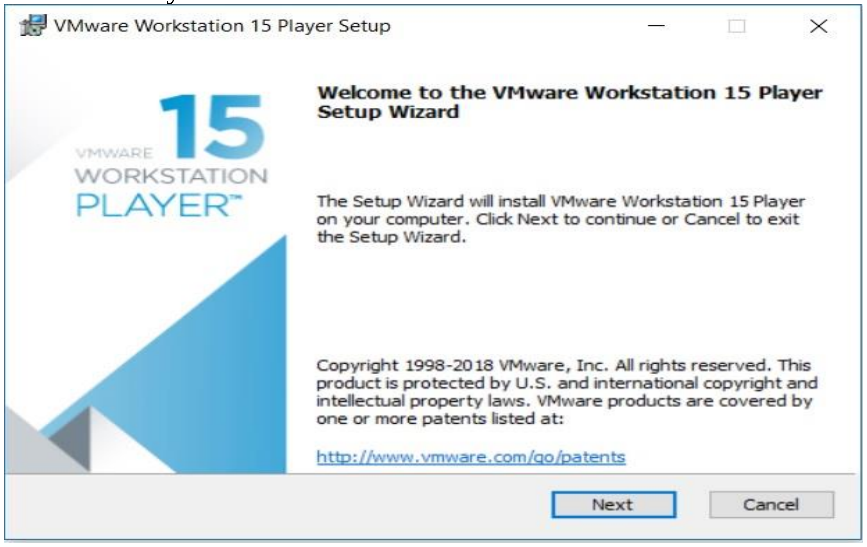
PRACTICAL –1

AIM-To install VMware Workstation on your system

PROCEDURE-

VMware Player 15 Installation –



VMware Workstation 15 Player Setup

Welcome to the VMware Workstation 15 Player Setup Wizard

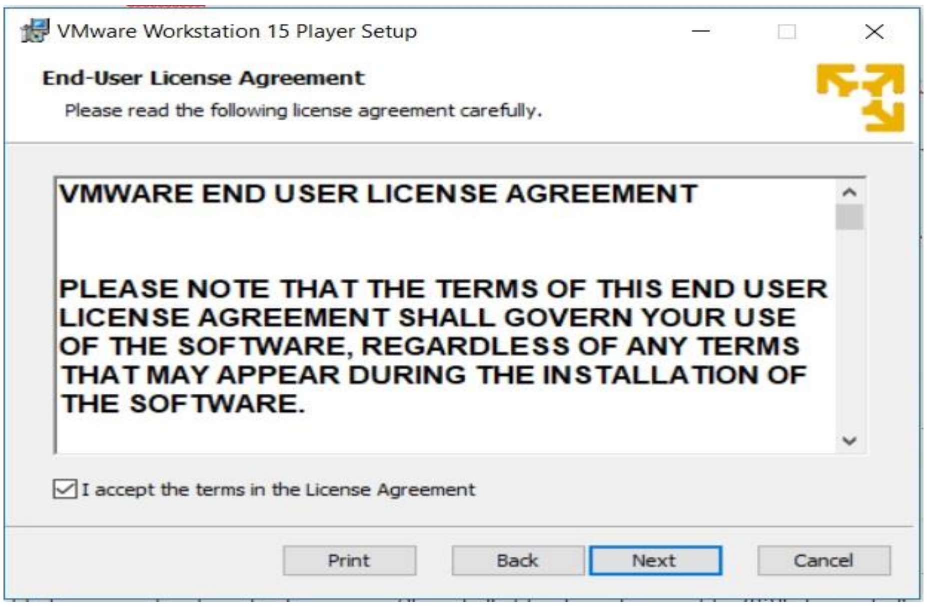
The Setup Wizard will install VMware Workstation 15 Player on your computer. Click Next to continue or Cancel to exit the Setup Wizard.

Copyright 1998-2018 VMware, Inc. All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. VMware products are covered by one or more patents listed at:

http://www.vmware.com/go/patents

VMware Player 15 Installation – Setup Wizard

Click next and accept the license terms and click next again to move on to the next screen.



VMware Workstation 15 Player Setup

End-User License Agreement

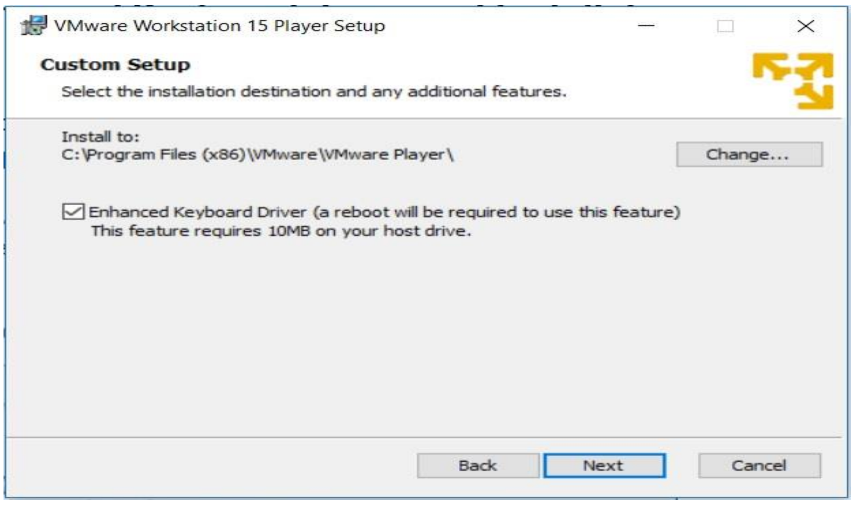
Please read the following license agreement carefully.

VMWARE END USER LICENSE AGREEMENT

PLEASE NOTE THAT THE TERMS OF THIS END USER LICENSE AGREEMENT SHALL GOVERN YOUR USE OF THE SOFTWARE, REGARDLESS OF ANY TERMS THAT MAY APPEAR DURING THE INSTALLATION OF THE SOFTWARE.

☑ I accept the terms in the License Agreement

Step 2 – Custom setup – Enhanced Keyboard driver and Installation directory


VMware Workstation 15 Player Setup

Custom Setup

Select the installation destination and any additional features.

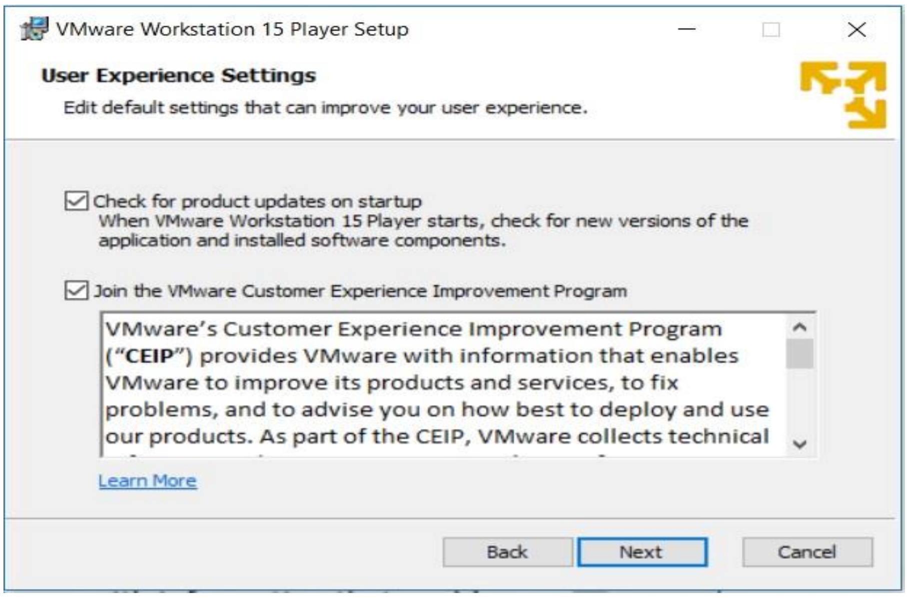
Install to:
C:\Program Files (x86)\VMware\VMware Player\

[✓] Enhanced Keyboard Driver (a reboot will be required to use this feature)
This feature requires 10MB on your host drive.

VMware Player 15 Installation – Custom Setup – Enhanced Keyboard Driver

Step 3 – User Experience Settings

Check the options for Check the product update at Startup and Join the VMware Customer Program. I normally leave it as it is. You can unchecked it if you so desire. Click next


VMware Workstation 15 Player Setup

User Experience Settings

Edit default settings that can improve your user experience.

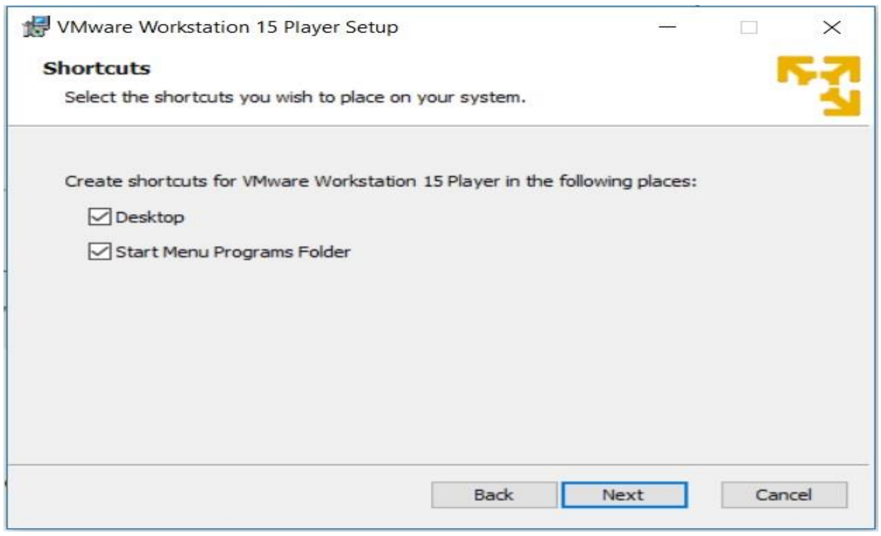
[✓] Check for product updates on startup
When VMware Workstation 15 Player starts, check for new versions of the application and installed software components.

[✓] Join the VMware Customer Experience Improvement Program

VMware's Customer Experience Improvement Program ("**CEIP**") provides VMware with information that enables VMware to improve its products and services, to fix problems, and to advise you on how best to deploy and use our products. As part of the CEIP, VMware collects technical

Step 4 – Select where the shortcuts will be installed

Check the box where the shortcut to run the application will be created. I leave it as it is. Click on next.


VMware Workstation 15 Player Setup

Shortcuts

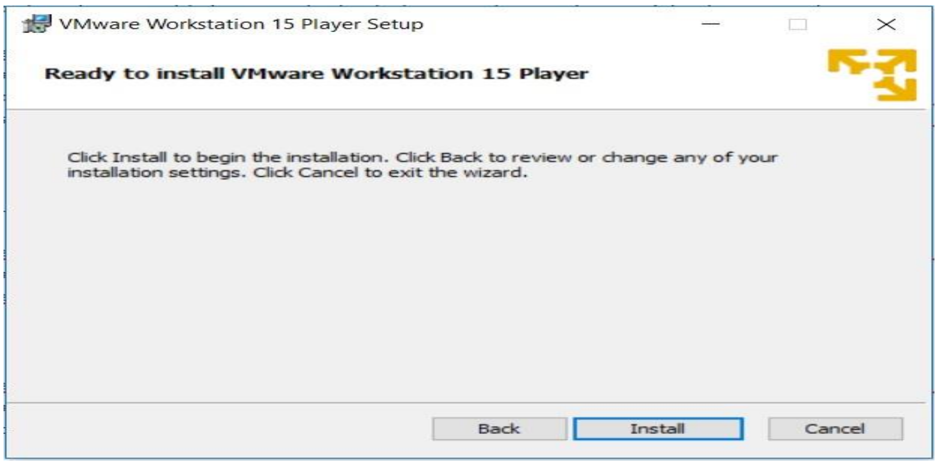
Select the shortcuts you wish to place on your system.

Create shortcuts for VMware Workstation 15 Player in the following places:

- [✓] Desktop
- [✓] Start Menu Programs Folder

Step 5 – Ready to install

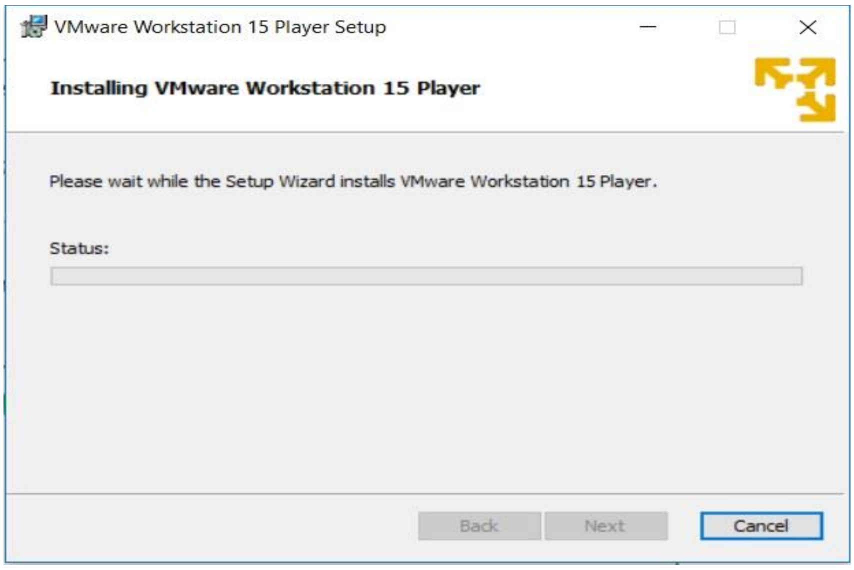
Now the installation wizard is ready to install. Click on install to begin the installation.


VMware Workstation 15 Player Setup

Ready to install VMware Workstation 15 Player

Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.

VMware Player 15 Installation – Ready to Install,Installation begins, wait for it to complete.

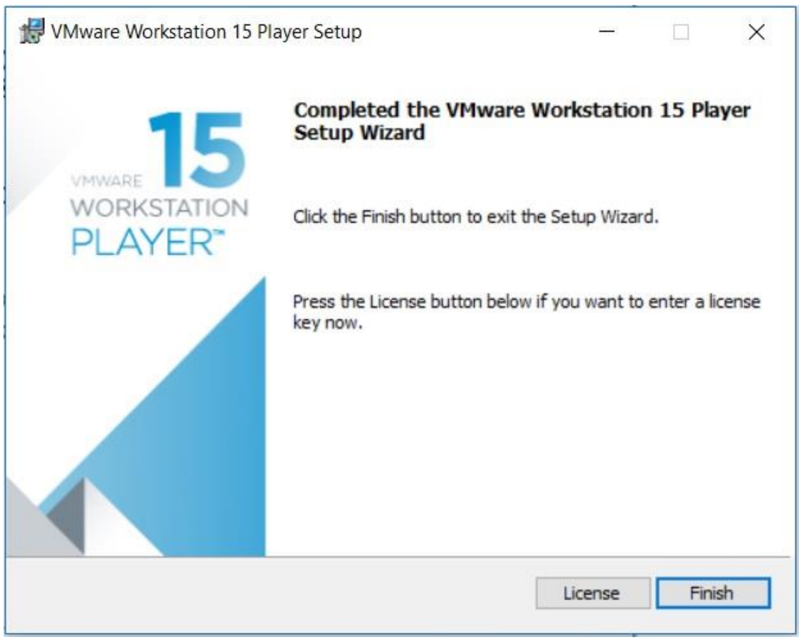

VMware Workstation 15 Player Setup

Installing VMware Workstation 15 Player

Please wait while the Setup Wizard installs VMware Workstation 15 Player.

Status:

After sometime, you will see installation compete message. You are done.


VMware Workstation 15 Player Setup

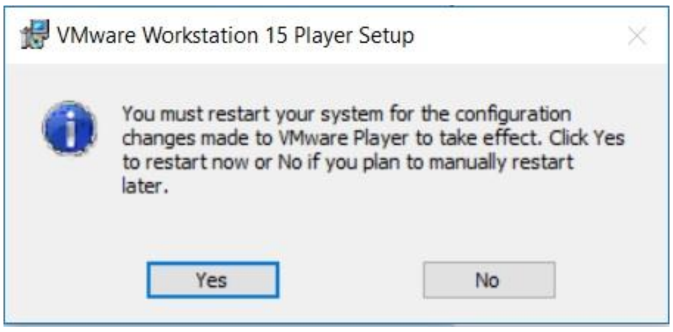
Completed the VMware Workstation 15 Player Setup Wizard

Click the Finish button to exit the Setup Wizard.

Press the License button below if you want to enter a license key now.

VMware Player 15 Installation – Installation Complete

Click on Finish to Complete the installation.


VMware Workstation 15 Player Setup

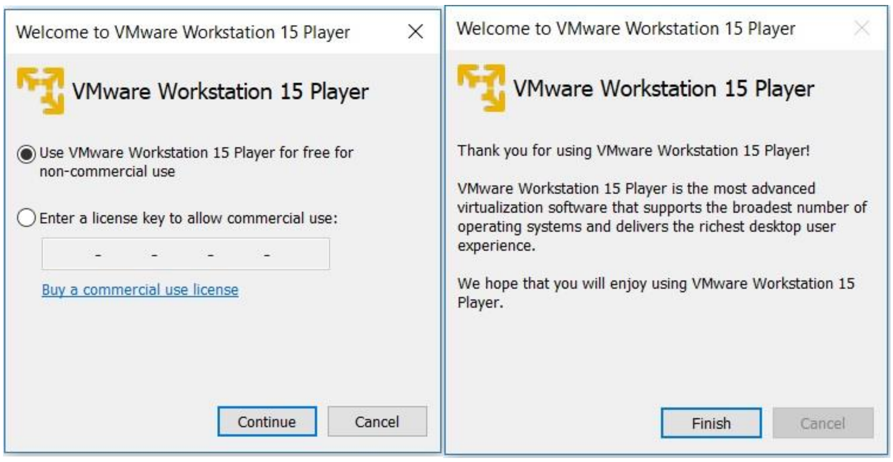
You must restart your system for the configuration changes made to VMware Player to take effect. Click Yes to restart now or No if you plan to manually restart later.

VMware Player 15 Installation – Reboot Required

Step 6 – License

Once you run the application for the first time, you will be asked for licence. Select the option Use VMware Workstation Player 15 for for free for non commercial use.

Click continue.


Welcome to VMware Workstation 15 Player

VMware Workstation 15 Player

[✓ Use VMware Workstation 15 Player for free for non-commercial use]

Enter a license key to allow commercial use:

Buy a commercial use license

Welcome to VMware Workstation 15 Player

VMware Workstation 15 Player

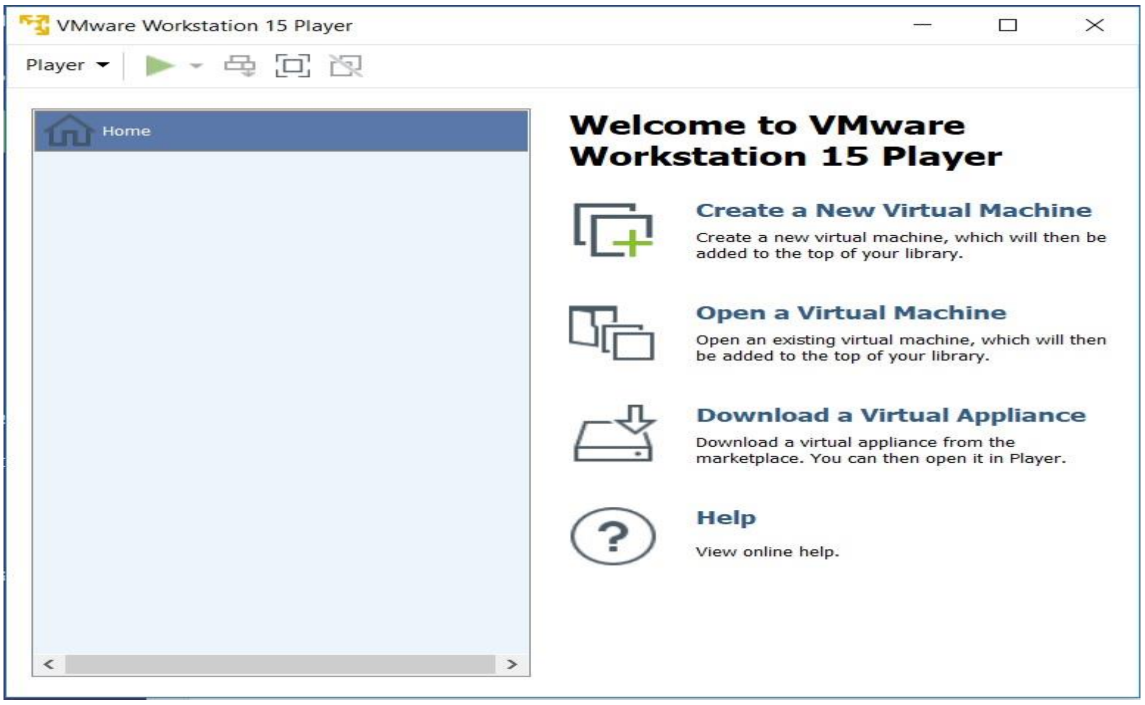
Thank you for using VMware Workstation 15 Player!

VMware Workstation 15 Player is the most advanced virtualization software that supports the broadest number of operating systems and delivers the richest desktop user experience.

We hope that you will enjoy using VMware Workstation 15 Player.

VMware Player 15 Installation – License,Click on Finish.

Now you will see VMware Workstation Player 15 ready to be used for free for non-commercial purpose.



Welcome to VMware Workstation 15 Player

Create a New Virtual Machine

Create a new virtual machine, which will then be added to the top of your library.

Open a Virtual Machine

Open an existing virtual machine, which will then be added to the top of your library.

Download a Virtual Appliance

Download a virtual appliance from the marketplace. You can then open it in Player.

Help

View online help.

VMware Player 15 – Home Screen

PRACTICAL –2

AIM:-Overview and getting started with Hadoop and Big Insights

PROCEDURE:-In this lab, we work on the InfoSphereBigInsights 3.0 Quick Start Edition VMware image.

The VMware image is set up in the following manner:

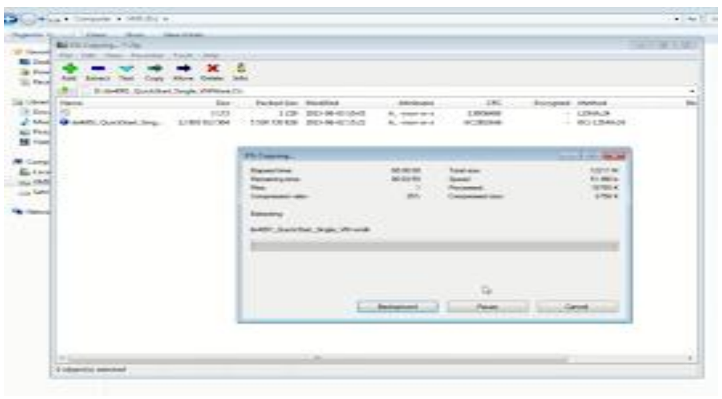
	User	Password
VM Image root account	root	password
VM Image lab user account	biadmin	biadmin
BigInsights Administrator	biadmin	Biadmin
Big SQL Administrator	bigsql	Bigsql
Lab user	biadmin	Biadmin

Property	Value
Host name	bivm.ibm.com
BigInsights Web Console URL	http://bivm.ibm.com:8080
Big SQL database name	Bigsql
Big SQL port number	51000

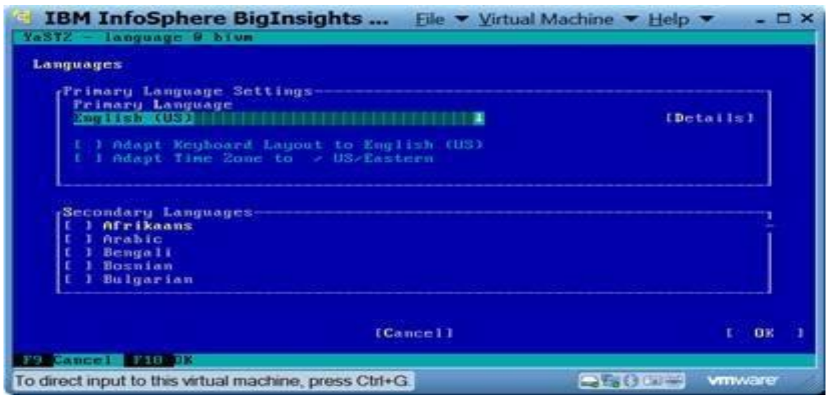
Step: To install and launch the VMware image as well as start the required services.

1. Obtain a copy of the BigInsights 3.0 Quick Start Edition VMware image from IBM's external download site (http://www-01.ibm.com/software/data/infosphere/biginsights/quick-start/downloads.html). Use the image for the single-node cluster.

2. Follow the instructions provided to decompress (unzip) the file and install the image on your laptop. Note that there is a README file with additional information.

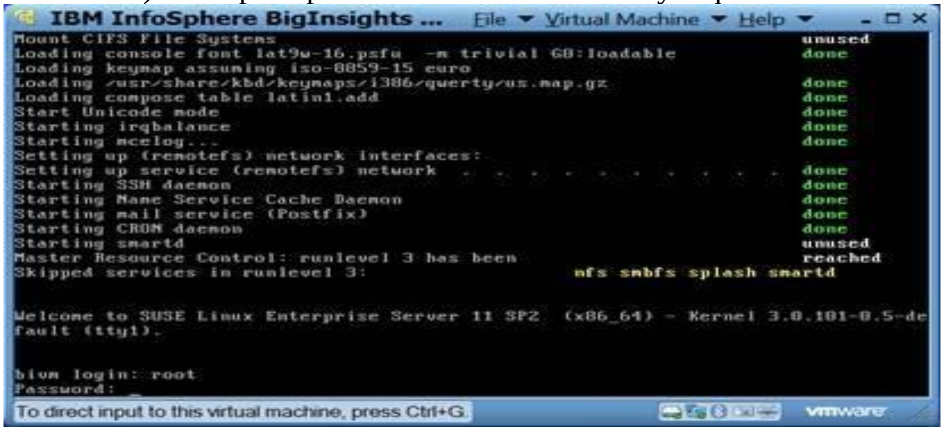


3. Install VMware player or other required software to run VMware images. Details are in the README file provided with the BigInsights VMware image.


IBM InfoSphere BigInsights ...
File
Virtual Machine
Help
To direct input to this virtual machine, press Ctrl+G.
vmware

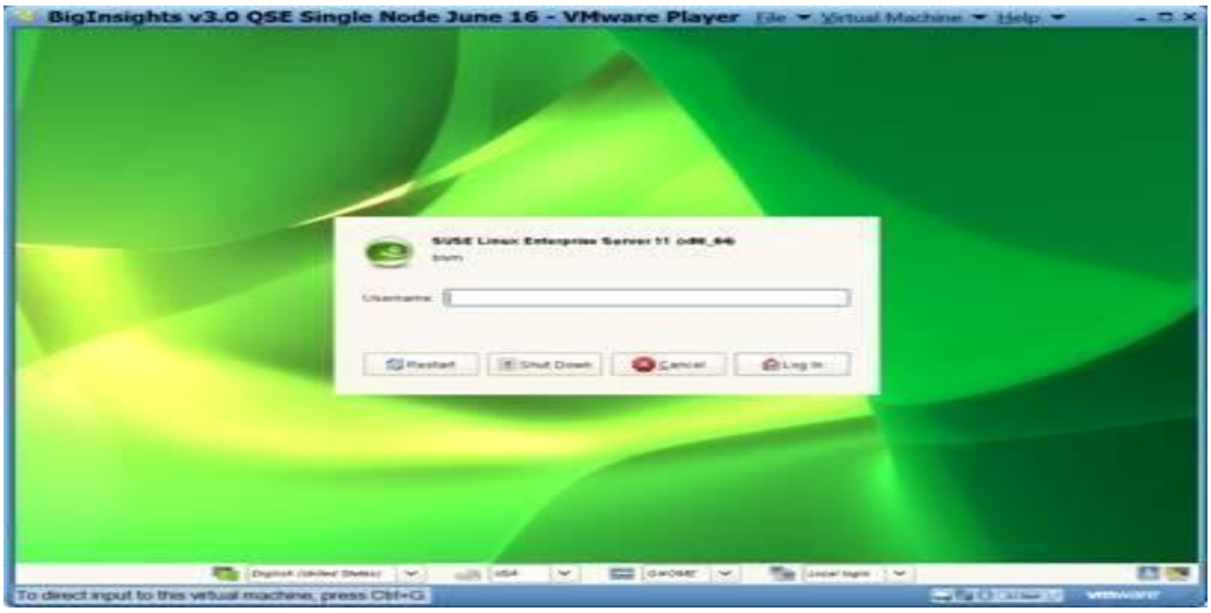
```
YaST2 - language @ bivm

Languages

┌Primary Language Settings──────────────────────────────────────────┐
│ Primary Language                                                  │
│ English (US)                                        [Details]     │
│                                                                   │
│ [ ] Adapt Keyboard Layout to English (US)                         │
│ [ ] Adapt Time Zone to  ✓ US/Eastern                              │
└───────────────────────────────────────────────────────────────────┘

┌Secondary Languages────────────────────────────────────────────────┐
│[ ] Afrikaans                                                      │
│[ ] Arabic                                                         │
│[ ] Bengali                                                        │
│[ ] Bosnian                                                        │
│[ ] Bulgarian                                                      │
└───────────────────────────────────────────────────────────────────┘

                         [Cancel]                          [ OK ]
F9 Cancel  F10 OK
```

4. Launch the VMware image. When logging in for the first time, use the root ID (with a password of password). Follow the instructions to configure your environment, accept the licensing agreement, and enter the passwords for the root and biadmin IDs (root/password and biadmin/biadmin) when prompted. This is a one-time only requirement.

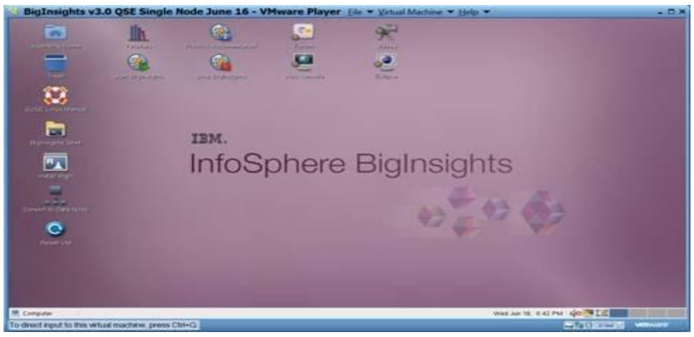

IBM InfoSphere BigInsights ...
File
Virtual Machine
Help
To direct input to this virtual machine, press Ctrl+G.
vmware

```
Mount CIFS File Systems                                                unused
Loading console font lat9w-16.psfu  -m trivial G0:loadable             done
Loading keymap assuming iso-8859-15 euro
Loading /usr/share/kbd/keymaps/i386/qwerty/us.map.gz                   done
Loading compose table latin1.add                                       done
Start Unicode mode                                                     done
Starting irqbalance                                                    done
Starting mcelog...                                                     done
Setting up (remotefs) network interfaces:
Setting up service (remotefs) network  . . . . . . . . . .             done
Starting SSH daemon                                                    done
Starting Name Service Cache Daemon                                     done
Starting mail service (Postfix)                                        done
Starting CRON daemon                                                   done
Starting smartd                                                        unused
Master Resource Control: runlevel 3 has been                           reached
Skipped services in runlevel 3:                     nfs smbfs splash smartd


Welcome to SUSE Linux Enterprise Server 11 SP2  (x86_64) - Kernel 3.0.101-0.5-de
fault (tty1).


bivm login: root
Password:
```

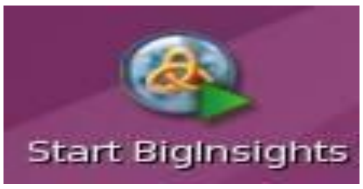
5. When the one-time configuration process is completed, you will be presented with a SUSE Linux log in screen. Log in as biadmin with a password of biadmin.


BigInsights v3.0 QSE Single Node June 16 - VMware Player
File
Virtual Machine
Help
SUSE Linux Enterprise Server 11 (x86_64)
bivm
Username:
Restart
Shut Down
Cancel
Log In
To direct input to this virtual machine, press Ctrl+G.
vmware

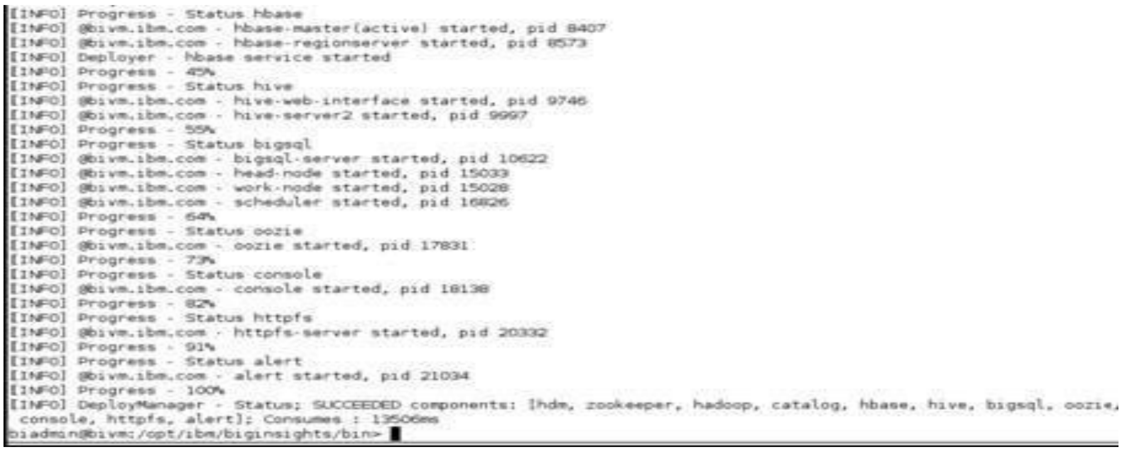
6. Verify that your screen that appears.


BigInsights v3.0 QSE Single Node June 16 - VMware Player
IBM.
InfoSphere BigInsights
To direct input to this virtual machine, press Ctrl+G.

7. Click Start BigInsights to start all required services. Wait until the operation completes. This may take several minutes, depending on your machine's resources.



8. Verify that all required BigInsights services are up and running. From a terminal window, issue this command:$BIGINSIGHTS_HOME/bin/status.sh.



```
[INFO] Progress - Status hbase
[INFO] @bivm.ibm.com - hbase-master(active) started, pid 8407
[INFO] @bivm.ibm.com - hbase-regionserver started, pid 8573
[INFO] Deployer - hbase service started
[INFO] Progress - 45%
[INFO] Progress - Status hive
[INFO] @bivm.ibm.com - hive-web-interface started, pid 9746
[INFO] @bivm.ibm.com - hive-server2 started, pid 9997
[INFO] Progress - 55%
[INFO] Progress - Status bigsql
[INFO] @bivm.ibm.com - bigsql-server started, pid 10622
[INFO] @bivm.ibm.com - head-node started, pid 15033
[INFO] @bivm.ibm.com - work-node started, pid 15028
[INFO] @bivm.ibm.com - scheduler started, pid 16826
[INFO] Progress - 64%
[INFO] Progress - Status oozie
[INFO] @bivm.ibm.com - oozie started, pid 17831
[INFO] Progress - 73%
[INFO] Progress - Status console
[INFO] @bivm.ibm.com - console started, pid 18138
[INFO] Progress - 82%
[INFO] Progress - Status httpfs
[INFO] @bivm.ibm.com - httpfs-server started, pid 20332
[INFO] Progress - 91%
[INFO] Progress - Status alert
[INFO] @bivm.ibm.com - alert started, pid 21034
[INFO] Progress - 100%
[INFO] DeployManager - Status; SUCCEEDED components: [hdm, zookeeper, hadoop, catalog, hbase, hive, bigsql, oozie,
 console, httpfs, alert]; Consumes : 13506ms
biadmin@bivm:/opt/ibm/biginsights/bin> 
```

9. Inspect the results, a subset of which are shown below. Verify that, at a minimum, the following components started successfully:hdm, zookeeper, hadoop, catalog, hive, bigsql, oozie, console, and httpfs.

PRACTICAL –3

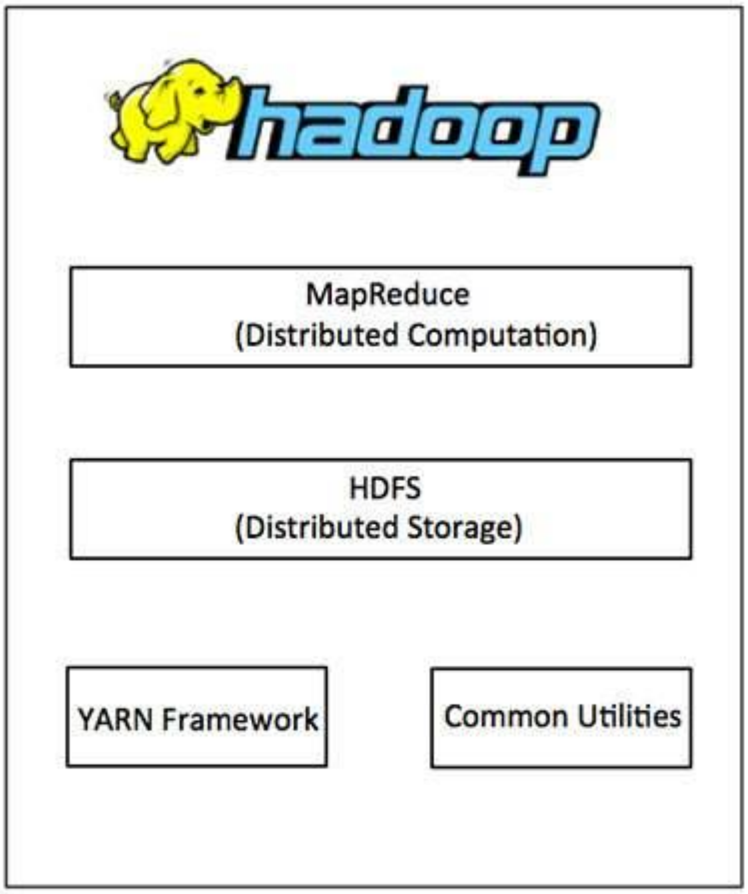
AIM:-Introduction to Hadoop.

PROCEDURE:-Hadoop is an Apache open source framework written in java that allows distributed processing of large datasets across clusters of computers using simple programming models. The Hadoop framework application works in an environment that provides distributed *storage* and *computation* across clusters of computers. Hadoop is designed to scale up from single server to thousands of machines, each offering local computation and storage.

Hadoop Architecture

At its core, Hadoop has two major layers namely −

- Processing/Computation layer (MapReduce), and
- Storage layer (Hadoop Distributed File System).


hadoop
MapReduce (Distributed Computation)
HDFS (Distributed Storage)
YARN Framework
Common Utilities

MapReduce

MapReduce is a parallel programming model for writing distributed applications devised at Google for efficient processing of large amounts of data (multi-terabyte data-sets), on large clusters (thousands of nodes) of commodity hardware in a reliable, fault-tolerant manner. The MapReduce program runs on Hadoop which is an Apache open-source framework.

Hadoop Distributed File System

The Hadoop Distributed File System (HDFS) is based on the Google File System (GFS) and provides a distributed file system that is designed to run on commodity hardware. It has many similarities with existing distributed file systems. However, the differences from other distributed file systems are significant. It is highly fault-tolerant and is designed to be deployed on low-cost hardware. It provides high throughput access to application data and is suitable for applications having large datasets.

Apart from the above-mentioned two core components, Hadoop framework also includes the following two modules −

- **Hadoop Common** − These are Java libraries and utilities required by other Hadoop modules.
- **Hadoop YARN** − This is a framework for job scheduling and cluster resource management.

PRACTICAL –3

AIM:-WAP to count words in a file using a Map Reduce application

PROCEDURE:-Word Count MapReduce Program in Hadoop

The first MapReduce program most of the people write after installing Hadoop is invariably the word count MapReduce program.
The detailed steps for writing word count MapReduce program in Java, IDE used is Eclipse.

Creating and copying file

1. If you already have a file in HDFS which you want to use as input then you can skip this step.
2. First thing is to create a file which will be used as input and copy it to HDFS.
3. Let's say you have a file wordcount.txt with the following content.
4. Hello wordcount MapReduce Hadoop program.
5. This is my first MapReduce program.
6. You want to copy this file to /user/process directory with in HDFS. If that path doesn't exist then you need to create those directories first.
   hdfsdfs -mkdir -p /user/process
7. Then copy the file wordcount.txt to this directory.
   hdfsdfs -put /netjs/MapReduce/wordcount.txt /user/process

Word count example MapReduce code in Java

1. Write your wordcount MapReduce code. WordCount example reads text files and counts the frequency of the words. Each mapper takes a line of the input file as input and breaks it into words. It then emits a key/value pair of the word (In the form of (word, 1)) and each reducer sums the counts for each word and emits a single key/value with the word and sum.
2. In the word count MapReduce code there is a Mapper class (MyMapper) with map function and a Reducer class (MyReducer) with a reduce function.
3. You will also need to add at least the following Hadoop jars so that your code can compile. You will find these jars inside the /share/hadoop directory of your Hadoop installation. With in /share/hadoop path look in hdfs, mapreduce and common directories for required jars.

hadoop-common-2.9.0.jar
hadoop-hdfs-2.9.0.jar
hadoop-hdfs-client-2.9.0.jar
hadoop-mapreduce-client-core-2.9.0.jar
hadoop-mapreduce-client-common-2.9.0.jar
hadoop-mapreduce-client-jobclient-2.9.0.jar
hadoop-mapreduce-client-hs-2.9.0.jar
hadoop-mapreduce-client-app-2.9.0.jar
commons-io-2.4.jar

Creating jar of your wordcount MapReduce code

Once you are able to compile your code you need to create jar file. In the eclipse IDE righ click on your Java program and select Export – Java – jar file.

Running the code

You can use the following command to run the program in your hadoop installation directory.

bin/hadoop jar /netjs/MapReduce/wordcount.jar org.netjs.WordCount /user/process /user/out

/netjs/MapReduce/wordcount.jar is the path to your jar file.

org.netjs.WordCount is the fully qualified path to your Java program class.

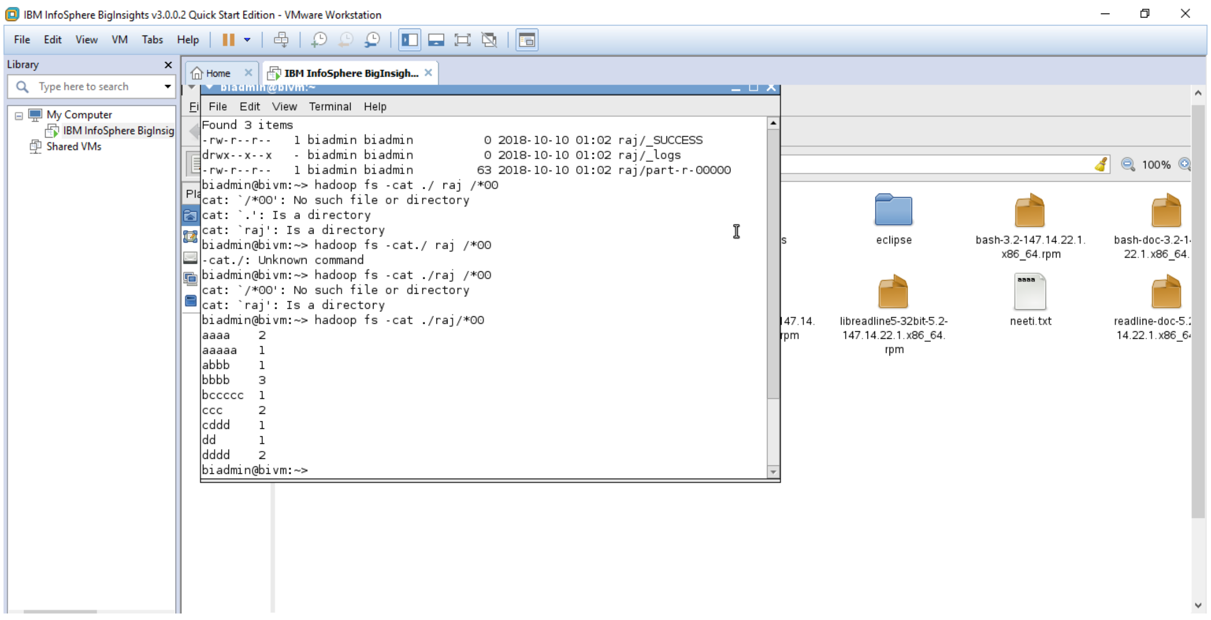
/user/process – path to input directory.

/user/out – path to output directory.

One your word count MapReduce program is succesfully executed you can verify the output file.

```
hdfsdfs -ls /user/out
Found 2 items
-rw-r--r--   1 netjs supergroup          0 2018-02-27 13:37 /user/out/_SUCCESS
-rw-r--r--   1 netjs supergroup         77 2018-02-27 13:37 /user/out/part-r-00000
```

As you can see Hadoop framework creates output files using part-r-xxxx format. Since only one reducer is used here so there is only one output file part-r-00000.


IBM InfoSphere BigInsights v3.0.0.2 Quick Start Edition - VMware Workstation
File Edit View VM Tabs Help
Library
Type here to search
My Computer
IBM InfoSphere BigInsig
Shared VMs
Home
IBM InfoSphere BigInsigh...
biadmin@bivm:~
File Edit View Terminal Help
eclipse
bash-3.2-147.14.22.1.x86_64.rpm
libreadline5-32bit-5.2-147.14.22.1.x86_64.rpm
neeti.txt
100%

```
Found 3 items
-rw-r--r--   1 biadmin biadmin          0 2018-10-10 01:02 raj/_SUCCESS
drwx--x--x   - biadmin biadmin          0 2018-10-10 01:02 raj/_logs
-rw-r--r--   1 biadmin biadmin         63 2018-10-10 01:02 raj/part-r-00000
biadmin@bivm:~> hadoop fs -cat ./ raj /*00
cat: `/*00': No such file or directory
cat: `.': Is a directory
cat: `raj': Is a directory
biadmin@bivm:~> hadoop fs -cat./ raj /*00
-cat./: Unknown command
biadmin@bivm:~> hadoop fs -cat ./raj /*00
cat: `/*00': No such file or directory
cat: `raj': Is a directory
biadmin@bivm:~> hadoop fs -cat ./raj/*00
aaaa    2
aaaaa   1
abbb    1
bbbb    3
bcccccc 1
ccc     2
cddd    1
dd      1
dddd    2
biadmin@bivm:~>
```

PRACTICAL-4

AIM:-HBase shell command

PROCEDURE:-HBase Shell

HBase contains a shell using which you can communicate with HBase. HBase uses the Hadoop File System to store its data. It will have a master server and region servers. The data storage will be in the form of regions (tables). These regions will be split up and stored in region servers.

The master server manages these region servers and all these tasks take place on HDFS. Given below are some of the commands supported by HBase Shell.

Creating a Table using HBase Shell

You can create a table using the create command, here you must specify the table name and the Column Family name. The syntax to create a table in HBase shell is shown on left page.

Listing a Table using HBase Shell

list is the command that is used to list all the tables in HBase. Given on the left page is the syntax of the list command.

Disabling a Table using HBase Shell

To delete a table or change its settings, you need to first disable the table using the disable command. You can re-enable it using the enable command.

Given below is the syntax to disable a table:

Enabling a Table using HBase Shell

Syntax to enable a table:

describe

This command returns the description of the table. Its syntax is as follows:

alter

Alter is the command used to make changes to an existing table. Using this command, you can change the maximum number of cells of a column family, set and delete table scope operators, and delete a column family from a table.

Changing the Maximum Number of Cells of a Column Family

Given below is the syntax to change the maximum number of cells of a column family.

Deleting a Specific Cell in a Table

Using the delete command, you can delete a specific cell in a table. The syntax of delete command is as follows:

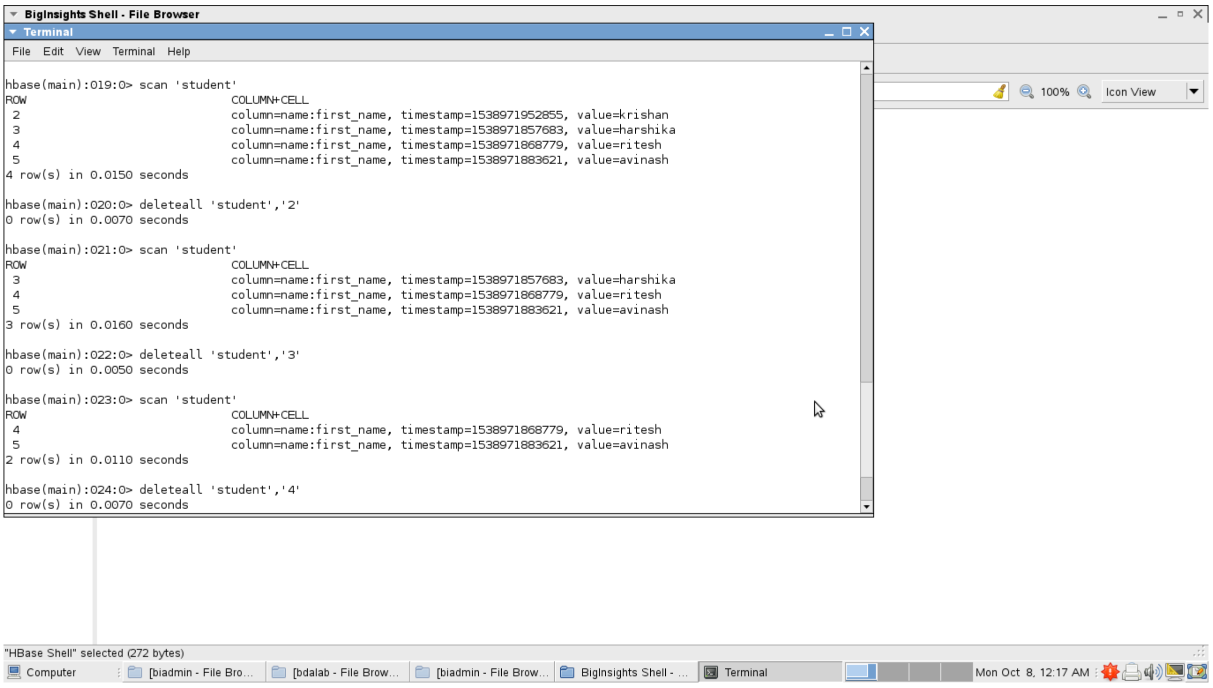
Inserting Data using HBase Shell

This chapter demonstrates how to create data in an HBase table. To create data in an HBase table, the following commands and methods are used:

- put command,
- add() method of Put class, and
- put() method of HTable class.
-

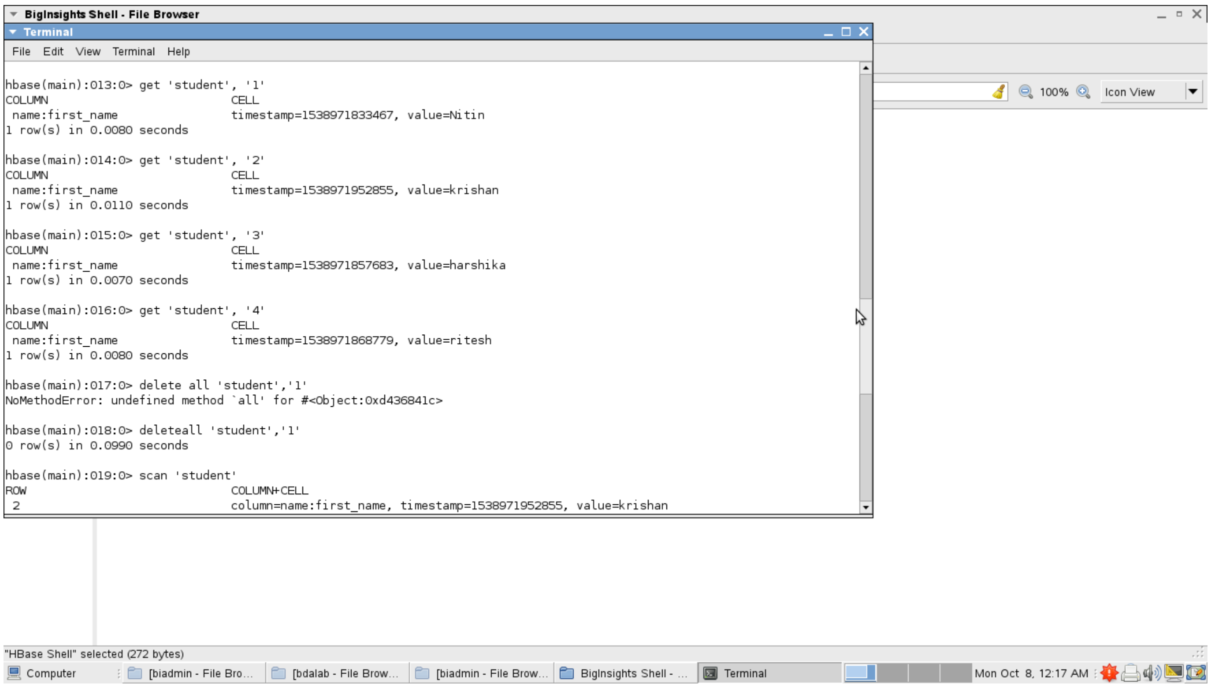
Inserting the First Row

Let us insert the first-row values into the emp table as shown on the left page.


BigInsights Shell - File Browser
Terminal
File Edit View Terminal Help
100%
Icon View

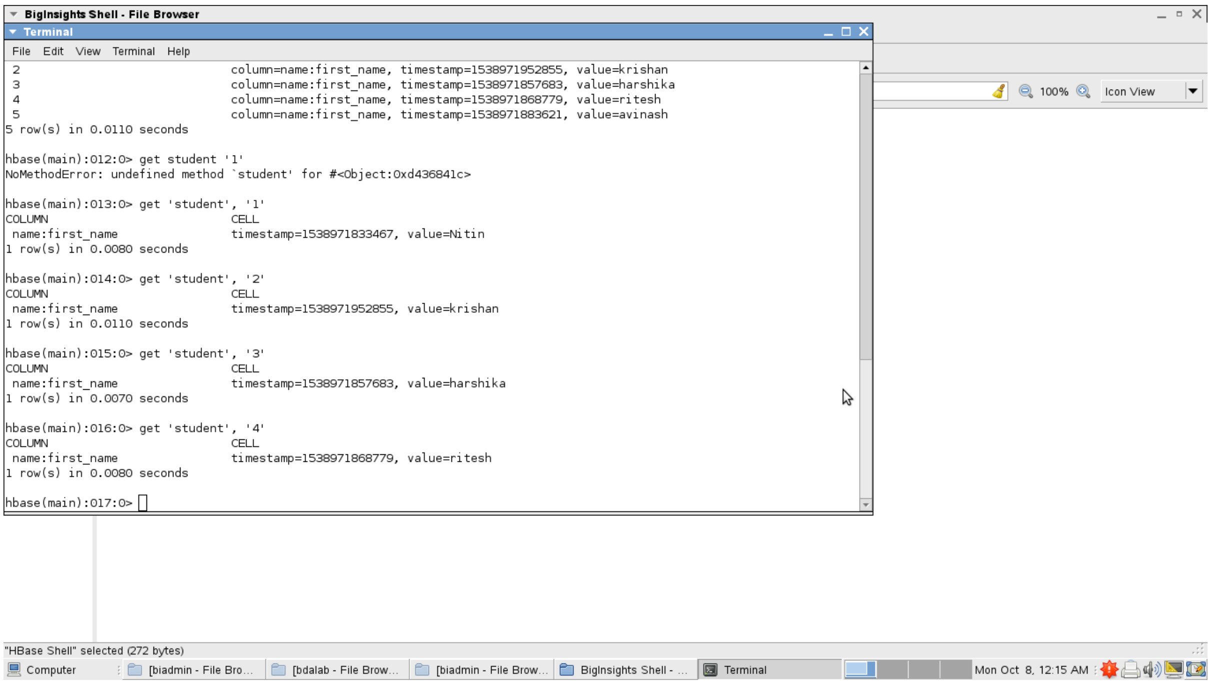
```
hbase(main):019:0> scan 'student'
ROW                              COLUMN+CELL
 2                               column=name:first_name, timestamp=1538971952855, value=krishan
 3                               column=name:first_name, timestamp=1538971857683, value=harshika
 4                               column=name:first_name, timestamp=1538971868779, value=ritesh
 5                               column=name:first_name, timestamp=1538971883621, value=avinash
4 row(s) in 0.0150 seconds

hbase(main):020:0> deleteall 'student','2'
0 row(s) in 0.0070 seconds

hbase(main):021:0> scan 'student'
ROW                              COLUMN+CELL
 3                               column=name:first_name, timestamp=1538971857683, value=harshika
 4                               column=name:first_name, timestamp=1538971868779, value=ritesh
 5                               column=name:first_name, timestamp=1538971883621, value=avinash
3 row(s) in 0.0160 seconds

hbase(main):022:0> deleteall 'student','3'
0 row(s) in 0.0050 seconds

hbase(main):023:0> scan 'student'
ROW                              COLUMN+CELL
 4                               column=name:first_name, timestamp=1538971868779, value=ritesh
 5                               column=name:first_name, timestamp=1538971883621, value=avinash
2 row(s) in 0.0110 seconds

hbase(main):024:0> deleteall 'student','4'
0 row(s) in 0.0070 seconds
```

"HBase Shell" selected (272 bytes)

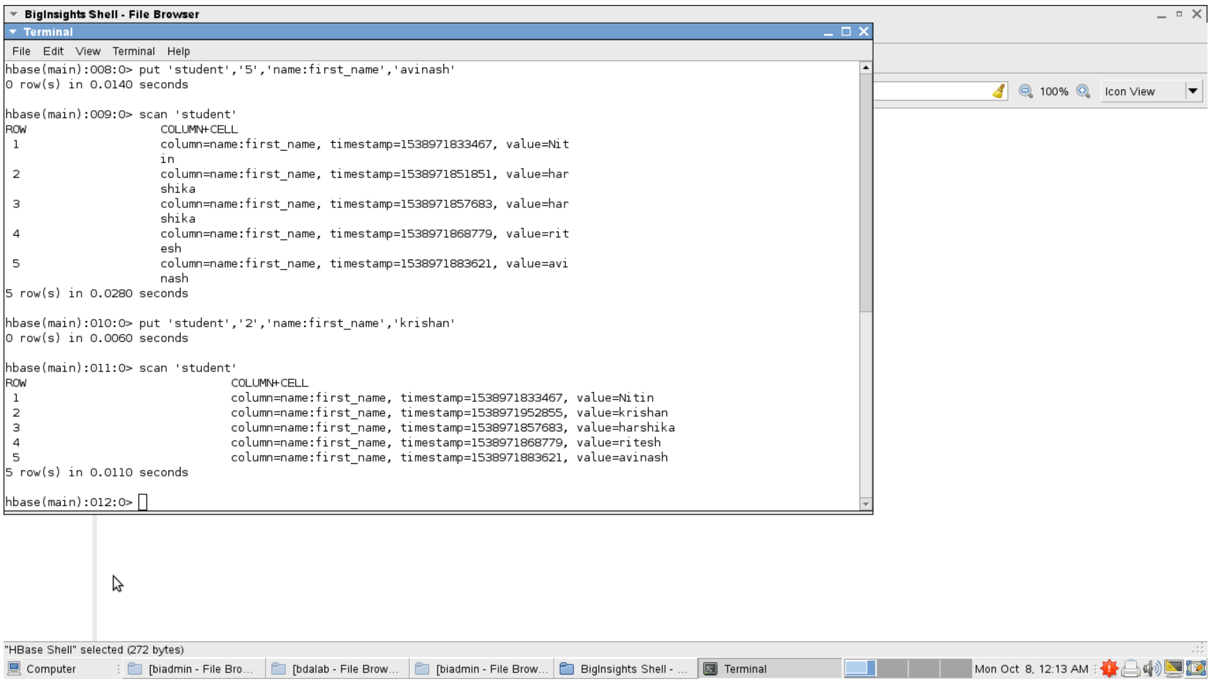

BigInsights Shell - File Browser
Terminal
File Edit View Terminal Help
100%
Icon View

```
hbase(main):013:0> get 'student', '1'
COLUMN                           CELL
 name:first_name                 timestamp=1538971833467, value=Nitin
1 row(s) in 0.0080 seconds

hbase(main):014:0> get 'student', '2'
COLUMN                           CELL
 name:first_name                 timestamp=1538971952855, value=krishan
1 row(s) in 0.0110 seconds

hbase(main):015:0> get 'student', '3'
COLUMN                           CELL
 name:first_name                 timestamp=1538971857683, value=harshika
1 row(s) in 0.0070 seconds

hbase(main):016:0> get 'student', '4'
COLUMN                           CELL
 name:first_name                 timestamp=1538971868779, value=ritesh
1 row(s) in 0.0080 seconds

hbase(main):017:0> delete all 'student','1'
NoMethodError: undefined method `all' for #<Object:0xd436841c>

hbase(main):018:0> deleteall 'student','1'
0 row(s) in 0.0990 seconds

hbase(main):019:0> scan 'student'
ROW                              COLUMN+CELL
 2                               column=name:first_name, timestamp=1538971952855, value=krishan
```

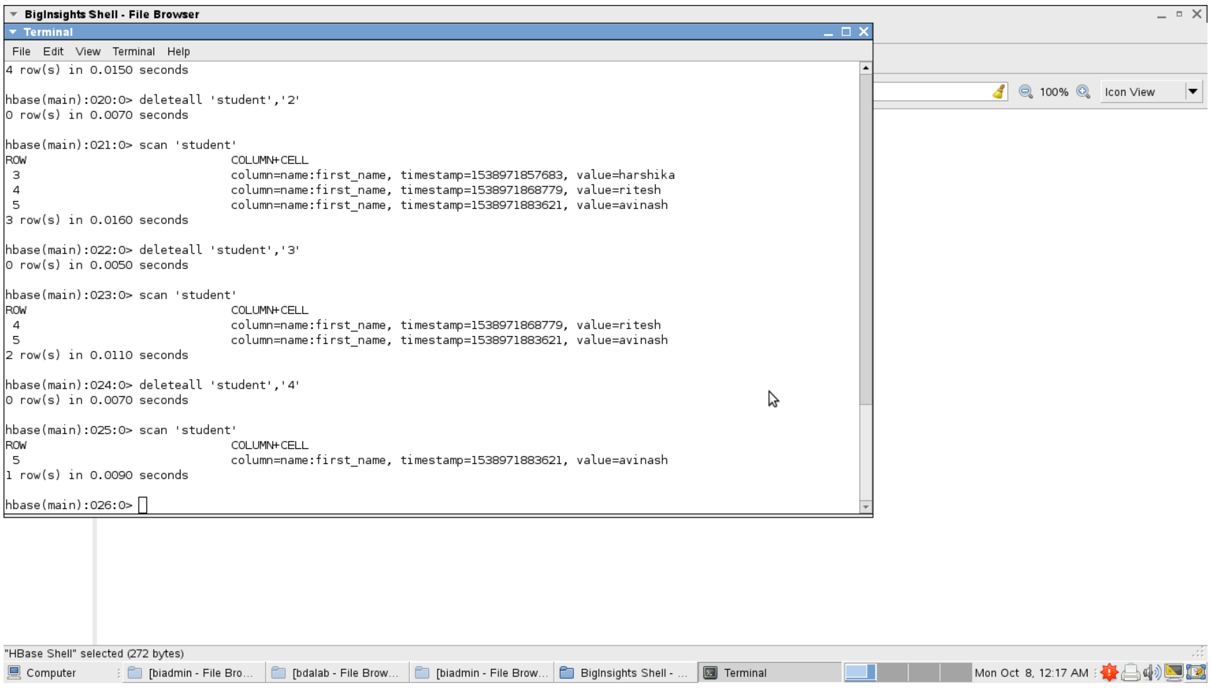
"HBase Shell" selected (272 bytes)



```
 2                             column=name:first_name, timestamp=1538971952855, value=krishan
 3                             column=name:first_name, timestamp=1538971857683, value=harshika
 4                             column=name:first_name, timestamp=1538971868779, value=ritesh
 5                             column=name:first_name, timestamp=1538971883621, value=avinash
5 row(s) in 0.0110 seconds

hbase(main):012:0> get student '1'
NoMethodError: undefined method `student' for #<Object:0xd436841c>

hbase(main):013:0> get 'student', '1'
COLUMN                         CELL
 name:first_name               timestamp=1538971833467, value=Nitin
1 row(s) in 0.0080 seconds

hbase(main):014:0> get 'student', '2'
COLUMN                         CELL
 name:first_name               timestamp=1538971952855, value=krishan
1 row(s) in 0.0110 seconds

hbase(main):015:0> get 'student', '3'
COLUMN                         CELL
 name:first_name               timestamp=1538971857683, value=harshika
1 row(s) in 0.0070 seconds

hbase(main):016:0> get 'student', '4'
COLUMN                         CELL
 name:first_name               timestamp=1538971868779, value=ritesh
1 row(s) in 0.0080 seconds

hbase(main):017:0>
```



```
hbase(main):008:0> put 'student','5','name:first_name','avinash'
0 row(s) in 0.0140 seconds

hbase(main):009:0> scan 'student'
ROW                    COLUMN+CELL
 1                     column=name:first_name, timestamp=1538971833467, value=Nit
                       in
 2                     column=name:first_name, timestamp=1538971851851, value=har
                       shika
 3                     column=name:first_name, timestamp=1538971857683, value=har
                       shika
 4                     column=name:first_name, timestamp=1538971868779, value=rit
                       esh
 5                     column=name:first_name, timestamp=1538971883621, value=avi
                       nash
5 row(s) in 0.0280 seconds

hbase(main):010:0> put 'student','2','name:first_name','krishan'
0 row(s) in 0.0060 seconds

hbase(main):011:0> scan 'student'
ROW                            COLUMN+CELL
 1                             column=name:first_name, timestamp=1538971833467, value=Nitin
 2                             column=name:first_name, timestamp=1538971952855, value=krishan
 3                             column=name:first_name, timestamp=1538971857683, value=harshika
 4                             column=name:first_name, timestamp=1538971868779, value=ritesh
 5                             column=name:first_name, timestamp=1538971883621, value=avinash
5 row(s) in 0.0110 seconds

hbase(main):012:0>
```


BigInsights Shell - File Browser
Terminal
File Edit View Terminal Help
100%
Icon View

```
4 row(s) in 0.0150 seconds

hbase(main):020:0> deleteall 'student','2'
0 row(s) in 0.0070 seconds

hbase(main):021:0> scan 'student'
ROW                              COLUMN+CELL
 3                               column=name:first_name, timestamp=1538971857683, value=harshika
 4                               column=name:first_name, timestamp=1538971868779, value=ritesh
 5                               column=name:first_name, timestamp=1538971883621, value=avinash
3 row(s) in 0.0160 seconds

hbase(main):022:0> deleteall 'student','3'
0 row(s) in 0.0050 seconds

hbase(main):023:0> scan 'student'
ROW                              COLUMN+CELL
 4                               column=name:first_name, timestamp=1538971868779, value=ritesh
 5                               column=name:first_name, timestamp=1538971883621, value=avinash
2 row(s) in 0.0110 seconds

hbase(main):024:0> deleteall 'student','4'
0 row(s) in 0.0070 seconds

hbase(main):025:0> scan 'student'
ROW                              COLUMN+CELL
 5                               column=name:first_name, timestamp=1538971883621, value=avinash
1 row(s) in 0.0090 seconds

hbase(main):026:0> 
```

PRACTICAL-5

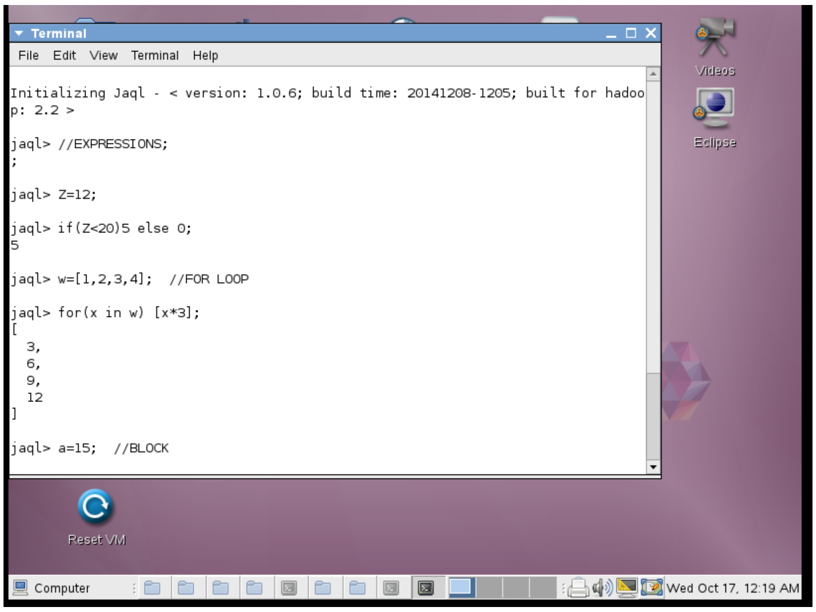
AIM:-JAQL shell commands

PROCEDURE:-Jaql basics

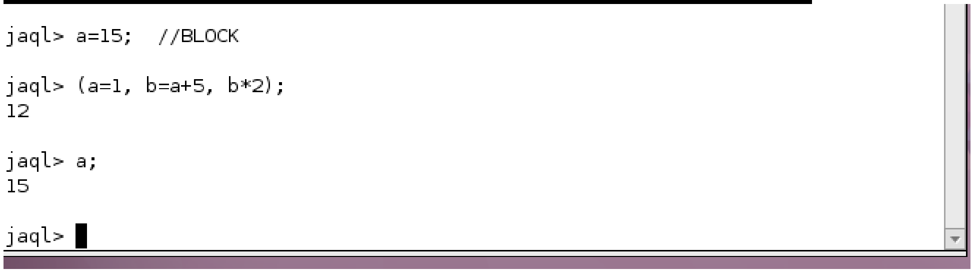
Statement, Assignment and Comments

Double and single quotes are treated the same. Semicolon terminates a statement.

```
jaql> "Hello world";
"Hello world"
jaql> a = 10*2;
jaql> a;
20
jaql> // This is a comment
jaql> /* and this is also
        a comment */
```



```
Initializing Jaql - < version: 1.0.6; build time: 20141208-1205; built for hadoo
p: 2.2 >

jaql> //EXPRESSIONS;
;

jaql> Z=12;

jaql> if(Z<20)5 else 0;
5

jaql> w=[1,2,3,4];  //FOR LOOP

jaql> for(x in w) [x*3];
[
  3,
  6,
  9,
  12
]

jaql> a=15;  //BLOCK
```



```
jaql> a=15;  //BLOCK

jaql> (a=1, b=a+5, b*2);
12

jaql> a;
15

jaql>
```

Data Types

Jaql is a loosely typed functional language, with lazy evaluation. Type is usually inferred by how a value is provided. Many types have a function of the same name to force conversions of a value or variable (e.g. string(), double()).

null – null

boolean – true, false

string – “hi”

long – 10
double – 10.2, 10d, 10e-2
array – [1, 2, 3]
record – {a : 1, b : 2}
others as jaql extensions – decfloat, binary, date, schema, function, comparator, regex

```
jaql>num = 10;
jaql> array = [1, 2, 3, "hello", 4, {color: "red"}];
jaql> array;
[
  1,
  2,
  3,
  "hello"
  ...
```

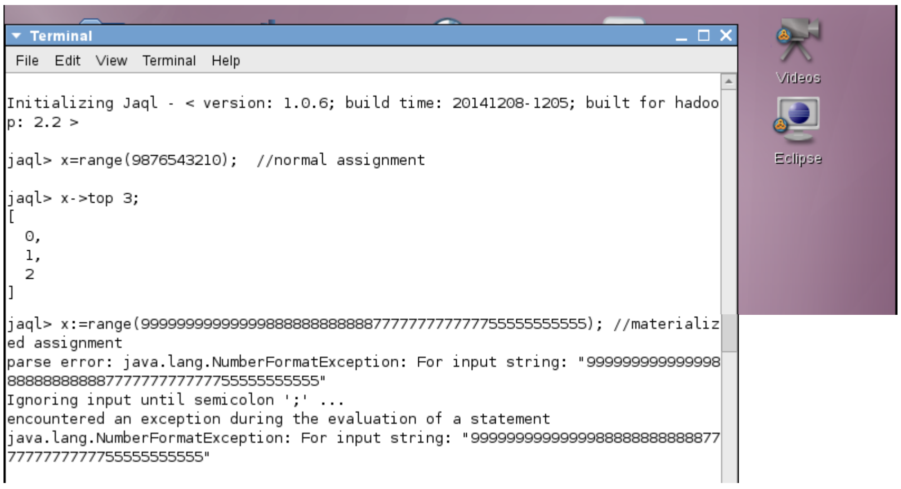
Operators
arithmetic (+, -, /, *)
boolean (and, or, not)
comparison (==, !=, <, >, in, isnull)

Arrays
Arrays can be accessed with the [] operator.

```
jaql> a = [1, 2, 3];
jaql>a[1]; //retrieves start from zero
2
jaql> a[1:2]; //retrieve a range/subset
[2,3]
```

But you can’t change a value, you need to use function replaceElement().

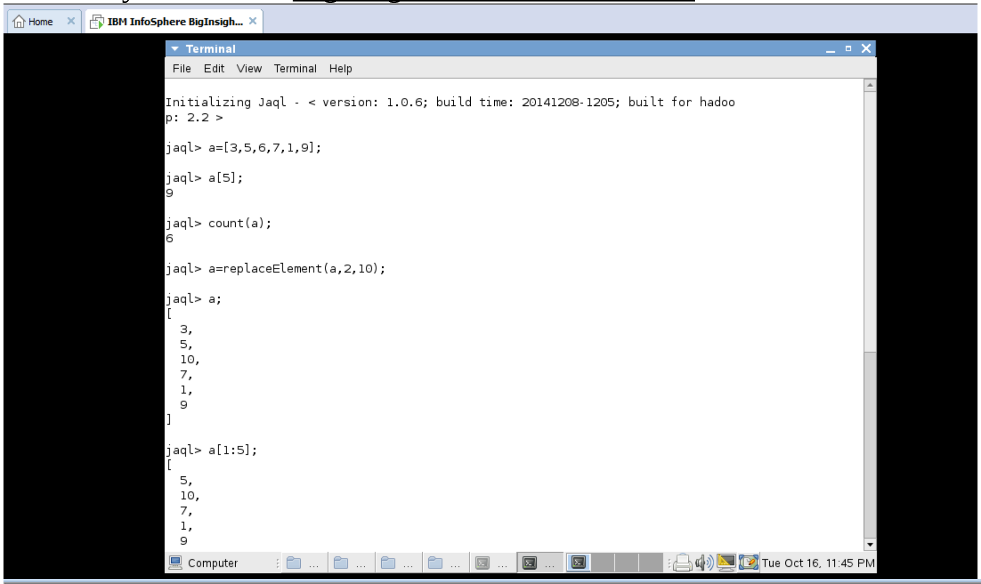
```
jaql> a = replaceElement(a, 1, 10);
jaql> a[1];
10
```



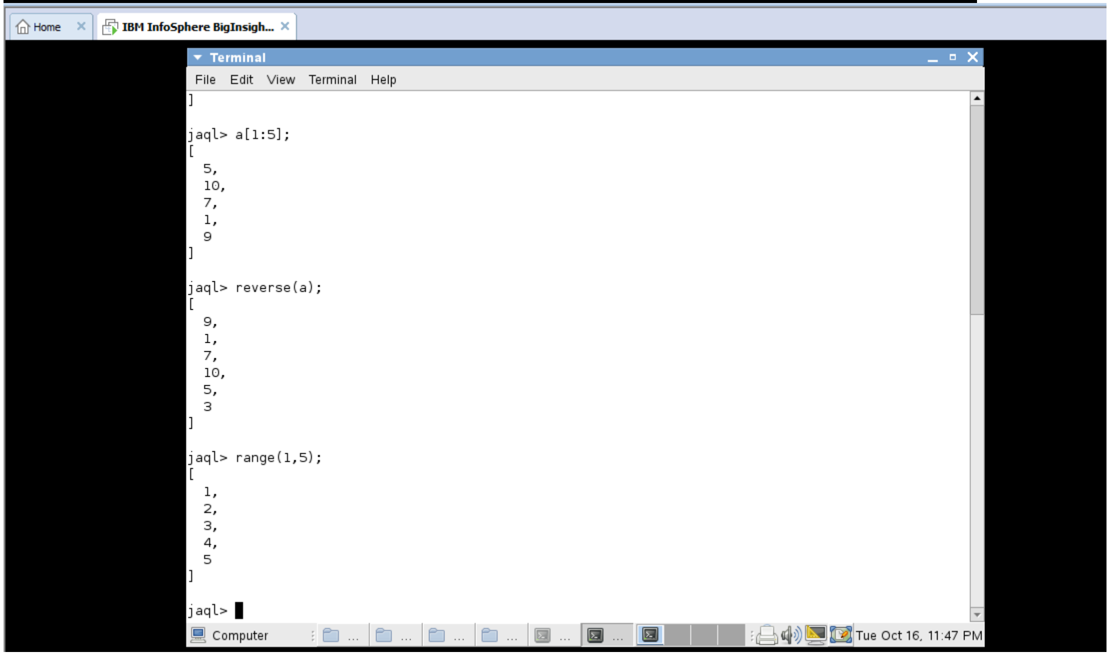
```
Initializing Jaql - < version: 1.0.6; build time: 20141208-1205; built for hadoo
p: 2.2 >

jaql> x=range(9876543210);  //normal assignment

jaql> x->top 3;
[
  0,
  1,
  2
]

jaql> x:=range(99999999999999888888888888777777777777755555555555); //materializ
ed assignment
parse error: java.lang.NumberFormatException: For input string: "99999999999998
888888888887777777777777555555555555"
Ignoring input until semicolon ';' ...
encountered an exception during the evaluation of a statement
java.lang.NumberFormatException: For input string: "9999999999999988888888888877
7777777777755555555555"
```

Other array functions
count() – count(['a', 'b', 'c']); returns 3
index() – index(['a', 'b', 'c'], 1); returns "b"

replaceElement() – replaceElement(['a', 'b', 'c'], 1, 'z'); returns ["a", "z", "c"]
slice() – index(['a', 'b', 'c', 'd'], 1, 2); returns ["b", "c"]
reverse() – reverse(['a', 'b', 'c']); returns ["c", "b", "a"]
range() – range(2, 5); returns [2, 3, 4, 5]
And many more – see BigInsights Information Center



```
Initializing Jaql - < version: 1.0.6; build time: 20141208-1205; built for hadoo
p: 2.2 >

jaql> a=[3,5,6,7,1,9];

jaql> a[5];
9

jaql> count(a);
6

jaql> a=replaceElement(a,2,10);

jaql> a;
[
  3,
  5,
  10,
  7,
  1,
  9
]

jaql> a[1:5];
[
  5,
  10,
  7,
  1,
  9
```



```
]

jaql> a[1:5];
[
  5,
  10,
  7,
  1,
  9
]

jaql> reverse(a);
[
  9,
  1,
  7,
  10,
  5,
  3
]

jaql> range(1,5);
[
  1,
  2,
  3,
  4,
  5
]

jaql> 
```

Records
Records are delineated by {} and contains a comma separated list of name:value pairs. Fields are then accessed by "." operator.
jaql> a = { name : "scott", age : 42, children : ["jake", "sam"] };
jaql> a.name;
"scott"
jaql>a.children[0];
"jake"
Again you cannot change an existing record, but you can produce a new one:
jaql> a = { a.name, age : 37, a.children };

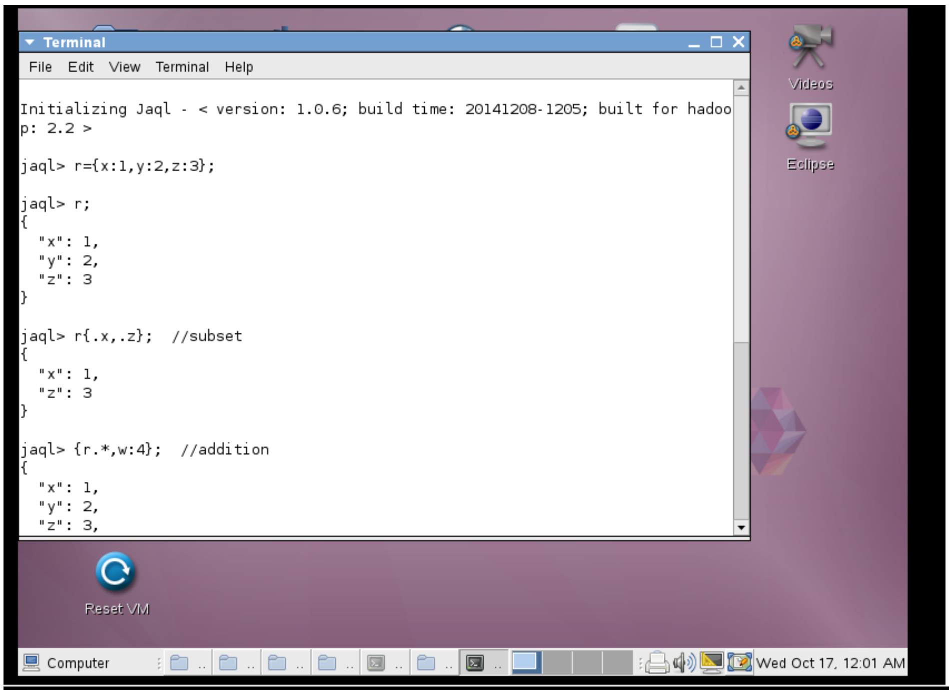
The -> operator
The -> operator “streams” an array through a function or core operator.

```
jaql>range(10) -> batch(5);
[
  [0, 1, 2, 3, 4],
  [5, 6, 7, 8, 9]
]
```

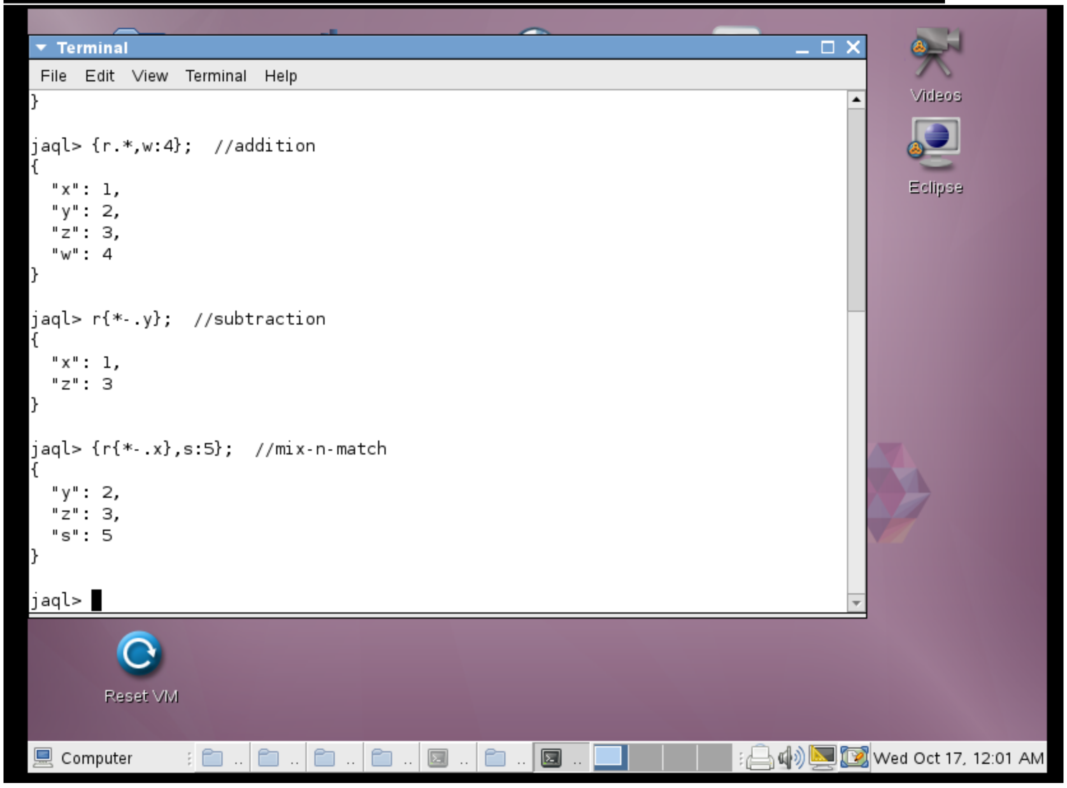
The array on the left is implicitly passed as the first argument to the function. This is identical to the above:

```
jaql> batch(range(10), 5);
```

Operator -> is just a “syntactic sugar” that can dramatically improve readability when multiple operations are involved on your data.



```
Initializing Jaql - < version: 1.0.6; build time: 20141208-1205; built for hadoo
p: 2.2 >

jaql> r={x:1,y:2,z:3};

jaql> r;
{
  "x": 1,
  "y": 2,
  "z": 3
}

jaql> r{.x,.z};  //subset
{
  "x": 1,
  "z": 3
}

jaql> {r.*,w:4};  //addition
{
  "x": 1,
  "y": 2,
  "z": 3,
```



```
}

jaql> {r.*,w:4};  //addition
{
  "x": 1,
  "y": 2,
  "z": 3,
  "w": 4
}

jaql> r{*-.y};  //subtraction
{
  "x": 1,
  "z": 3
}

jaql> {r{*-.x},s:5};  //mix-n-match
{
  "y": 2,
  "z": 3,
  "s": 5
}

jaql> 
```

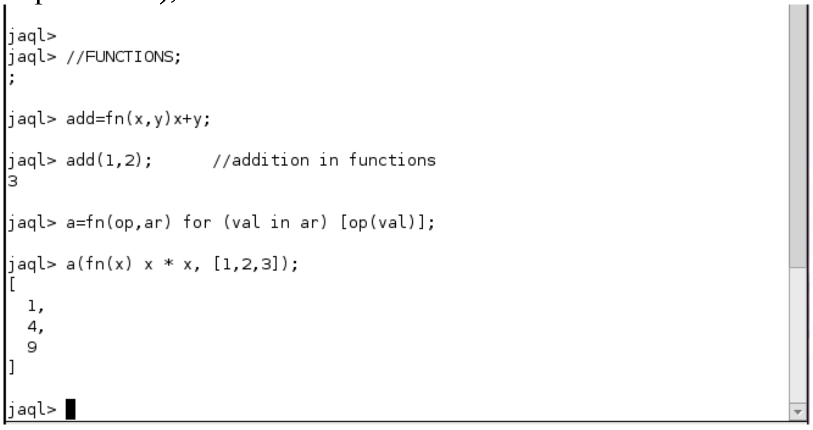
Input/Output with Jaql

Input/Output operations are performed through I/O adapters. Adapters are a description of how to access and process a data source. I/O adapter is then passed into I/O function (e.g. read() or write()). Following is an example of writing array into a (comma-)delimited csv file where we used delimited file I/O adapter:

jaql> [1, 2, 3, 4, 5] -> write(del("test.csv"));
jaql> read(del("test.csv"));
[1, 2, 3, 4, 5]

Local files or HDFS are then accessed by specyfing the full path as URI:
jaql> read(del("file:///home/user/test.csv")); // for local file system
jaql> read(del("hdfs://localhost:9000/user/test.csv")); // for hdfs file system

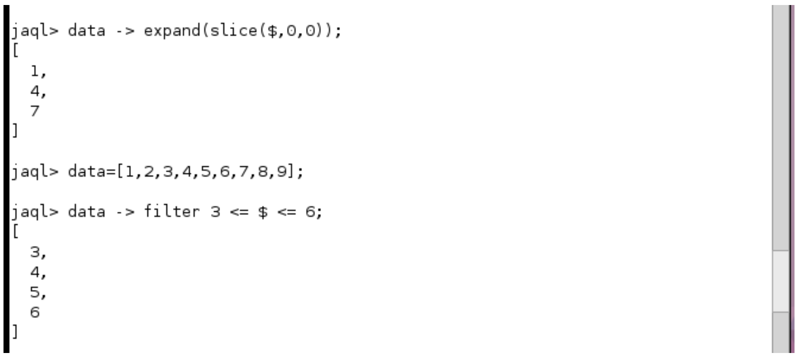
To read a JSON data from a URL, you can use jaglGet() function:
jaqlGet("file:///tmp/test.txt");



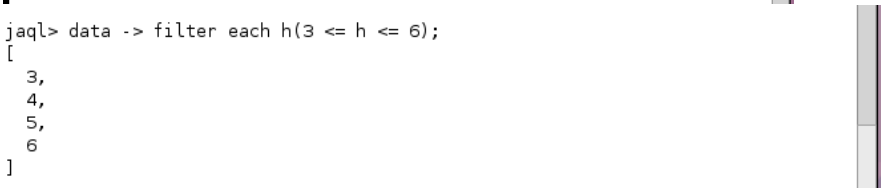
```
jaql>
jaql> //FUNCTIONS;
;

jaql> add=fn(x,y)x+y;

jaql> add(1,2);      //addition in functions
3

jaql> a=fn(op,ar) for (val in ar) [op(val)];

jaql> a(fn(x) x * x, [1,2,3]);
[
  1,
  4,
  9
]

jaql> 
```

Data manipulation (Core operators).
Core operators manipulate streams (arrays) of data, much in the way SQL clauses interact with data.

Filter
$ represents the current array value being evaluated.
jaql> read(del("file:///path/to/people.txt")) -> filter $.name == "Fred";
[
  { fname: "Fred", lname: "Johnson", age: 20 }
]
Other example could be:
jaql> data = [1, 2, 3, 4, 5, 6, 7, 8, 9];
jaql> data -> filter 3 <= $ <= 6;
[3, 4, 5, 6]
Alternatively, the each clause can be used to provide a name different than $:
jaql> data = [1, 2, 3, 4, 5, 6, 7, 8, 9];
jakq> data -> filter each num (3 <= num<= 6);
[3, 4, 5, 6]



```
jaql> data -> expand(slice($,0,0));
[
  1,
  4,
  7
]

jaql> data=[1,2,3,4,5,6,7,8,9];

jaql> data -> filter 3 <= $ <= 6;
[
  3,
  4,
  5,
  6
]
```



```
jaql> data -> filter each h(3 <= h <= 6);
[
  3,
  4,
  5,
  6
]
```

Transform

The transform operator allows you to manipulate the values in an array. An expression is applied to each element in the array:

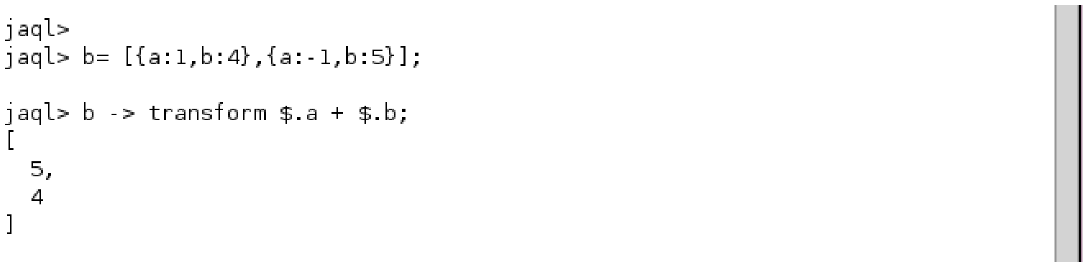
jaql> recs = [{a: 1, b: 4}, {a: 2, b: 5}, {a: -1, b: 4}];

jaql> recs -> transform $.a + $.b;

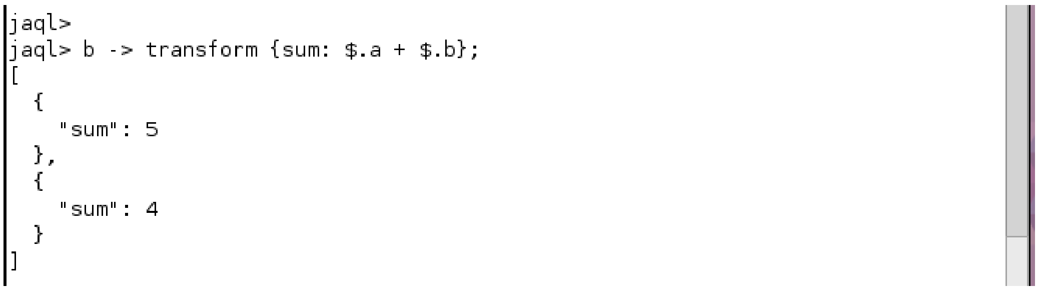
[5, 7, 4]

jaql> recs -> transform { sum: $.a + $.b };

[{ sum: 5 }, { sum: 7 }, { sum: 3 }]



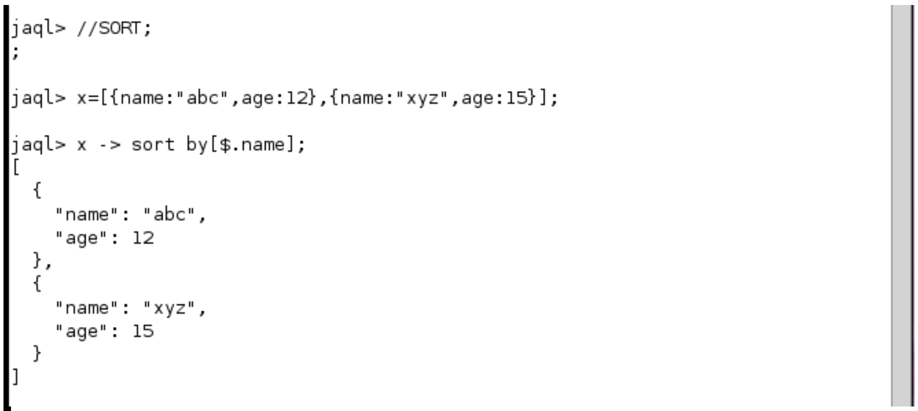
```
jaql>
jaql> b= [{a:1,b:4},{a:-1,b:5}];

jaql> b -> transform $.a + $.b;
[
  5,
  4
]
```



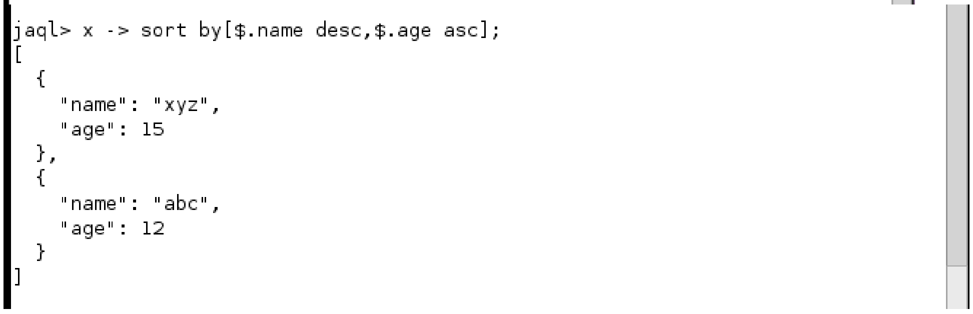
```
jaql>
jaql> b -> transform {sum: $.a + $.b};
[
  {
    "sum": 5
  },
  {
    "sum": 4
  }
]
```

Sort

jaql> read(del("file:///path/to/people.txt")) -> sort by [$.ageasc];



```
jaql> //SORT;
;

jaql> x=[{name:"abc",age:12},{name:"xyz",age:15}];

jaql> x -> sort by[$.name];
[
  {
    "name": "abc",
    "age": 12
  },
  {
    "name": "xyz",
    "age": 15
  }
]
```

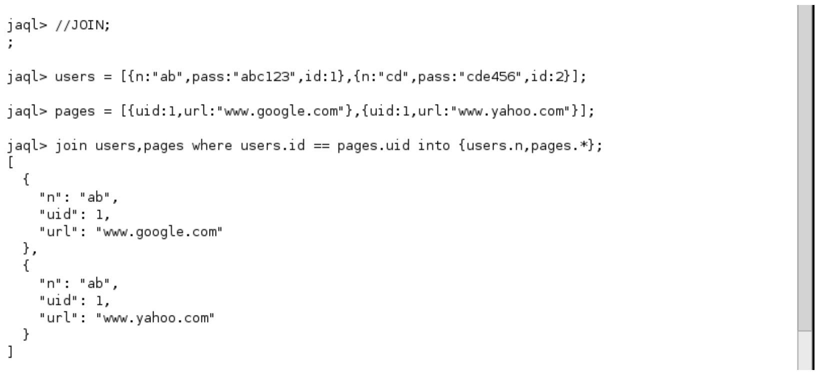


```
jaql> x -> sort by[$.name desc,$.age asc];
[
  {
    "name": "xyz",
    "age": 15
  },
  {
    "name": "abc",
    "age": 12
  }
]
```

Other data manipulation operators

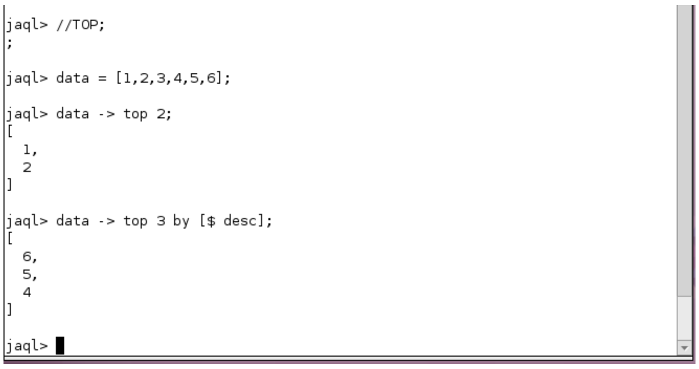
Other data manipulation operators are expand, group, join, top.

Join



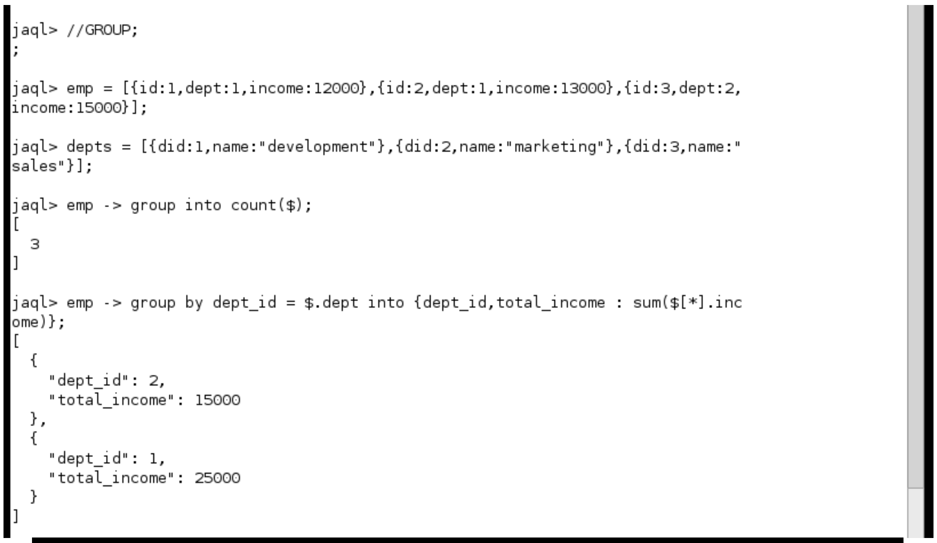
```
jaql> //JOIN;
;

jaql> users = [{n:"ab",pass:"abc123",id:1},{n:"cd",pass:"cde456",id:2}];

jaql> pages = [{uid:1,url:"www.google.com"},{uid:1,url:"www.yahoo.com"}];

jaql> join users,pages where users.id == pages.uid into {users.n,pages.*};
[
  {
    "n": "ab",
    "uid": 1,
    "url": "www.google.com"
  },
  {
    "n": "ab",
    "uid": 1,
    "url": "www.yahoo.com"
  }
]
```

Top

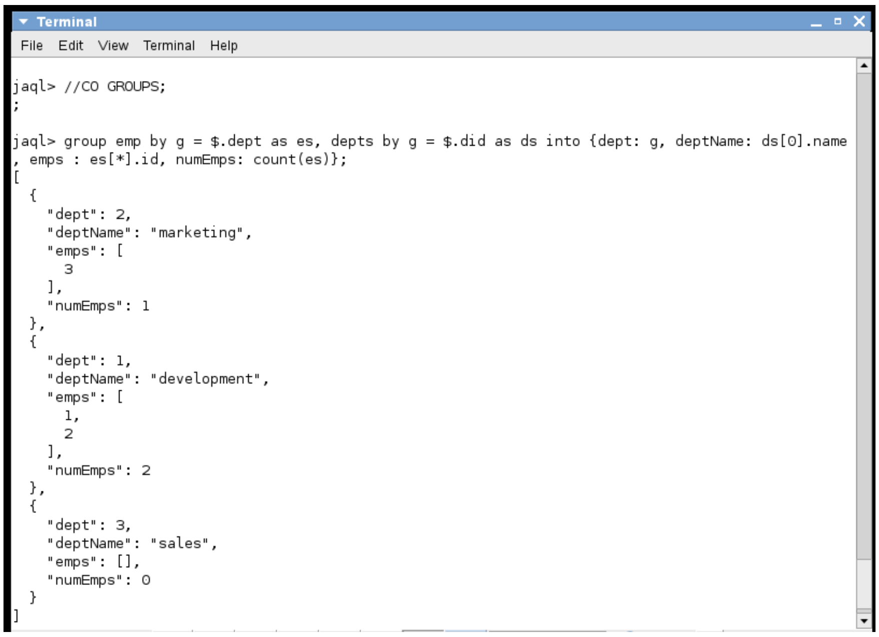


```
jaql> //TOP;
;

jaql> data = [1,2,3,4,5,6];

jaql> data -> top 2;
[
  1,
  2
]

jaql> data -> top 3 by [$ desc];
[
  6,
  5,
  4
]

jaql> █
```

Group



```
jaql> //GROUP;
;

jaql> emp = [{id:1,dept:1,income:12000},{id:2,dept:1,income:13000},{id:3,dept:2,
income:15000}];

jaql> depts = [{did:1,name:"development"},{did:2,name:"marketing"},{did:3,name:"
sales"}];

jaql> emp -> group into count($);
[
  3
]

jaql> emp -> group by dept_id = $.dept into {dept_id,total_income : sum($[*].inc
ome)};
[
  {
    "dept_id": 2,
    "total_income": 15000
  },
  {
    "dept_id": 1,
    "total_income": 25000
  }
]
```



```
jaql> //CO GROUPS;
;

jaql> group emp by g = $.dept as es, depts by g = $.did as ds into {dept: g, deptName: ds[0].name
, emps : es[*].id, numEmps: count(es)};
[
  {
    "dept": 2,
    "deptName": "marketing",
    "emps": [
      3
    ],
    "numEmps": 1
  },
  {
    "dept": 1,
    "deptName": "development",
    "emps": [
      1,
      2
    ],
    "numEmps": 2
  },
  {
    "dept": 3,
    "deptName": "sales",
    "emps": [],
    "numEmps": 0
  }
]
```