# Big Data Hadoop Training

Session 2 Assignment 2 Solution:

**Task 1:**

**Q) Create a file max-temp.txt in local FS. Put some 10-15 records of date and temperature example:**

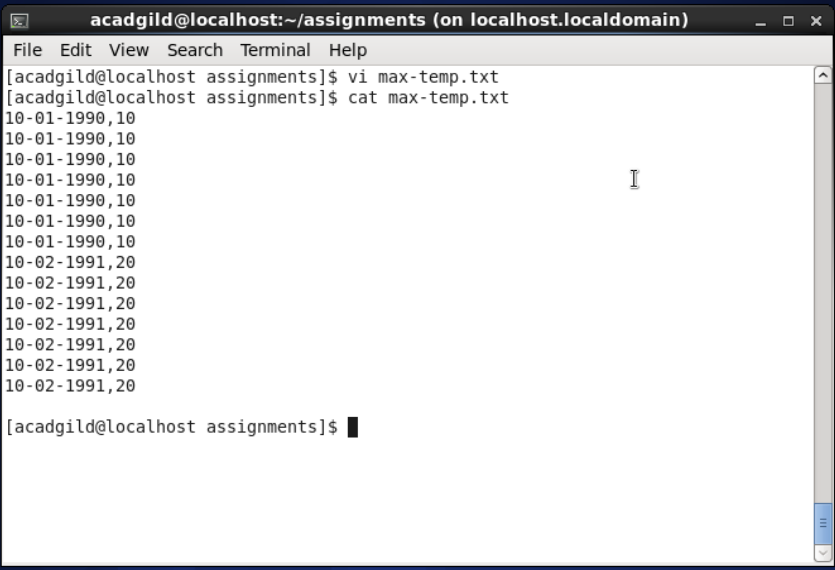
**dd-mm-yyyy,temperature**

**Example:**

**10-01-1990,10**

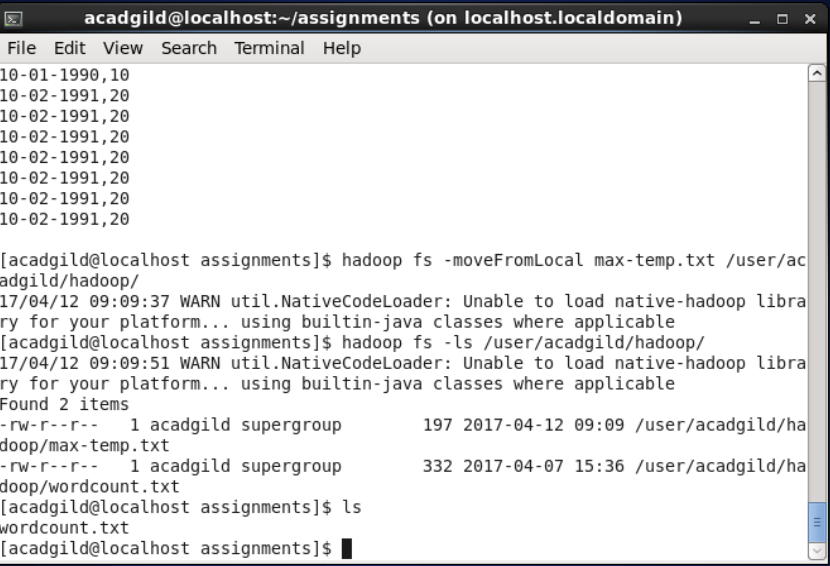
**10-02-1991,20**

A) A file max-temp.txt is created in local FS with some records as shown in the screenshot.



**Q) Move this file to HDFS at /user/acadgild/hadoop.**

A) Using cmd: “hadoop fs –moveFromLocal max-temp.txt /user/acadgild/hadoop/” we moved the file from local FS into specified HDFS location. We also observe that local FS now doesn’t contain the max-temp.txt since we moved the file into HDFS as shown in the screenshot.



**Task 2:**

**Q)** **Change the permission of the file /user/acadgild/hadoop/max-temp.txt, such that only the owner and the group members have full control over the file. Others do not have any control over it.**

A) Linux have three user classes as follows:

* User (u): The owner of file
* Group (g): Other user who are in group (to access files)
* Other (o): Everyone else

We can setup following mode on each files. In a Linux and UNIX set of permissions is called as mode:

* Read (r)
* Write (w)
* Execute (x)

Read mode permissions

* Read access on a file allows you to view file
* Read access on a directory allows you to view directory contents with ls command

Write mode permissions

* Write access on a file allows you to write to file
* Write access on a directory allows you to remove or add new files

Execute mode permissions

* Execute access on a file allows to run program or script
* Execute access on a directory allows you access file in the directory

Octal numbers and permissions

We use octal number to represent mode/permission:

* r: 4
* w: 2
* x: 1

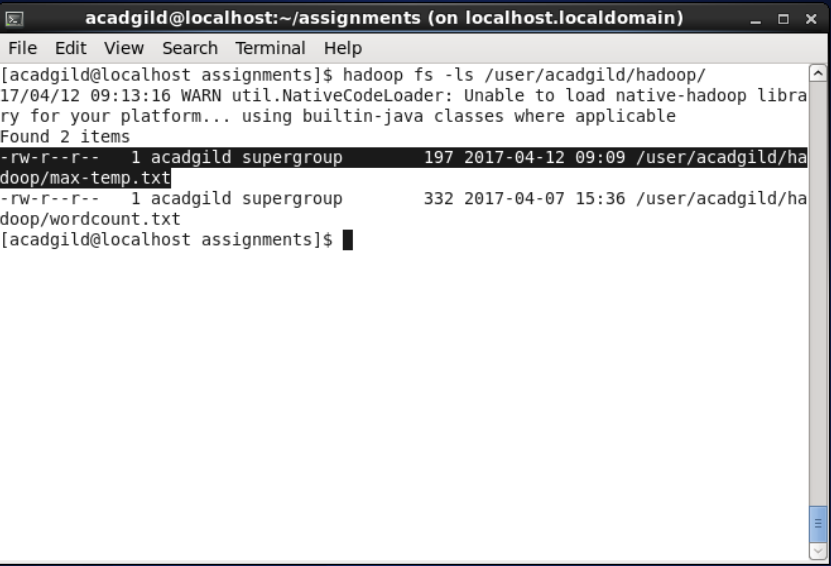
Here by default, permission for a file is showing as readable by anyone and writable by owner only in the screenshot.

**-rw-r--r-- indicates file permissions for user, group and others respectively**.

First 3 modes for User, here: **rw-** signifies only read,write permission for user.

Next 3 modes for Group, here: r-- signifies only read permission for group members.

Last 3 modes for others, here: r-- signifies only read permission for others.



**Our Requirement:**

In our case, for file owner to have read, write and execute (full) permission/control on file in octal is  
0+r+w+x = 0+4+2+1 = 7

Similarly, for group members to have read, write and execute (full) permission/control on file in octal is  
0+r+w+x = 0+4+2+1= 7

For others to have no control on file in octal is

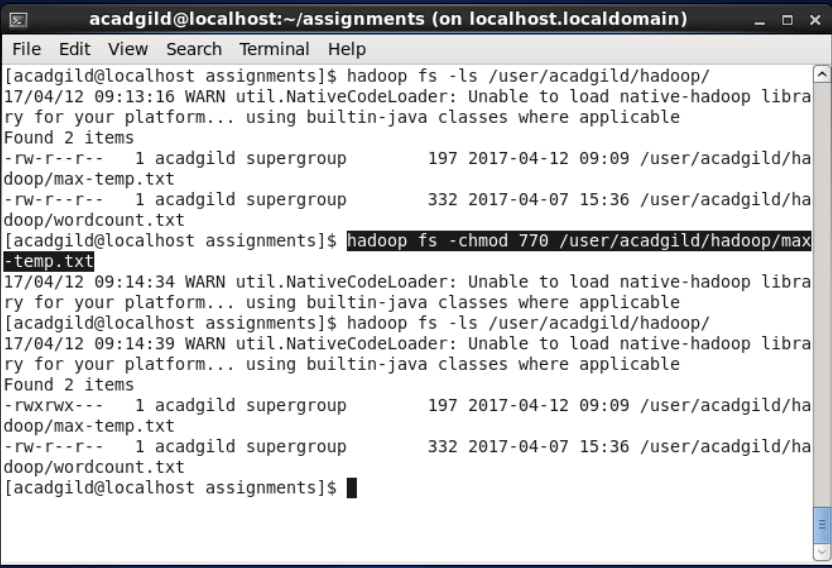
0+r+w+x = 0+0+0+0 = 0

**Effective permission is 770.**

To setup file permission we need to use chmod command: **chmod {mode} {file-name}**

In this case, since file is in HDFS, we use the cmd:

**“hadoop fs –chmod 770 /user/acadgild/hadoop/max-temp.txt”**



Now, we see the permissions getting changed such that only the owner and the group members have full control over the file. Others do not have any control over it.



**-rwxrwx--- indicates file permissions for user, group and others respectively that we modified using chmod command.**

First 3 modes for User, **rwx** signifies read,write,execute = full control for user.

Next 3 modes for Group, **rwx** signifies read,write,execute = full control for group members.

Last 3 modes for Others, **---** signifies no control for others.

Thus, we achieved our required task.