

CMS Data Analysis School Pre-Exercises - First Set

Contents:

[Introduction](#)

[Obtain a computer account:](#)

[Obtain a Grid Certificate and CMS VO registration](#)

[Obtain a github account:](#)

[Exercise 1 - Cut and Paste](#)

[Exercise 2 - Simple Edit Exercise](#)

[Exercise 3 - Setup a CMSSW release area like CMSSW_8_0_24_patch1](#)

[Exercise 4 - Find data in the DAS \(Data Aggregation Service\)](#)

[Exercise 5 - EDM \(Event Data Model framework\) standalone utilities -](#)

[edmFileUtil](#)

[edmDumpEventContent](#)

[edmProvDump](#)

[edmEventSize](#)

[Exercise 6 - Familiar with MiniAOD Format](#)

Introduction

Welcome to the first set of CMS Data Analysis School (CMSDAS) pre-exercises. The purpose of these exercises is to become familiar with the basic software tools required to perform physics analysis at the school. Please run and complete these exercises. Questions for each exercise are in **red** font. Post the answers in the online response form available from **the course web area** - [For CMSDAS@LPC 2017](#), [Fermilab](#). A large amount of additional information about these exercises is available in the twikis that we reference. Please remember that twikis evolve but aim to provide the best information at any time. If, at any time problems are encountered please e-mail **LPC contact** cmsdasatlpc@fnal.gov with a detailed description of your problem. The instructors will be happy to help you.

The CMSDAS exercises (pre-exercises as well as exercises during the school) are intended to be as generic as possible. However, CMSDAS is held at different CMS collaborating institutes - e.g. the LPC at Fermilab, DESY, etc.) Participants are expected to **request and obtain local (at the intended school location) computer accounts** well in advance of the school start date, to ensure they will be able to work right away. It is very important for participants to use the pre-exercises as a setup tool, so we recommend for everyone to **use the laptop they intend to bring with them at the school** (NO computer/laptop will be provided at the school), and to **connect to the local computing resources that will be used for the school**. In most cases laptops will need to be registered to be used in the local school network, so please make sure you take care of this aspect too.

For the upcoming CMSDAS@LPC2017, please find the information on getting LPC accounts, registering your laptop and connecting to the LPC cluster, at this [indico link](#).

There are several sets of pre-exercises. As outlined above, if you are going through the pre-exercises in preparation to attending a CMSDAS, we strongly recommend using the laptop you intend to bring to the school and logging into the cluster local to the school. Once you have run in the school local cluster you can also try them out in your local cluster/Tier-2/Tier-3 facilities with the necessary changes (accounts, login, local storage, local batch systems etc). It is assumed that there would be no difference in doing these exercises and the instructions should work the same way on any other cluster that has the CMS computing environment.

For the exercises that will be done during the school, instructions on those exercises will be found on twikis similarly to the pre-exercises.

Obtain a computer account:

To have a CERN account, please have a look at [Get Account at CERN](#). Obtaining a CERN account can be time-consuming. To expedite the process please ask the relevant institutional team leader to perform the necessary "signing" after the online form has been submitted and received for initial processing by the secretariat.

NOTE: If you need an account elsewhere, you need to contact your local cluster admins and follow their instructions.

For some sites, specific instructions exist. If they apply to you, please have a look at:

- [LPC/Fermilab](#) (For [CMSDAS@LPC 2017](#) please read the information at this link! Get your account but also see details on how to log in, mass storage etc.)
- [DESY/Germany](#)

Here are some additional (very old!) links in case you are using Windows:

- [Windows Kerberos at FNAL](#)
- [Hints for setting up Cygwin](#)
- In case one is using [PuTTY](#)

If you are attending CMSDAS at the LPC, you should do the pre-exercises on the **cmslpc-s16** cluster using the laptop you intend to bring to the LPC. You will also have to register that device on the FNAL network before coming, see link on the left bar of the [CMSDAS@LPC2017 indico agenda](#).

Obtain a Grid Certificate and CMS VO registration

- A Grid Certificate and CMS VO registration will be needed for the next set of exercises. The registration process can be time-consuming (actions by several people are required), so it is important to start it as soon as possible. There are two main requirements which can be simply summarized: A certificate ensures that you are who you claim to be. A registration in the VO recognizes your (identified by your certificate) as a member of CMS. Use the following link for this: [Get Your Grid Certificate and CMSVO](#). Both are needed to submit jobs on the Grid. Make sure you follow any additional instructions for US-CMS users.

Obtain a github account:

Since Summer 2013, most of the CMS software are hosted on [Github](#). Github is a Git repository web-based hosting service, while Git is a distributed revision control system. In

your future analysis work, version control of your analysis code will become a very important task and git will be very useful. A small git tutorial will wait for you in a later exercise set.

In order to checkout and develop CMS software, you will need a github account, which is free.

- In case you don't have one already, simply go to: <https://github.com/join> and follow the instructions to create a new account. Make sure you use a username people can recognize you easily or to specify your real name.
- In case you already have an account you can simply use "the Sign in" dialog and put your username and password. <https://github.com/login>
- **Make sure you register your ssh key in Github.**

NOTE: Legend of colors for this tutorial:

GRAY background for the commands to execute (cut&paste)

GREEN background for the output sample of the executed commands

BLUE background for the configuration files (cut&paste)

PINK background for the code (EDAnalyzer etc.) (cut&paste)

Exercise 1 - Cut and Paste

This exercise is designed to run only on **cmslpc-s16** as copies of the scripts are present there.

Login to the **cmslpc-s16** cluster. If you are preparing for CMSDAS@LPC2017 please know that the **cmslpc-s16** is the cluster you are supposed to use. By now you should have a FNAL account that you can use to get kerberos credential and follow the instructions on [how to log in to the LPC cluster](#).

As the exercises often require copying and pasting from instruction, we will make sure that you will have no problems. To verify if cut and paste to/from a terminal window works, first copy the script `runThisCommand.py` as follows.

To run on `cmslpc-s16` at Fermilab, try the following commands:

```
kinit YourUsername@FNAL.GOV
Password for YourUsername@FNAL.GOV:
```

Enter the password and then do:

```
ssh -Y YourUsername@cmslpc-s16.fnal.gov
cp -cmsdas/runThisCommand.py .
chmod +x runThisCommand.py
```

and then cut and paste the following and then hit return

```
./runThisCommand.py "asdf;klasdf;kakjsdf;akjf;aksdljf;a" "sldjfqewradsfafaw4efaeafawefzdxffasdfw4ffawefawe4fawasdffadsfef"
```

The response should be your username followed by alphanumeric string of characters unique to your username, for example for a user named gbenelli:

```
success: gbenelli toraryyv
```

QUESTION 1 - Post the alphanumeric string of characters unique to your username. For CMSDAS@LPC2017 please write your answers at [CMSDAS@LPC2017 espace](#).

If you executed the command without copy-pasting:

```
./runThisCommand.py
```

the command will return:

```
Error: You must provide the secret key
```

Alternately, copying incorrectly will return

```
Error: You didn't paste the correct input string
```

If you are not running on `cmslpc-s16` (for example locally on a laptop), will result in:

```
bash: ./runThisCommand.py: No such file or directory
```

OR:

```
Unknown user: gbenelli.
```

Exercise 2 - Simple Edit Exercise

This exercise is designed to run only on **cmslpc-s16**.

The purpose of this exercise is to ensure that the user can edit files. **This means that you need to be able to use one of the standard text editors (emacs, pico, nano, vi, vim, etc.) available on the cluster you are running (cmslpc-s16), open a file, edit it and save it!**

On the **cmslpc-s16** cluster:

```
cp ~cmsdas/editThisCommand.py .
```

Then open `editThisCommand.py` with your favorite editor (e.g. `emacs -nw editThisCommand.py`) and make sure that the 11th line has `#` (hash character) as the first character of the line. If not, explicitly change the following three lines:

```
# Please comment the line below out by adding a '#' to the front of
# the line.
raise RuntimeError, "You need to comment out this line with a #"
```

to:

```
# Please comment the line below out by adding a '#' to the front of
# the line.
#raise RuntimeError, "You need to comment out this line with a #"
```

Save the file (e.g. in emacs `CTRL+x CTRL+s` to save, `CTRL+x CTRL+c` to quit the editor) and execute the command:

```
./editThisCommand.py
```

If this is successful, the result will be:

```
success: gbenelli 0x6D0DB4E0
```

QUESTION 2 - Paste the line beginning with "success" into the form provided.

If the file has not been successfully edited, an error message will result such as:

```
Traceback (most recent call last):
  File "./editThisCommand.py", line 11, in ?
    raise RuntimeError, "You need to comment out this line with a #"
RuntimeError: You need to comment out this line with a #
```

Exercise 3 - Setup a CMSSW release area like CMSSW_8_0_24_patch1

CMSSW is the CMS SoftWare framework used in our collaboration to process and analyze data. In order to use it, you need to set up your environment and set up a local CMSSW release.

At Fermilab `cmslpc-s16` one can use the `nobackup` area linked from your home directory at `cmslpc-s16` (if do `ls -alh |grep nobackup` you will see something like `nobackup -> /uscms_data/d3/YOURUSERNAME`) for the exercises.

```
source /cvmfs/cms.cern.ch/cmsset_default.csh #or .sh for bash
```

Actually you should edit your `~/tcshrc` file (or `~/bashrc` if `bash` is your default shell), create it if you do not have one, to include the above command so that you do not have to execute each time you log into the cluster.

Then proceed with the creation of a working area (called `YOURWORKINGAREA` in the following):

```
cd ~/nobackup
mkdir YOURWORKINGAREA
cd YOURWORKINGAREA
### If you are using the default tcsh shell (or csh shell)
setenv SCRAM_ARCH slc6_amd64_gcc530
### Alternatively, If you are using Bash shell
export SCRAM_ARCH=slc6_amd64_gcc530
cmsrel CMSSW_8_0_24_patch1
cd CMSSW_8_0_24_patch1/src
cmsenv
git cms-init
```

This last command will take some time to execute and will produce some long output, be patient.

When you get the prompt again, run the following command:

```
echo $CMSSW_BASE
```

QUESTION 3 - Paste the result of executing the above command in the form

Note: The directory (on `cmslpc-s16`) `~/nobackup/YOURWORKINGAREA/CMSSW_8_0_24_patch1/src` is referred to as your `WORKING DIRECTORY`.

Every time you log out or exit a session you will need to setup your environment in your working directory again. To do so, once you have executed once the steps above (assuming you have added the `source /cvmfs/cms.cern.ch/cmsset_default.csh #or .sh` for bash in your `~/tcsh` or `~/bashrc` file), you simply:

```
cd ~/nobackup/YOURWORKINGAREA/CMSSW_8_0_24_patch1/src
cmsenv
```

And you are ready to go!

Exercise 4 - Find data in the DAS (Data Aggregation Service)

In this exercise we will locate the MC dataset `ReValZMM` and the collision dataset `/DoubleMuon/CMSSW_8_1_0_pre6-80X_dataRun2_reval_v12_ReVal_doubMu2016B-v1/MINIAOD` using the **Data Aggregation Service** (not to be confused with the **Data Analysis School** in which you are partaking!). Also remember DAS is an improved (faster) database access service previously known as DBS (Dataset Bookkeeping System).

Go to the url [DAS](#), **NOTE** that you will be asked for your Grid certificate which you should have loaded into your browser by now, (also note that there may be a security warning message, which you will need to ignore and still load the page) and type in the space provided:

```
dataset release=CMSSW_8_1_0 dataset=/ReValZMM*/*CMSSW_8_1_0*/MINIAOD*
```

This will search for datasets, processed with release `CMSSW_8_1_0`, which is named like `/ReValZMM*/*CMSSW_8_1_0*/MINIAOD*`. The syntax for searches is found [here](#), with many useful common search patterns under "CMS Queries".

For this query, several results should be displayed (you may be queried for security exceptions in the process). Select (click) on the dataset name `/ReValZMM_13/CMSSW_8_1_0-81X_mcRun2_asymptotic_v12-v1/MINIAODSIM` and after a few seconds another page will appear.

QUESTION 4.1 - What is the size of this dataset? Click on "Sites" to get a list of sites hosting this data. Is this data at FNAL? Is this data at DESY?

Click on the link "Files" to get a list of the root files in this dataset. One of the files it contains should look like this:

```
/store/reval/CMSSW_8_1_0/ReValZMM_13/MINIAODSIM/81X_mcRun2_asymptotic_v12-v1/10000/0EE14BA8-41BB-E611-AD2F-0CC47A4D760A.root
```

If you want to know the name of the dataset from the name of a file, one can go to [DAS](#) and type

```
dataset file=/store/reval/CMSSW_8_1_0/ReValZMM_13/MINIAODSIM/81X_mcRun2_asymptotic_v12-v1/10000/0EE14BA8-41BB-E611-AD2F-0CC47A4D760A.root
```

in the command line and hit "Enter".

Now we will locate a collisions dataset skim using the keyword search which is sometimes more convenient if you know the dataset you are looking for.

In [DAS](#), type

```
dataset=/DoubleMu*/*Run2016C*/MINIAOD*
```

and hit Enter. Answer the following question:

QUESTION 4.2 - What release was this dataset collected in? (If you see more than one release, just answer one)

Having set your CMSSW environment one can also search for the dataset `/DoubleMuon/Run2016C-PromptReco-v2/MINIAOD` by invoking the DAS command in your `WORKING DIRECTORY`. In order to do so, you need to get the DAS Command Line Interface (CLI) in your working directory. Following the directions [here](#) (under "DAS Command Line Tool"), you should be able to get the CLI with:

```
curl -k --cert ~/.globus/usercert.pem --key ~/.globus/userkey.pem https://cmsweb.cern.ch/das/cli > das_client.py
```

(You will need your Grid passphrase and the command is assuming your `usercert.pem` `userkey.pem` files are in the standard directories with those standard names, you will need to use your exact paths). Your CLI is now called `das_client.py`.

Alternatively (if the `curl` command above is not working for you) use your browser to open the ["CLI" macro](#) and download it, following the instructions to change the name to `das_client.py`. You will need to copy the file in your `cmslpc-s16` working directory (`scp` if you downloaded locally to your laptop).

Remember to:

```
chmod a+x das_client.py
```

The query we're interested in is: `/DoubleMuon/Run2016C-PromptReco-v2/MINIAOD`, see the commands below on how to execute it command-line.

NOTE: For `cmslpc-s16` at the LPC at Fermilab you will need to init your Grid proxy beforehand:

```
voms-proxy-init --voms cms
```

(You will be asked for your grid certificate passphrase). Then you can execute the query with:

```
./das_client.py --query="dataset=/DoubleMuon*/Run2016C-PromptReco-v2/MINIAOD" --format=plain
```

You will see something like

More information about accessing data in the [Data Aggregation Service](#) can be found in [WorkBookDataSamples](#)

Exercise 5 - EDM (Event Data Model framework) standalone utilities -

edmFileUtil, edmDumpEventContent, edmProvDump, edmEventSize

Make sure CMSSW has been set up as in [Exercise 3](#).

The overall collection of CMS software, referred to as [CMSSW](#), is built around a Framework, an Event Data Model (EDM), and Services needed by the simulation, calibration and alignment, and reconstruction modules that process event data so that physicists can perform analysis. The primary goal of the Framework and EDM is to facilitate the development and deployment of reconstruction and analysis software. The CMS Event Data Model (EDM) is centered around the concept of an Event. An Event is a C++ object container for all RAW and reconstructed data related to a particular collision. To understand what is in a data file and more, several EDM utilities are available. In this exercise, one will use three of these EDM utilities. They will be very useful at CMS DAS and after. More about these EDM utilities can be found at [WorkBookEdmUtilities](#). These together with the [Github web interface for CMSSW](#) and the [CMS LXR Cross Referencer](#) are very useful to understand and write CMS code.

edmFileUtil

First we will use the `edmFileUtil` to find the **physical file name** (PFN) where the file is actually stored at your site, given the **logical file name** (LFN) which is an alias that can be used in CMSSW at any site.

- Use `edmFileUtil` to find the **physical file name** (PFN) corresponding to the **logical file name** (LFN) from the MiniAOD data file located above.
 - To do this execute

```
edmFileUtil -d /store/relval/CMSSW_8_1_0/RelValZMM_13/MINIAODSIM/81X_mcRun2_asymptotic_v12-v1/10000/0EE14BA8-41BB-E611-AD2F-0CC47A4D760A.root
```

- Since you are working on `cmslpc-s16` this will return:

```
root://cmsxrootd-site.fnal.gov//store/relval/CMSSW_8_1_0/RelValZMM_13/MINIAODSIM/81X_mcRun2_asymptotic_v12-v1/10000/0EE14BA8-41BB-E611-AD2F-0CC47A4D760A.root
```

edmDumpEventContent

Next we will use `edmDumpEventContent` to dump a summary of the products that are contained within the file we're interested in, on `cmslpc-s16`:

- Use `edmDumpEventContent` to see what class names etc. to use in order to access the objects in the MiniAOD data file located above. If you want to look at a specific object (say, `slimmedMuons`) then execute

```
edmDumpEventContent --all --regex slimmedMuons root://cmsxrootd.fnal.gov//store/user/cmsdas/2017/pre_exercises/0EE14BA8-41BB-E611-AD2F-0CC47A4D760A.root
```

This will return:

Type	Module	Label	Process	Full Name
vector<pat::Muon>	"slimmedMuons"	"	"RECO"	patMuons_slimmedMuons_RECO

- The output of `edmDumpEventContent` has information divided into four variable width columns. The first column is the C++ class `type` of the data, the second is `module label`, the third is `product instance label` and the fourth is `process name`. More information is available at [Identifying Data in the Event](#).
 - **QUESTION 5.1 - How many modules produce products of type `vector` ? What are their names?**
 - NOTE: Instead of the above, try without the option `--regex slimmedMuons`. This will dump the entire event content - a file with many lines. For this reason we'll send the output to a file called `EdmDumpEventContent.txt` with a UNIX pipe command.
- To do this at the LPC, execute

```
edmDumpEventContent root://cmsxrootd.fnal.gov//store/user/cmsdas/2017/pre_exercises/0EE14BA8-41BB-E611-AD2F-0CC47A4D760A.root > EdmDumpEventContent.txt
```

edmProvDump

To aid in understanding the full history of an analysis, the framework accumulates provenance for all data stored in the standard ROOT output files. Using the command `edmProvDump` one can print out all the tracked parameters used to create the data file. For example, one can see which modules were run and the CMSSW version used to make the MiniAOD file. In executing the command below it is important to follow the instructions carefully, otherwise a large number of warning messages may appear. The `ROOT` warning messages can be

ignored.

- To do this at the LPC

```
edmProvDump root://cmsxrootd.fnal.gov//store/user/cmsdas/2017/pre_exercises/0EE14BA8-41BB-E611-AD2F-0CC47A4D760A.root > EdmProvDump.txt
```

- NOTE: `EdmProvDump.txt` is a very large file of the order of 40000-60000 lines. Open and look at this file and locate `Processing History` (about 20-40 lines from the top).
- **QUESTION 5.2 - Which version of CMSSW_?_?_? was used to produce the data?**

edmEventSize

Finally we will execute `edmEventSize` to determine the size of different branches in the data file. Further details may be found here: [SWGuideEdmEventSize](#). `edmEventSize` isn't actually a 'Core' helper function (anyone can slap 'edm' on the front of a program in CMSSW). You can use `edmFileUtil` to get a PFN from an LFN (as shown above) so you could combine the call

- Execute at `cmslpc-s16`:

```
edmEventSize -v `edmFileUtil -d /eos/uscms/store/user/cmsdas/2017/pre_exercises/0EE14BA8-41BB-E611-AD2F-0CC47A4D760A.root` > EdmEventSize.txt
```

- **QUESTION 5.3 What is the number of events if you execute the command at `cmslpc-s16`:**

```
edmEventSize -v `edmFileUtil -d /eos/uscms/store/user/cmsdas/2017/pre_exercises/0EE14BA8-41BB-E611-AD2F-0CC47A4D760A.root` > EdmEventSize.txt
```

• Open and look at file `EdmEventSize.txt` and locate the line containing the text `patJets_slimmedJetsPuppi__RECO`. There are two numbers following this text that measure the plain and the compressed size of this branch.

- **QUESTION 5.3 - What are these two numbers?**

Exercise 6 - Familiar with MiniAOD Format

Analyzing physics data at CMS is a very complicated task involving multiple steps, sharing of expertise, cross checks, and comparing different analysis. To maximize physics productivity, CMS has developed a new high-level data tier **MiniAOD** in Spring 2014 to serve the needs of the mainstream physics analyses while keeping a small event size (30-50 kb/event), with easy access to the algorithms developed by Physics Objects Groups (POGs) in the framework of the CMSSW offline software. The production of MiniAODs will be done centrally for common samples. Its goal is to centralize the production of PAT tuple which were used among the Physics Analysis Groups (PAGs) in Run 1. (Information about PAT can be found in [SWGuidePAT](#) and in a [CMS conference note](#).) MiniAOD samples will be used in the Run 2 analysis. Hence it is important to know about this tool. More information about MiniAOD can be found in [WorkBookMiniAOD](#).

The main contents of the MiniAOD are:

- High level physics objects (leptons, photons, jets, E_T^{miss}), with detailed information in order to allow e.g. retuning of identification criteria, saved using [PAT dataformats](#). Some preselection requirements are applied on the objects, and objects failing these requirements are either not stored or stored only with a more limited set of information. Some high level corrections are applied: L1+L2+L3(+residual) corrections to jets, type1 corrections to E_T^{miss} .
- The full list of particles reconstructed by the ParticleFlow, though only storing the most basic quantities for each object (4-vector, impact parameter, pdg id, some quality flags), and with reduced numerical precision; these are useful to recompute isolation, or to perform jet substructure studies. For charged particles with $p_T > 0.9$ GeV, more information about the associated track is saved, including the covariance matrix, so that they can be used for b-tagging purposes.
- MC Truth information: a subset of the [genParticles](#) enough to describe the hard scattering process, jet flavour information, and final state leptons and photons; GenJets with $p_T > 8$ GeV are also stored, and so are the other mc summary information (e.g event weight, LHE header, PDF, PU information). In addition, all the stable `genParticles` with mc status code 1 are also saved, to allow reclustering of GenJets with different algorithms and substructure studies.
- Trigger information: MiniAOD contains the trigger bits associated to all paths, and all the trigger objects that have contributed to firing at least one filter within the trigger. In addition, we store all objects reconstructed at L1 and the L1 global trigger summary, and the prescale values of all the triggers.

Please note that the files used in the following are from older releases, but they still illustrate the points they intended to. To avoid the fact that RelVal files (produced to validate new release in the rapid CMSSW development cycle) become unavailable on a short (month) timescale, a small set of files have been copied to the LPC EOS storage. They are available at `root://cmseos.fnal.gov//store/user/cmsdas/2017/pre_exercises/`.

The Z to dimoun MC file `root://cmseos.fnal.gov//store/user/cmsdas/2017/pre_exercises/CMSDataAnaSch_Data_706_MiniAOD.root` is made in `CMSSW_7_3_0_pre1` release and the datafile `root://cmseos.fnal.gov//store/user/cmsdas/2017/pre_exercises/CMSDataAnaSch_Data_706_MiniAOD.root` made from the collisions dataskim **/DoubleMu/CMSSW_7_0_6-GR_70_V2_AN1_ReIVal_zMu2011A-v1/MINIAOD**.

In your working directory, try to open the root file `root://cmseos.fnal.gov//store/user/cmsdas/2017/pre_exercises/CMSDataAnaSch_MiniAOD2MM730pre1.root`

```
root -l
```

On the `ROOT` prompt type the following:

```

gSystem->Load("libFWCoreFWLite.so");
AutoLibraryLoader::enable();
gSystem->Load("libDataFormatsFWLite.so");
gROOT->SetStyle ("Plain");
gStyle->SetOptStat(111111);

TFile *theFile = TFile::Open("root://cmseos.fnal.gov//store/user/cmsdas/2017/pre_exercises/CMSDataAnaSch_MiniAODZMM730pre1.root");

TBrowser b;

```

Note: TBrowser is a graphical browser. It runs on the computer, where you started ROOT. Its graphical interface needs to be forwarded to your computer. This can be very slow. You either need a lot of patience, a good connection or you can try to run ROOT locally, copying the root files that are to be inspected. Since everyone is running a different operating system on their local computer, we do not support the setup of ROOT on your local computer. However, instructions exist on the [official ROOT website](#).

To be able to use the member functions of a CMSSW data class from within ROOT, a 'dictionary' for that class needs to be available to ROOT. To obtain that dictionary, it is necessary to load the proper library into ROOT. The first three lines of the code above do exactly that. More information is at [WorkbookFWLiteExamples](#). Note that `gROOT->SetStyle ("Plain");` sets a plain white background for all the plots in ROOT.

NOTE: If the `rootlogon.C` is created in the home area, and the above four lines of code are in that file, the dictionary will be obtained, and all the plots will have a white background automatically upon logging in to ROOT.

Now a ROOT browser window opens and looks like this:

In this window click on `ROOT Files` on the left menu and now the window looks like this:

Click `CMSDataAnaSch_MiniAODZMM730pre1.root`, then `Events`, then scroll down and click `patMuons_slimmedMuons__PAT` and then `patMuons_cleanPatMuons__PAT.obj`. A window appears that looks like this:

Scroll a long way down the file (not too fast) and click on `pt()`. A PAT Muon Pt distribution will appear. These muons have been produced in the Z to mumu interactions as the name of the data sample implies.

QUESTION 6.1 - What is the mean value of the muon pt() for the MC data?

Note: To exit ROOT simply type `.q` in the command line.

Now open the data file `root://cmseos.fnal.gov//store/user/cmsdas/2017/pre_exercises/CMSDataAnaSch_Data_706_MiniAOD.root`. Similarly run the following command, and answer the following question:

```
root -l
```

On the ROOT prompt type the following:

```

gSystem->Load("libFWCoreFWLite.so");
AutoLibraryLoader::enable();
gSystem->Load("libDataFormatsFWLite.so");
gROOT->SetStyle ("Plain");
gStyle->SetOptStat(111111);
TBrowser b;

TFile *theFile = TFile::Open("root://cmseos.fnal.gov//store/user/cmsdas/2017/pre_exercises/CMSDataAnaSch_Data_706_MiniAOD.root");

```

QUESTION 6.2 - What is the mean value of the muon pt() for the collision data? * At `cmslpc-sl6` follow the same steps in `CMSSW_8_0_24_patch1` using the data file already used above: `root://cmseos.fnal.gov//store/user/cmsdas/2017/pre_exercises/0044DA75-708B-E611-8F8B-008CFA1974A4.root`

QUESTION 6.3 - What is the mean value of the muon pt() for the 2016 collision data?

Link to [SWGGuideCMSDataAnalysisSchoolPreExerciseSecondSet](#)





















Link to [SWGGuideCMSDataAnalysisSchoolPreExerciseThirdSet](#)

Link to [SWGGuideCMSDataAnalysisSchoolPreExerciseFourthSet](#)

Link to [SWGGuideCMSDataAnalysisSchoolPreExerciseFifthSet](#)

Link to [SWGGuideCMSDataAnalysisSchoolPreExerciseSixthSet](#)

- [runThisCommand.py.txt](#): python scripts for questions 1 and 2
- [editThisCommand.py.txt](#): python scripts for questions 1 and 2

I	Attachment	History	Action	Size	Date	Who	Comment
	DBS_snapshot.png	r1	manage	100.2 K	2009-10-16 - 15:09	SudhirMalik	
	DBS_snapshot_386.png	r1	manage	66.1 K	2010-11-08 - 01:25	SudhirMalik	
	DBS_snapshot_387.png	r1	manage	67.7 K	2010-12-13 - 04:26	SudhirMalik	
	GetInterestingEvents_cfg.py.txt	r1	manage	1.1 K	2011-11-04 - 20:23	SudhirMalik	
	MiniAOD_View1.png	r1	manage	15.7 K	2014-11-24 - 13:01	BenWu	
	MiniAOD_View2.png	r1	manage	16.9 K	2014-11-24 - 13:01	BenWu	
	MiniAOD_View3.png	r2 r1	manage	42.6 K	2014-11-24 - 13:01	BenWu	
	MiniAOD_View4.png	r2 r1	manage	39.5 K	2014-11-24 - 13:01	BenWu	
	copy_cfg.py.txt	r1	manage	3.3 K	2011-11-04 - 20:21	SudhirMalik	
	editThisCommand.py.txt	r1	manage	0.4 K	2015-08-21 - 03:08	SuyongChoi	python scripts for questions 1 and 2
	pickevents_cfg.py.txt	r1	manage	3.7 K	2011-11-04 - 20:24	SudhirMalik	
	putty.txt	r2 r1	manage	1.7 K	2009-10-19 - 16:13	SudhirMalik	
	root1.png	r1	manage	17.0 K	2009-10-16 - 14:01	SudhirMalik	
	root2.png	r2 r1	manage	16.8 K	2010-12-13 - 06:34	SudhirMalik	
	root6.png	r2 r1	manage	35.0 K	2010-12-13 - 06:35	SudhirMalik	
	root_428_1.png	r1	manage	24.1 K	2011-10-31 - 19:53	SudhirMalik	
	root_428_2.png	r1	manage	24.7 K	2011-10-31 - 19:52	SudhirMalik	
	root_428_6.png	r1	manage	28.4 K	2011-10-31 - 19:52	SudhirMalik	
	root_428_7.png	r1	manage	36.1 K	2011-10-31 - 19:52	SudhirMalik	
	runThisCommand.py.txt	r1	manage	0.8 K	2015-08-21 - 03:08	SuyongChoi	python scripts for questions 1 and 2
	simplePAT314.log	r1	manage	29.9 K	2009-10-16 - 15:15	SudhirMalik	
	simplePAT314_cfg.py.txt	r1	manage	0.7 K	2009-10-22 - 05:12	SudhirMalik	
	simplePAT386.log.txt	r1	manage	30.5 K	2010-11-08 - 03:21	SudhirMalik	
	simplePAT_314_cfg.py.txt	r1	manage	0.7 K	2009-10-22 - 05:18	SudhirMalik	
	simplePAT_MC_387.txt	r1	manage	5.5 K	2010-12-13 - 06:38	SudhirMalik	
	simplePAT_MC_525.txt	r1	manage	35.0 K	2012-07-21 - 21:44	NitishDhingra	
	simplePAT_MC_536.txt	r1	manage	34.3 K	2013-10-02 - 19:57	NitishDhingra	
	simplePAT_cfg_386.py.txt	r1	manage	0.5 K	2010-11-08 - 03:05	SudhirMalik	
	simple_PAT_MC_387_cfg.py.txt	r1	manage	1.5 K	2010-12-13 - 06:38	SudhirMalik	
	simple_PAT_MC_428_cfg.py.txt	r1	manage	0.7 K	2011-10-31 - 19:53	SudhirMalik	
	simple_PAT_MC_525_cfg.py.txt	r1	manage	0.8 K	2012-07-21 - 21:43	NitishDhingra	
	simple_PAT_MC_536_cfg.py.txt	r1	manage	0.9 K	2013-10-02 - 19:52	NitishDhingra	
	simple_PAT_data_387_cfg.py.txt	r1	manage	1.6 K	2010-12-13 - 06:37	SudhirMalik	
	simple_PAT_data_428_cfg.py.txt	r1	manage	1.2 K	2011-10-31 - 19:54	SudhirMalik	
	simple_PAT_data_525_cfg.py.txt	r1	manage	1.4 K	2012-07-21 - 21:44	NitishDhingra	
	simple_PAT_data_536_cfg.py.txt	r1	manage	1.4 K	2013-10-03 - 23:22	NitishDhingra	

Topic revision: r163 - 2017-01-08 - [BasilSchneider](#)

Copyright &© 2008-2017 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.
Ideas, requests, problems regarding TWiki? [Send feedback](#)

