Google Summer Of Code 2022 1st Evaluation Report:(CERN-HSF)

_By :_ Sayandeep Ghosh , Undergraduate Engineering Pre Final Year Student(Electronics & Instrumentation)
Jadavpur University, Kolkata, West bengal, India

Software Environment Used: Ubuntu(WSL-Windows Linux Subsystem)

Task 1 : Building ROOT from Sources :

```
$ git clone --branch latest-stable https://github.com/root-project/root.git root_src
$ mkdir root_build root_install && cd root_build
$ cmake -DCMAKE_INSTALL_PREFIX=../root_install ../root_src # && check cmake configurati
$ cmake --build . -- install -j4 # if you have 4 cores available for compilation
```

I used these lines of code for building ROOT from sources.
After building here is my result:

```
sayang@DESKTOP-IFS80HM:~$ source root_install/bin/thisroot.sh
sayang@DESKTOP-IFS80HM:~$ root
   ------------------------------------------------------------------
  | Welcome to ROOT 6.27/01                        https://root.cern |
  | (c) 1995-2021, The ROOT Team; conception: R. Brun, F. Rademakers |
  | Built for linuxx8664gcc on Mar 14 2022, 18:22:00                 |
  | From heads/master@v6-25-02-721-gb92cbb9ea9                       |
  | With c++ (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0                   |
  | Try '.help', '.demo', '.license', '.credits', '.quit'/'.q'       |
   ------------------------------------------------------------------

root [0]
```

I also familiarized myself with ROOT with the help of:
https://root.cern/primer/

I also checked whether RNTuple was build or not.

```
sayang@DESKTOP-IFS80HM:~$ root-config --features
cxx17 asimage builtin_afterimage builtin_clang builtin_cling builtin_llvm builtin_lz4 builtin_lzma builtin_nlohmannjson
builtin_openui5 builtin_pcre builtin_xxhash builtin_zstd dataframe exceptions gdml http mlp minuit2 pyroot roofit root7
rpath runtime_cxxmodules shared ssl tmva tmva-pymva spectrum x11 xml
sayang@DESKTOP-IFS80HM:~$ root-config --has-root7
yes
sayang@DESKTOP-IFS80HM:~$
```

Task 2: RNTuple Specific Task :

For the input file I took
root_src/tutorials/tree/temperature_Prague.dat

```
sayang@DESKTOP-IFS80HM:~$ cd root_src
sayang@DESKTOP-IFS80HM:~/root_src$ cd tutorials
sayang@DESKTOP-IFS80HM:~/root_src/tutorials$ cd tree
sayang@DESKTOP-IFS80HM:~/root_src/tutorials/tree$ ls
JetEvent.cxx   cernstaff.dat.pdf  example1.csv           hsimpleProxy.C        run_h1analysis.C        tree2.C
JetEvent.h     circular.C         h1analysis.C           hsimpleProxyDriver.C  simpleAnalysis.txt      tree2a.C
basic.C        clonesA_Event.C    h1analysis.h           hsimpleReader.C       spider.C                tree3.C
basic.dat      clonesA_Event.cxx  h1analysisProxy.C      htest.C               staff.C                 tree4.C
basic2.C       clonesA_Event.h    h1analysisProxy.h      hvector.C             tcl.C                   treefriend.C
bill.C         copytree.C         h1analysisProxyCut.C   jets.C                temperature.C           treegetval.C
cern.dat       copytree2.C        h1analysisTreeReader.C ntuple1.C             temperature_Prague.dat  tv3.C
cernbuild.C    copytree3.C        h1analysisTreeReader.h parallelcoord.C       tree.C                  tvdemo.C
cernstaff.C    drawsparse.C       h1chain.C              parallelcoordtrans.C  tree0.C
cernstaff.dat  example.dat        hsimple.root           printSizes.C          tree1.C
sayang@DESKTOP-IFS80HM:~/root_src/tutorials/tree$
```

I mainly used the above database(Temperature in Prague from 1775 to 2004) **but any database(.csv) will work provided that the user will have to mention the name(filename.csv after saving it in the tutorials/tree directory of the installation path) within the command line**. I created the following RNTuple ROOT File:
gsoc_eval_RNTuple.C(Source Code below which I tried to make it a little self explanatory):

```
R__LOAD_LIBRARY(ROOTNTuple)
#include <ROOT/RNTuple.hxx>
#include <ROOT/RNTupleModel.hxx>

#include <TCanvas.h>
#include <TH1I.h>
#include <TROOT.h>
#include <TString.h>

#include <cassert>
#include <cstdio>
#include <fstream>
#include <iostream>
#include <memory>
#include <string>
#include <sstream>
#include <utility>

using RNTupleModel = ROOT::Experimental::RNTupleModel;
using RNTupleReader = ROOT::Experimental::RNTupleReader;
using RNTupleWriter = ROOT::Experimental::RNTupleWriter;

constexpr char const* kNTupleFileName = "gsoc_eval_RNTuple.root";

std::vector<string> h;
std::vector<std::shared_ptr<int>> v1;
std::vector<std::shared_ptr<float>> v2;
std::pair<std::shared_ptr<int>,std::shared_ptr<float>> p;

void Ingest() {
    // The input file temperature_prague.dat is a copy of the temperature data base in Prague from 1775 to 2004
    //"/tree/temperature_Prague.csv"
    char filename[100];
    //I specifically mentioned the tuitorials directory since I found most of the datasets are stored there
    std::cout << "Enter the csv filename path(no more than 100 characters) present in the installation/tutorials directory of
ROOT):"<<std::endl;
    std::cin >> filename;
    ifstream fin(gROOT->GetTutorialDir() + "/tree/" + filename);
    assert(fin.is_open());
```

```
    //creating unique pointer to an empty data model
    auto model = RNTupleModel::Create();

    std::string record_header,word;
    getline(fin,record_header);
    std::istringstream iss(record_header);
    while(getline(iss,word,','))
    {
      h.push_back(word);
    }
    //std::cout<<h.size()<<std::endl;
      for(int i=0;i<h.size();i++)
      {
        if(h[i].find("int")!= string::npos){
        auto fld = model->MakeField<int>(h[i]);
        v1.push_back(fld);
        //std::cout<<"int"<<std::endl;
        }
        else if(h[i].find("float")!= string::npos) {
        auto fld = model->MakeField<float>(h[i]);
        v2.push_back(fld);
        }
      }

    // We hand-over the data model to a newly created ntuple of name "new_ntuple", stored in kNTupleFileName
    auto ntuple = RNTupleWriter::Recreate(std::move(model), "new_ntuple", kNTupleFileName);

    auto size = h.size();
    std::string record;
    while(std::getline(fin,record)) {
       std::istringstream iss(record);
       //iss>>*fld1>>*fld2>>*fld3>>*fld4;
       for(auto fld : v1)iss>>*fld;
       for(auto fld : v2)iss>>*fld;
       ntuple->Fill();
    }


}
```

```
void Analyze() {
   // Get a unique pointer to empty RNTuple models
   auto model = RNTupleModel::Create();

   std::string entry;
   std::cout<<"Enter the entry whose distribution you wish to see in the following way(Name:Type)!"<<std::endl;
   std::cin>>entry;

   //defining field(fldTemp from Ingest function where it was declared) that is needed for reading
   if(entry.find("int")!=string::npos){auto fld = model->MakeField<int>(entry);
     p.first = fld;
   }
   else if(entry.find("float")!=string::npos){auto fld = model->MakeField<float>(entry);
     p.second = fld;
   }

   // Quick overview of the ntuple and list of fields.
   auto ntuple = RNTupleReader::Open(std::move(model),"new_ntuple", kNTupleFileName);
   ntuple->PrintInfo();

   std::cout << "The first entry in JSON format:" << std::endl;
   ntuple->Show(0);

   auto c = new TCanvas("c", "", 200, 10, 700, 500);
   TH1I h("h", " Distribution for your entry", 100, -100, 100);
   h.SetFillColor(40);


   for (auto entryId : *ntuple) {
      ntuple->LoadEntry(entryId);
      if(entry.find("int")!=string::npos)h.Fill(*p.first);
      else if(entry.find("float")!=string::npos)h.Fill(*p.second);
   }

   h.DrawCopy();
}
```

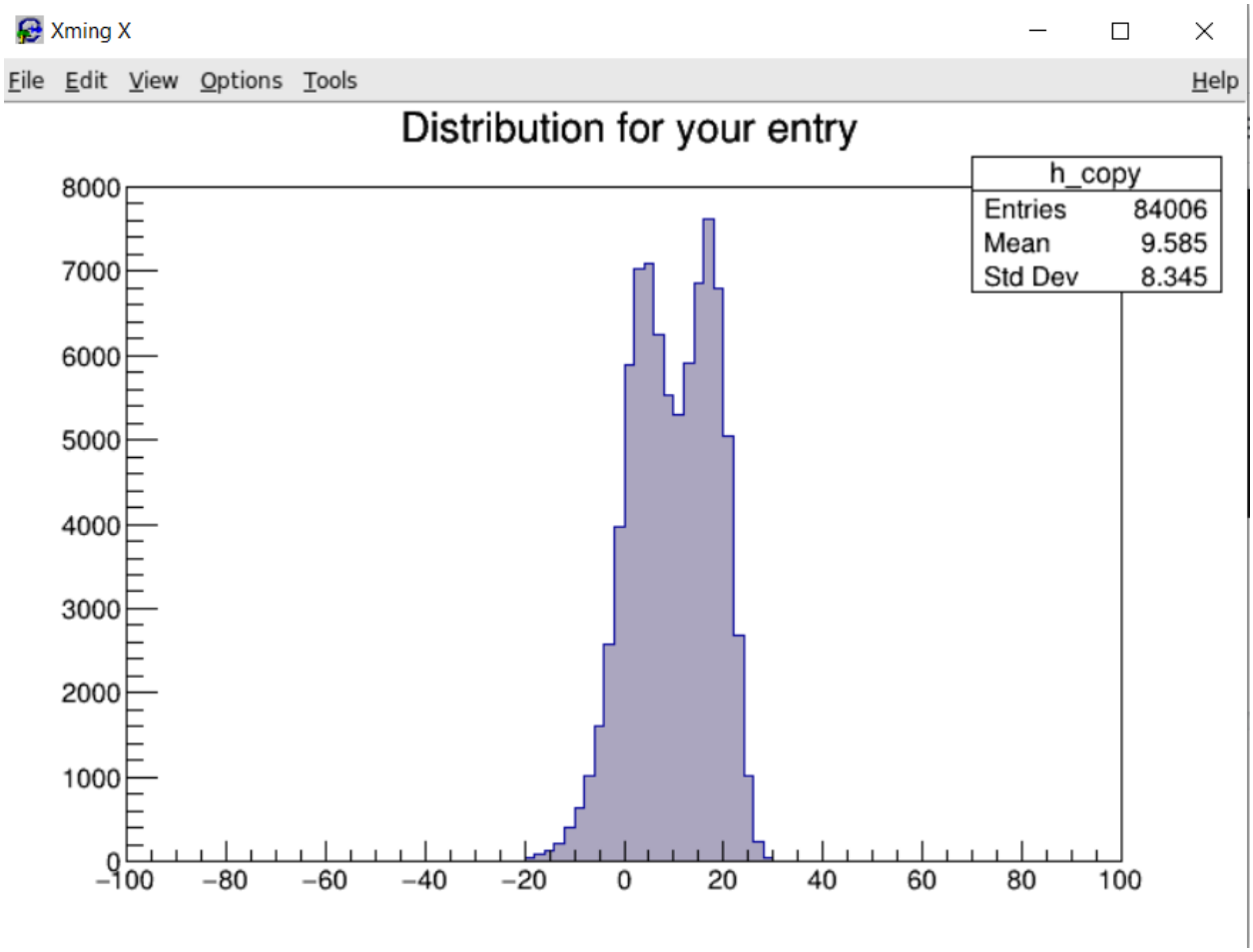
```
void gsoc_eval_RNTuple() {
   Ingest();
   Analyze();
}
```

After Running the file,**the user will have to enter the csv file path as well as the parameter(Name:type) whose distribution he/she wishes to see in a specific way.For simplicity as mentioned types should be int or float.**
It shows the following :

```
sayang@DESKTOP-IFS80HM:~/root_src/tutorials/v7/ntuple$ root gsoc_eval_RNTuple.C
   ------------------------------------------------------------
  | Welcome to ROOT 6.24/02                        https://root.cern |
  | (c) 1995-2021, The ROOT Team; conception: R. Brun, F. Rademakers |
  | Built for linuxx8664gcc on Jun 28 2021, 09:28:51                 |
  | From tags/v6-24-02@v6-24-02                                      |
  | With                                                             |
  | Try '.help', '.demo', '.license', '.credits', '.quit'/'.q'       |
   ------------------------------------------------------------

root [0]
Processing gsoc_eval_RNTuple.C...
Enter the csv filename (no more than 30 characters) present in the installation/tutorials directory of ROOT): temperatur
e_Prague.csv
```

```
Enter the entry whose distribution you wish to see in the following way(Name:Type)!
Temperature:float
********************************** NTUPLE ***********************************
* N-Tuple : new_ntuple                                                       *
* Entries : 84006                                                            *
******************************************************************************
* Field 1   : Year:int (std::int32_t)                                        *
* Field 2   : Month:int (std::int32_t)                                       *
* Field 3   : Day:int (std::int32_t)                                         *
* Field 4   : Temperature:float (float)                                      *
******************************************************************************
The first entry in JSON format:
{
  "Temperature:float": -7.4
}
root [1]
```


Xming X
File Edit View Options Tools Help
Distribution for your entry
h_copy
Entries 84006
Mean 9.585
Std Dev 8.345
8000
7000
6000
5000
4000
3000
2000
1000
0
−100 −80 −60 −40 −20 0 20 40 60 80 100