Cheat sheat for the ROOT tutorial

The following color scheme is used:
```
ROOT classes and methods.
Objects.
C++.
```

How to define a histogram in ROOT

```
TH1D* histogram = new TH1D(<name>, <title>, nBins, xMin, xMax);
```

TH1D means a 1 dimensional (1D) histogram where you can fill double (D). If memory is not an issue then it is always safer to use TH1D as this has the largest range.
<name> is the name of the object (the way that ROOT handles it internally – objects should have different names).
<title> is the title that will be drawn.
Nbins is the number of bins and xMin and xMax gives the range.

Example:
```
TH1D* hSin = new TH1D("hist", "ROOT func generated sin(x)
distribution; x; Counts", 100, 0, TMath::Pi());
```

will define a histogram with 100 bins from 0–π.
Note that for the title the semi colons select different regions:
"<title above histogram>; <x axis title>; <y axis title>"

How to create a fit function and fit a histogram.

Use the class TF1 (the 1 means that it is a 1 dimensional function).
```
TF1* function = new TF1(<name>, <expression>, xMin, xMax);
```
Example:
```
TF1* fitFunc = new TF1("fitFunc", "[0]*sin(x)", 0, TMath::Pi());
```
`[0]` is the free parameter that we want to fit. If the function has two parameters, the 2nd parameter is `[1]`.
```
fitFunc->SetParameter(0, 10);
```
Set the initial value (start guess for our fit) of parameter 0 to 10.
```
fitFunc->SetLineColor(kRed);
```
Set the color of the function to red.
```
hSin->Fit(fitFunc);
```
Fit histogram `hSin` with the function `fitFunc`. Note that the range of the function will be changed to match the range of the histogram (fit), unless one uses the option "R" when fitting.

Note that one can also do this interactively to check things, e.g.:
```
TF1* fitFunc = new TF1("fitFunc", "[0]*sin(x)", 0, Tmath::Pi());
fitFunc->SetParameter(0, 10);
fitFunc->Draw();
```
This will draw the function in a Canvas so one can see how it looks.