Physics and software tutorials:
Data analysis examples (HZ channels etc.)

https://hep-fcc.github.io/fcc-tutorials/master/fast-sim-and-analysis/fccanalyses/doc/starterkit/FccFastSimAnalysis/Readme.html

Learning objectives:

- Read the edm4hep data format and construct physics observable
- Define C++ helper functions
- Apply an event selection and fill histograms in a single iteration using the histmaker option
- Produce flat ntuples with observables of interest with FCCAnalyses
- Produce plots

Tutorial 1: recoil analysis

Reconstruct the Z recoil mass using di-muon events

- Select at least 2 muons with some kinematic selection criteria
- Build the Z resonance: loop over all muon candidates to form the m($\mu\mu$) closest to the Z mass
- Apply additional cuts (recoil window, Z momentum, …)

Samples:

- Signal: p8_ee_ZH_Zmumu_ecm240
- Backgrounds: p8_ee_ZZ_mumubb_ecm240, p8_ee_WW_ecm240 (as an example, others are missing!)

Steps to run the analysis:

1. Process the edm4hep samples using the Histmaker function: `fccanalysis run histmaker_recoil.py`
   a. The histmaker option produces directly histograms in ROOT files
   b. Using additional c++ functions to compute more sophisticated observables/selections
2. Produce plots: `fccanalysis plots plots_recoil.py`
3. Run statistical analysis (not covered today – will soon be added to the tutorial page)

Tutorial 2: flavor analysis

Analyse Z($\mu\mu$)H(bb) events

- Same base selection as recoil analysis
- Exploit jet clustering and flavor tagger to identify the b-jets

Samples: same as recoil analysis (no WW)

Steps to run the analysis:

1. Run over edm4hep samples and produce a mini tree: `fccanalysis run treemaker_flavor.py`
   a. For all events, compute jet clustering, apply tagger inference
   b. Using the same c++ functions for the recoil part
   c. Save the relevant branches (muons, jets)
2. Process the mini tree using the Histmaker function: `fccanalysis run histmaker_flavor.py`
   a. Produces final histograms
3. Produce plots: `fccanalysis plots plots_flavor.py`