# Export statement

When we are defining environmental variables we can use the export statement:

* Export X=1

This way the specified variable will be accessible in all the subprocesses (compiled programs) which we run after defining that variable.

Here are examples of subprocesses using environmental variable:

* Example 1. A bash subprocess:
  + export X=1
  + bash -c 'echo $X'
* Example 2. A Python subprocess:
  + export X=1
  + python3 -c 'import os; print(os.environ["X"])'

# Init system

That is a process which runs as the first one after kernel boots.

# Shell

Shell is a program which allows using commands in terminal and returns outputs. Those commands are from the shell’s language. Popular shells:

* sh (Bourne Shell)
* bash (Bourne Again Shell)

On Linux both those shells are saved in the /bin/sh and /bin/bash respectively.

# Processes

## Daemons

Daemons are background, long running processes.

## Commands

* ps aux – show all processes

# Editing files

## Commands

* Replace all the strings matching the pattern with a new string in a specific file:
  + sed -i ‘s/pattern/replacement/’ file\_path
* Add multiple line to a file:
  + Cat << EOF >> file\_path

Line1

Line2

EOF

If is a combination of multiple commands:

* + Cat
  + heredoc (<< EOF)
  + >> file\_path (redirect output of a command to a file)
* Display content of a file:
  + Cat file\_path
* Heredoc:
  + Command << EOF

Line1

Line2

EOF

It passes a multiline text as an argument to a command.

* redirect output of a command to a file (and create that file if it doesn’t exist):
  + Command >> file\_path – overwrite a file
  + Command > file\_path – append an output to a file