Process

A process is defined as a program in execution,

Un processus peut etre defini comme un thread avec un contexte en memoire programme en cours d'execution

Between the terminal emulator and the shell there is a middleman

Process intercommunication

Pipes

Variables

1. Using `export` : the variable can only be used in the current shell session and its subchilds.
2. Using `here document` : We can pass it using a here document that uses `EOF`

Example:

```
#! /bin/bash

var="value"
./child.sh << EOF
$var
EOF
```

```
#! /bin/bash

read VAR;
```

3. Using an intermediary file.
4. Using pipes.

The role of init

Although `init` is gradually being supplanted by `systemd` in many areas, it remains one of the fundamental concepts in a UNIX operating system.

The `init` process, assigned with `PID 1` , is initiated by the kernel upon boot as the foremost process and daemon. It assumes the role of the final active process just prior to system shutdown and serves as the direct or indirect progenitor of all other processes.

Its primary function lies in initializing daemons during system startup and iteratively managing zombie processes lacking parental supervision. `init` adopts these orphaned processes, designating itself as their parent, and concludes its existence with a `wait()` operation.