

Pan: High Level Description Language



3. Exercises

1. Introduction to Pan
2. Pan Syntax
- 3. Exercises**

Pan: *High Level Description Language*

3. Exercise 1

- 1.- Being given a template (`first_obj.tpl`) and an xml profile (`first_obj.xml-precomp`) find the differences in the contents and fix template to get the same xml profile.
- 2.- Look at the contents of the `second_obj.xml-precomp` file. Create template `second_obj`, and put it to a file called `second_obj.tpl`. Compare if you got the same contents as in `second_obj.xml-precomp`.
- 3.- Correct the compilation errors of the `third_obj.tpl` template. See if you got the same result as in the `third_obj.xml-precomp` file.

Pan: *High Level Description Language*

3. Exercise 2

Given the following schema:

```
+--network
| $ ip : string
| $ netmask : string
| $ gateway : string
+--system
| $ swap_size : long
+-- hardware
  +-ram
  | +-0
  | +-...
  +-cpu
  | $ name : string
  | $ speed : long
  | $ perf : double
```

And the list of templates:

```
ram.tpl
hardware.tpl
network.tpl
lxplus.tpl
lxplus001.tpl
```

Pan: *High Level Description Language*

3. Exercise 2

1.- Create a structure template `cpu` and add two `cpus` to the hardware template, in the way it corresponds to the schema. Use for example the values:

```
name           = "Pentium IV"  
speed          = 2400  
performance   = 2334.12
```

2.- Create a `system` template with temporary space information (512) and include it to the `lxplus` template.

3.- Compile the profile `lxplus001.tpl` and review the `lxplus001.xml` to check that it matches the proposed schema:

```
panc -x compact lxplus001.tpl
```

Note: Do not use the check scripts because we are solving a little bit different exercise!