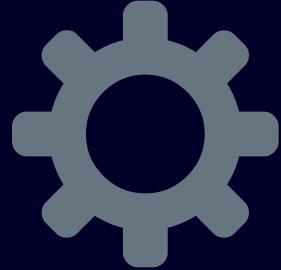


# Coding Style para PLC



Cómo nombrar a los elementos  
de un programa - Datos



⚠️ Post técnico, Tila incluida!

Empecemos





# DATOS



## Conceptos clave

### ● CONSTANTES

- Datos fijos/inmutables.

### ● VARIABLES

- Datos mutables en runtime.

### ● ESTRUCTURAS y TIPOS

- Grupo de variables que conforman una sola unidad.

### ● ENUMERACIONES

- Conjunto de constantes con un nombre.

Constantes





# Constantes



## Reglas

- Snake-case
- Mayúsculas

 NOMBRE\_DE\_LA\_CONSTANTE

 nombre\_de\_la\_constante, x\_const\_1

 ¿Por qué? Es una convención casi unánime en la mayoría de lenguajes de programación.

Variables





# Variables



## Reglas

- Camel-case
- Minúsculas

✓ `nombreDeLaVariable`

✗ `nombre_de_la_variable`, `x_nom_var`

? ¿Por qué? Snake-case hace variables largas

🚫 No uses la **notación húngara**, tuvo su razón de ser en su momento, ahora es muy fácil saber el tipado de la variable o constante.

ℹ Usa `_` cuando no puedes estructurar un grupo de variables:

- ✓ `motor1_isRunning`
- ✓ `motor1_command`
- ✓ `motor1_error`

Estructuras y Tipos





# Estructuras y Tipos



- Camel-case
- Minúsculas

✓ `nombreDeLaEstructura, nombreDelTipo`  
✗ `nombre_de_la_variable, udt_nom_var`

? ¿Por qué? Para mantener coherencia con las variables ya que una estructura o tipo puede ser una variable o parte de otra estructura.

```
STRUCTURE controllerA
    speed: real
    run: bool
END STRUCTURE
```

```
STRUCTURE motor
    isRunning: bool
    command: int
    error: bool
    controller: controllerA
END STRUCTURE
```

Enumeraciones





# Enumeraciones



## Reglas

- Camel-case
- Minúsculas

✓ `nombreDeLaEnumaracion`

✗ `nombre_de_la_enumeracion, enum_nom_v`

? ¿Por qué? Para mantener coherencia con las estructuras y tipos

! Las reglas para los valores de los elementos de una enumeración es la de las constantes.

```
ENUMERATION status
```

```
    STATUS_0_NIL : int := 0  
    STATUS_2_BUSY : int := 2  
    STATUS_8000_ERROR : int := 16#8000
```

```
END ENUMERATION
```

i No todos los PLC tienen enumeraciones.

Contenedores





# Contenedores DB/DI



Reglas

- Snake-case
- Mayúsculas

- ✓ NOMBRE\_DEL\_DB, NOMBRE\_DEL\_DI
- ✗ dbNombre\_del\_db, dbi\_nom\_inst\_blabla

? ¿Por qué? Un contenedor es un bloque inmutable, los datos almacenados son variables.

ℹ Los contenedores son principalmente para los PLC Siemens.

Para finalizar



# ¿Y TÚ?

¿Sabías que en los programas se debe aplicar un coding style?

¿Conocías alguna de estas reglas?

👉 ¡Debate en los comentarios!

#PLC #CodingStyle

 Primera parte