

# BUENAS PRÁCTICAS PARA LAS CONVENCIONES DE NOMENCLATURAS EN UIPATH

## Introducción

Una convención de nomenclatura es un conjunto de reglas para elegir la secuencia de caracteres que se utilizará para los identificadores que denotan variables, argumentos y otras entidades en la construcción de cualquier desarrollo.

## Desarrollo

Las razones para usar una convención de nomenclatura (en lugar de permitir que los programadores elijan cualquier secuencia de caracteres) incluyen las siguientes:

- Reducir el esfuerzo necesario para leer y comprender el código fuente.
- Para ayudar a formalizar las expectativas y promover la coherencia dentro de un equipo de desarrollo.
- Para proporcionar una mejor comprensión en caso de reutilización del código después de un largo intervalo de tiempo.
- Para mejorar la apariencia estética y profesional del producto del trabajo.

#### Estilos de nombrado

#### Camel case (camelCase)

Camel case es un estilo de escritura que se aplica a frases o palabras compuestas. El nombre se debe a que las mayúsculas a lo largo de una palabra en CamelCase se asemejan a las jorobas de un camello, La sintaxis "mi primera variable" en notación Snake Case sería "miPrimeraVariable".







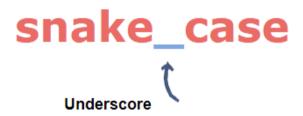
## Pascal case (PascalCase)

Pascal case es similar a camel case, excepto que la primera letra también comienza con una letra mayúscula. En este caso sería MiPrimeraVariable en lugar de miPrimeraVariable.



### **Snake case** (Snake\_case)

La notación Snake Case combina las palabras usando un guión bajo "\_" como nexo. La notación "mi primera variable" en notación Snake Case sería mi\_primera\_variable.



#### Nomenclatura de variables

En UiPath, una nomenclatura recomendada para el nombrado de variables es incluir la abreviación del tipo de dato de la variable (por ejemplo, para string sería str), junto con la descripción breve y representativa de la variable separadas por un guión (*snake case*), y la primera letra de cada palabra en mayúscula exceptuando la primera (*camel case*), como se muestra a continuación:





## Nomenclatura de argumentos

En UiPath, una nomenclatura recomendada para el nombrado de argumentos es incluir la dirección (in, out, inout), posteriormente la abreviación del tipo de dato de la variable (por ejemplo, para integer sería int), junto con la descripción breve y representativa de la variable separadas por un guión (*snake case*), y la primera letra de cada palabra en mayúscula (*Pascal case*), como se muestra a continuación:



Ejemplos de nomenclatura según el tipo de variable/argumento

Tipo de variable	Convención de nomenclatura	Ejemplo
String	str_VariableName	str_FirstName
Boolean	bln_VariableName	bln_DataExists
Integer	int_VariableName	int_RowNumber
String Array	strarr_VariableName	strarr_StringArray
Integer Array	intarr_VariableName	intarr_NumberCount
Data Table	dt_VariableName	dt_InvoiceTable
Data Row	dr_VariableName	dr_InvoiceRow
DataRowArray	drarr_ VariableName	drarr_InvoiceDataArray
List	lst_ VariableName	lst_Exception
Window	wnd_ VariableName	wnd_WindowVariable
Selector	selector_ VariableName	selector_DynamicSelector
Dictionary	dict_ VariableName	dict_DictionaryObject
Input Argument(String)	in_str_ VariableName	in_str_FirstName
Output Argument(String)	out_str_ VariableName	out_str_FirstName
Input Argument(Integer)	in_int_ VariableName	in_int_Numbers
OutputArgument(Integer)	out_int_ VariableName	in_int_Numbers
Input/Output Argument(String)	inout_str_ VariableName	inout_str_FirstName
Input/Output Argument(Integer)	inout_int_ VariableName	inout_int_FirstName
Queue	queue_ VariableName	queue_ItemCollection



## Comparación de la correcta e incorrecta generación de variables

