

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.

PascalCase

Capitalized

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.

snake_case

Underscore

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:

str_nombreDeLaVariable

abreviación del tipo de variable descripción breve y representativa de la variable

camelCase
snake_case

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:


dirección



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



| Name | Variable type | Scope | Default |
|-----------------|---------------|----------|-----------------------|
| variable1 | String | Sequence | Enter a VB expression |
| var | String | Sequence | Enter a VB expression |
| dato | String | Sequence | Enter a VB expression |
| texto | String | Sequence | Enter a VB expression |
| v1 | String | Sequence | Enter a VB expression |
| v2 | String | Sequence | Enter a VB expression |
| Create Variable | | | |

Variables Arguments Imports



| Name | Variable type | Scope | Default |
|-----------------------|---------------|----------|-----------------------|
| str_userName | String | Sequence | Enter a VB expression |
| str_address | String | Sequence | Enter a VB expression |
| bln_accessToPortalWeb | Boolean | Sequence | Enter a VB expression |
| int_rowCounter | Int32 | Sequence | Enter a VB expression |
| int_transactionNumber | Int32 | Sequence | Enter a VB expression |
| Create Variable | | | |

Variables Arguments Imports