

Artículo

[Mathew Lambert](#) · 17 dic, 2020 · Lectura de 3 min

Cómo crear código de "Solo escritura" (Irónicamente)

Uno de los principales beneficios de ObjectScript es la velocidad y eficiencia que ofrece a un desarrollador experimentado. Veamos un ejemplo de todos los beneficios que se pueden obtener con ObjectScript.

Imagina que tienes una clase que almacena nombres de sus usuarios. Llamaremos a la clase `**Data.User**`, y le daremos las mismas propiedades que tiene una cadena `Name_A` continuación, necesitaremos un método para crear un nuevo usuario o actualizar uno ya existente. Un enfoque inexperto y simplista sería muy similar al siguiente ejemplo:

```
ClassMethod UpdateOrCreateUser(id, name) As %String
{
    if ('$data(name)){
        quit "Name required"
    }
    if ($data(id)){
        set user = ##class(Data.User).%OpenId(id)
    }
    else{
        set user = ##class(Data.User).%New()
    }
    if ('user){
        quit "User not found"
    }
    set user.Name = name
    set status = user.%Save()
    if (status '= 1){
        quit "Error saving"
    }
    else{
        quit "User saved"
    }
}
```

El problema con este enfoque es que es demasiado detallado y extenso. Lleva mucho tiempo escribir el código, los nombres de las variables, los corchetes... y, cuando por fin has terminado, un fragmento de código trivial ocupará la mitad de la pantalla del monitor. Afortunadamente, ObjectScript permite que los desarrolladores simplifiquen el proceso y eliminen las partes que no son necesarias. Echemos un vistazo a cómo un desarrollador experimentado puede reescribir la parte complicada del código para obtener un resultado mucho más limpio:

```
ClassMethod CRUDUser(id, name) As %String
```

```
{
  q: '$d(name) "Name required"
  s x = $s($d(id):##class(Data.User).%OpenId(id),1:##class(Data.User).%New()) i 'x
  q "User not found"
  s x.Name = name q $s(x.%Save()=1:"User saved",1:"Error saving")
}
```

¡Y eso es todo! Mira todo lo que ObjectScript puede lograr con solo tres líneas de código. No necesitas un ordenador con dos monitores solo para leer el código. Y debido a que eliminamos todas las expresiones innecesarias, ni siquiera necesitas un ratón, ni una rueda de desplazamiento, ni la función copiar/pegar, ni cambiar el tamaño de las ventanas de la aplicación. Si te sientes cómodo desarrollando aplicaciones potentes utilizando solamente una pantalla monocromática de 80 columnas x 24 líneas con un teclado simple, un procesador de 8 Mhz y un disquete de 720 KB, entonces puedes aprovechar al máximo las ventajas de ObjectScript para aumentar tu productividad.

Ahora, puede que algunos principiantes tengan problemas para leer tu código optimizado. Y cuando llegue el momento de adaptar la aplicación a nuevos requisitos, a otros desarrolladores podría resultarles más sencillo reescribir el código por completo, en vez de tratar de descifrar tus breves líneas. ¡Incluso podrías tener dificultades para seguir tu propio código dentro de unos cuantos años! ¡Pero no hay nada que se pueda igualar a introducir tantas funcionalidades en un espacio tan pequeño!

Descargo de responsabilidad: Espero que todos los lectores se hayan dado cuenta de que este artículo tiene el modo ironía. Por respeto hacia tus colaboradores y hacia otros desarrolladores, no escribas un código que se parezca a una versión reducida, llena de abreviaturas y con una sola línea. A primera vista, esto puede ahorrarte algo de tiempo de teclado y varios clics con el ratón, pero se pierde mucho más tiempo cuando alguien distinto a ti debe mantener y actualizar dicho código. El código condensado es más difícil de leer y de seguir, y por lo tanto, es un buen candidato para escribirlo una sola vez y tirar el código posterior.

[#Consejos y trucos](#) [#ObjectScript](#) [#Principiante](#) [#Caché](#)

URL de
fuente: <https://es.community.intersystems.com/post/c%C3%B3mo-crear-c%C3%B3digo-de-solo-escritura-ir%C3%B3nicamente>