

Ejercicio A.8: Extender campos de tipo Option

En Dynamics NAV 2018 no era posible extender campos de tipo Option creados en el estándar de NAV o en una extensión que pretendemos extender.

En Dynamics 365 Business Central es posible extender estos tipos de campos a través de un nuevo tipo de objeto: **Enum**.

En este ejercicio aprenderemos a crear y a extender **Enums**.

Datos de conexión

Para este desarrollo, hay que utilizar la siguiente base de datos:

- Entorno de desarrollo:
 - Servidor: localhost\NAVDEMO
 - BBDD: BCCU0_AppDevelopment
 - Autenticación: Windows

Indicaciones para el ejercicio

Realiza las siguientes acciones:

1. Crea un nuevo enum y muéstralo en la ficha de clientes
2. Utiliza el enum en una función
3. Extiende un campo de tipo Option del estándar de NAV

Indicaciones paso a paso

Sigue los siguientes pasos en Visual Studio Code:

1. Crea un nuevo objeto de tipo enum, con las siguientes opciones:

```
enum 50120 "Customer Type"
{
    Extensible = true;
    value(0; " ")
    {
        Caption = ' ';
    }
    value(1; Company) { }
    value(2; Freelance) { }
}
```

2. Crea el campo Tipo de Cliente en la tabla de clientes

```
tableextension 50120 "Enum Customer" extends Customer
{
    fields
    {
        field(50120; "Customer Type"; enum "Customer Type")
        {
            CaptionML = ENU = 'Customer Type', ESP = 'Tipo cliente';
        }
    }
}
```

3. Muestra el campo en la ficha de clientes

```
pageextension 50120 "Enum Customer Card" extends "Customer Card"
{
    layout
    {
        addafter(Name)
        {
            field("Customer Type"; "Customer Type")
            {
                ApplicationArea = All;
            }
        }
    }
}
```

4. Utiliza el campo creado en su propio trigger OnValidate()

```
trigger OnValidate()
begin
    case "Customer Type" of
        "Customer Type"::Company:
            Message('Una empresa');
        "Customer Type"::Freelance:
            Message('Un freelance');
    end;
end;
```

- Los valores de un campo de tipo Enum se pueden acceder por código del mismo modo como accedemos a los valores de un campo de tipo Option.
- Campos y variables Enum solo se pueden asignar a campos y variables Enum del mismo tipo.
- En los campos de tipo Enum no hay conversión implícita con Integer (cómo si pasaba con los Options).
- Por ahora se soporta la conversión entre Option y Enum.
- Microsoft está trabajando en métodos de conversión, pero por ahora no existen.

5. Crea una nueva extensión que dependa de la extensión que contiene el enum para poderlo extender

```
"dependencies": [
  {
    "appId": "ca060ff2-ddca-4474-a1d4-3bfc80a27640",
    "name": "EjercicioA8.Enum",
    "publisher": "ClipDynamics",
    "version": "1.0.0.0"
  }
]
```

6. Extiende el Enum Customer Type

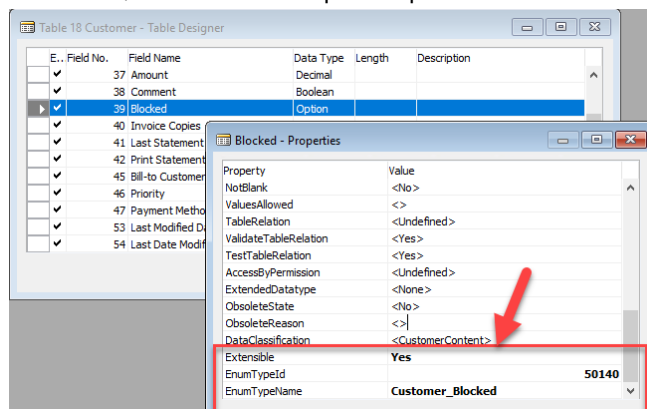
```
enumextension 50130 "Extended Customer Type" extends "Customer Type"
{
  value(50130; Individual) { }
  value(50131; Group) { }
}
```

7. Publica la extensión y comprueba que los nuevos valores son accesibles en la ficha del cliente
8. Para extender un campo de tipo **Option**, éste tiene que estar marcado como **Extensible** y tiene que estar vinculado a un **Enum**.

Para ello, en el campo de tipo Option, las siguientes propiedades tienen que tener información:

- Extensible
- EnumTypeId
- EnumTypeName

9. En C/SIDE, modifica el campo Bloqueado de la tabla de clientes para hacerlo extensible



10. Regenera los símbolos en C/SIDE y descárgalos de nuevo en Visual Studio Code

11. Extiende el Enum Customer_Blocked creado en pasos anteriores

```
enumextension 50101 "Customer Blocked" extends Customer_Blocked
{
    value(50101; Sales){}
}
```

12. Publica la extensión y comprueba que el nuevo valor es visible en el campo Bloqueado de la ficha del cliente

Microsoft no ha habilitado todos los options como enums extendibles por código como éste:

Codeunit 1314 – "Purch. Doc. From Sales Doc."

```
case SalesLine.Type of
    SalesLine.Type::" ":
        PurchaseLine.Type := PurchaseLine.Type::" ";
    SalesLine.Type::Item:
        PurchaseLine.Type := PurchaseLine.Type::Item;
    else
        Error(TypeNotSupportedErr, Format(SalesLine.Type));
end;
```

Cada vez que habilitan que un Option sea extendible, tienen que revisar el código y añadir eventos y formas de que a parte de extender las opciones que le aparecen al usuario, también podamos añadir código para gestionar la nueva opción.

Aunque no se trata de un Option, un ejemplo de opciones extendibles lo podemos encontrar en la Codeunit 5807 Item Charge Assgnt. (Sales), en las funciones **SuggestAssignment** y **AssignItemCharges**