

# Curso de Desarrollo

**Dynamics 365 Business Central**

**- La sesión empezará en breve -**

*ClipPlatform*

# Queries

Las Queries nos permiten obtener un conjunto de datos de la Base de Datos.

# Queries

Las Queries permiten:

- Seleccionar subconjuntos de campos de múltiples tablas
- Hacer JOINS con distintos criterios
- Filtrar tablas con distintos criterios
- Agrupar o agregar datos
- Ordenar
- Limitar el número de líneas

# Queries

Las Queries se pueden utilizar:

- Como origen de datos para gráficos
- Para guardar su resultado como XML o CSV
- Para acceder a su resultado en código AL
- Para publicarlas como WebServices OData para ser consumidas por otras aplicaciones

# Queries

- Crea una nueva Query 50100 Simple Item Query
  - Utiliza la tabla Item como DataItem
  - Añade los campos No., Description, Base Unit of Measure y Unit Cost

# Queries

```
query 50100 "Simple Item Query"
{
  QueryType = Normal;

  elements
  {
    0 references
    dataitem(Item; Item)
    {
      0 references
      column(No; "No.") { }
      0 references
      column(Description; Description) { }
      0 references
      column(Base_Unit_of_Measure; "Base Unit of Measure") { }
      0 references
      column(Unit_Cost; "Unit Cost") { }
    }
  }
}
```

# Queries

- Vamos a ampliar la Query para hacer JOINS con otras tablas.
- Antes, debemos crear una serie de datos

# Queries

- Selecciona el proveedor 30000 en los productos 70001, 70002, 70003 y 70010



# Queries

- Añade a la Query un DataItem de Vendor
  - Añade los campos Name y City
  - Cambia los nombres de las columnas por Vendor\_Name y Vendor\_City
  - Enlaza el DataItem de Vendor con el de Item

# Queries

```
dataitem(Vendor; Vendor)
{
  DataItemLink = "No." = Item."Vendor Item No.";
  0 references
  column(VendorName; Name) { }
  0 references
  column(VendorCity; City) { }
}
```

# Queries

- Configura el Join en Vendor para mostrar solo líneas de Item que tengan un registro en Vendor
  - Selecciona el valor InnerJoin en la propiedad DataItemLinkType

# Queries

- Hay varias formas de establecer un filtro en una Query
  - Utilizando la propiedad DataItemFilter de los DataItems
  - Utilizando la propiedad ColumnFilter en una columna
  - Creando líneas de tipo Filtro
  - Escribiendo código AL en el trigger OnBeforeOpen

# Queries

- Filtra la Query para mostrar sólo productos que utilizan Compra como Sistema de Reposición
  - Utiliza la propiedad DataItemTableFilter

```
dataitem(Item; Item)  
{  
    DataItemTableFilter = "Replenishment System" = const(Purchase);  
}
```

# Queries

- Filtra la Query para mostrar sólo productos que tienen un Coste Unitario definido
  - Utiliza la propiedad ColumnFilter

```
column(Unit_Cost; "Unit Cost")  
{  
    ColumnFilter = Unit_Cost = filter(<> 0);  
}
```

# Queries

- Permite filtrar la Query por Vendor No. y por Vendor Posting Group
  - Crea líneas de tipo Filter

```
dataitem(Item; Item)
{
    DataItemTableFilter = "Replenishment System" = const(Purchase);
    0 references
    filter(Vendor_No_; "Vendor No.") { }
```

# Queries

- Acceso a las Queries desde AL

```
WITH SimpleItemQuery DO BEGIN
  IF OPEN THEN BEGIN
    WHILE READ DO BEGIN
      // Do some logic
      VendorName := SimpleItemQuery.VendorName;
    END;
  CLOSE;
END;
END;
```



# Queries

- Filtrado de las Queries desde AL
  - Solo se puede filtrar por campos utilizados en la Query, o por campos no mostrados en la Query pero definidos como filtro

```
SimpleItemQuery.SETRANGE(Vendor_Posting_Group,'NAC');  
IF SimpleItemQuery.OPEN THEN BEGIN  
    WHILE SimpleItemQuery.READ DO BEGIN  
        // Do some logic  
        VendorName := SimpleItemQuery.VendorName;  
    END;  
CLOSE;  
END;
```

# Queries

- Mapping de Queries a Transact-SQL

| Transact-SQL | Query Feature  |
|--------------|--|
| SELECT       | Row of Type Column in the <b>Query Designer</b> window   |
| FROM         | Row of Type DataItem in the <b>Query Designer</b> window   |
| JOIN type    | DataItemLinkType and SQLJoinType query properties  |
| ON           | DataItemLink data item property  |
| WHERE        | DataItemTableFilter data item property<br>ColumnFilter property of columns and filters<br>Row of Type Filter |

# Queries

- Mapping de Queries a Transact-SQL

| Transact-SQL | Query Feature   |
|--------------|---|
| HAVING       | ColumnFilter property of columns and filters, when aggregation is used          |
| GROUP BY     | Automatically switched on for each row of Type Column, when aggregation is used |
| ORDER BY     | OrderBy query property  |
| TOP          | TopNumberOfRows query property  |

# Práctica

- Crea una codeunit que ejecute la Query 50100 Simple Item Query y recorra todos los registros resultado

# Práctica

- Crea un report que ejecute la Query 50100 Simple Item Query y imprima una lista con los resultados de la misma