Creating a new window

To create a new window we need to decide if we want to base the window on the already existing table or we want to create a new table.

Before creating a new table or window, you need to study the content of data model of iDempiere. Very good source of it is https://globalqss.com/idempiere/1.0c/schemaspy/Printing/index.html and also the page of idempiere wiki: http://wiki.idempiere.org/en/Database_structure

Create a new table

The table can be created in two ways:

- Directly in the database using any admin db tool (like PgAdmin)
- In the iDempiere

Each table **must contain** following columns:

- [TABLE NAME]_id numeric(10,0) PRIMARY KEY,
- [TABLE NAME]__uu character varying(36) DEFAULT NULL::character varying UNIQUE,
- ad_client_id numeric(10,0) NOT NULL,
- ad_org_id numeric(10,0) NOT NULL,
- isactive character(1) NOT NULL DEFAULT 'Y'::bpchar,
- created timestamp without time zone NOT NULL DEFAULT now(),
- createdby numeric(10,0) NOT NULL,
- updated timestamp without time zone NOT NULL DEFAULT now(),
- updatedby numeric(10,0) NOT NULL,

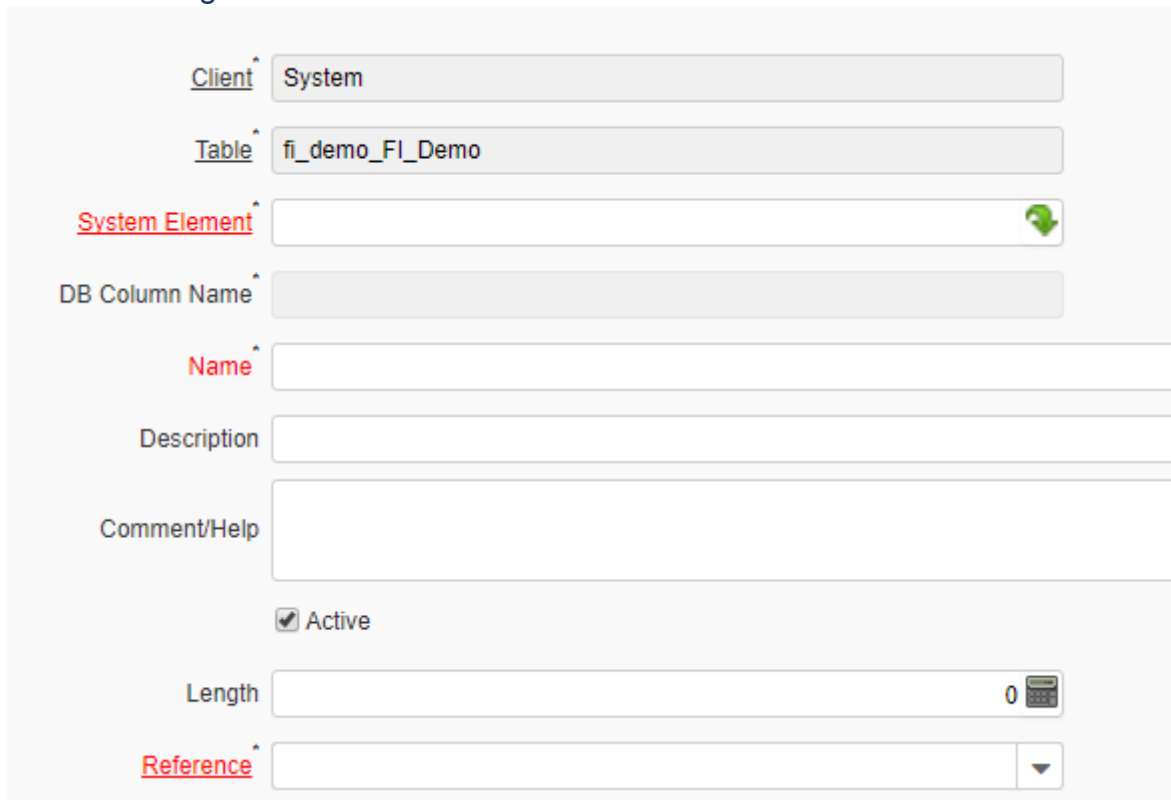Here is the script you can copy to begin your table (PostgeSQL):

CREATE TABLE [TABLE_NAME](

-- Mandatory columns

  [TABLE_NAME]_id numeric(10,0) PRIMARY KEY,
  [TABLE_NAME]_uu character varying(36) DEFAULT NULL::character varying UNIQUE,
  ad_client_id numeric(10,0) NOT NULL, ad_org_id numeric(10,0) NOT NULL, isactive character(1) NOT NULL DEFAULT 'Y'::bpchar, created timestamp without time zone NOT NULL DEFAULT now(), createdby numeric(10,0) NOT NULL, updated timestamp without time zone NOT NULL DEFAULT now(), updatedby numeric(10,0) NOT NULL,

-- User-defined columns

);

You can also define the table in the iDempiere:

1. Log as System Administrator
2. In the Menu select Application Dictionary -> Table and Column
3. Create New table and fill:
   a. DB Table Name = name the table will have in the database
   b. Name = Name for iDempiere
   c. Data Access Level = (All for every cases, Client/Organization for running as normal client)

4. Go to Column bookmark and add mandatory fields manually

Manual adding of columns

The most important is System Element. Everything in iDempiere is built on using centralized list of elements. The already defined elements are:

- ad_client_id
- ad_org_id
- isactive
- created
- createdby
- updated
- updatedby

To use already defined element, just write its name into System Element field. Everything else is done automatically. The last two fileds needs to be defined as new system elements:

1. [TABLE NAME]_id is the primary key of our new table.

a. Right-click to System Element and select New
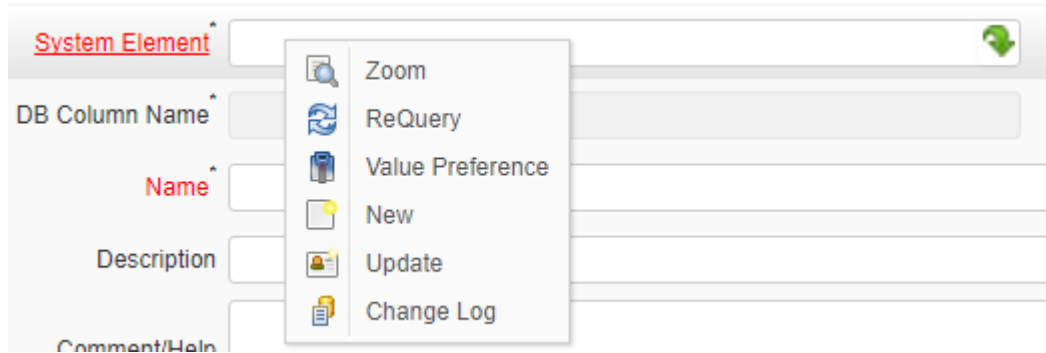
b. Fill the opened window according following picture ([TABLE NAME]=fi_demo:
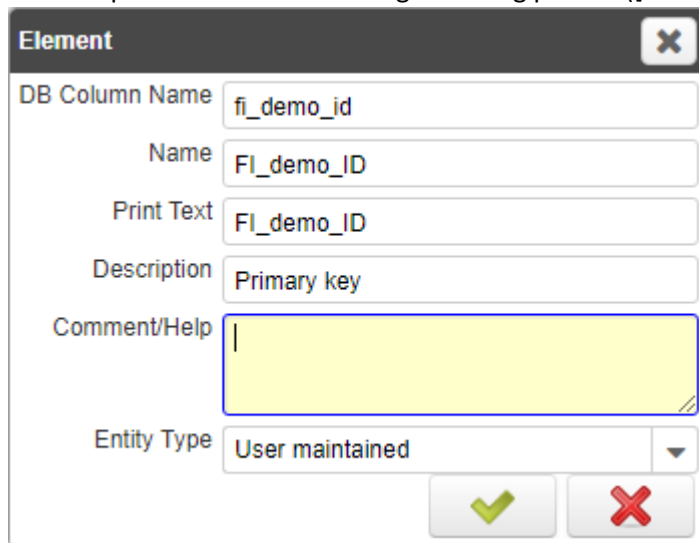
c. Set **Length** to 22
d. In Reference Field select **ID**
e. Check **Key column**
f. Uncheck **Updatable**
g. Uncheck **Allow Copy**

2. [TABLE NAME]_UU needs to be set in the following way:

a. Right-click to System Element and select New
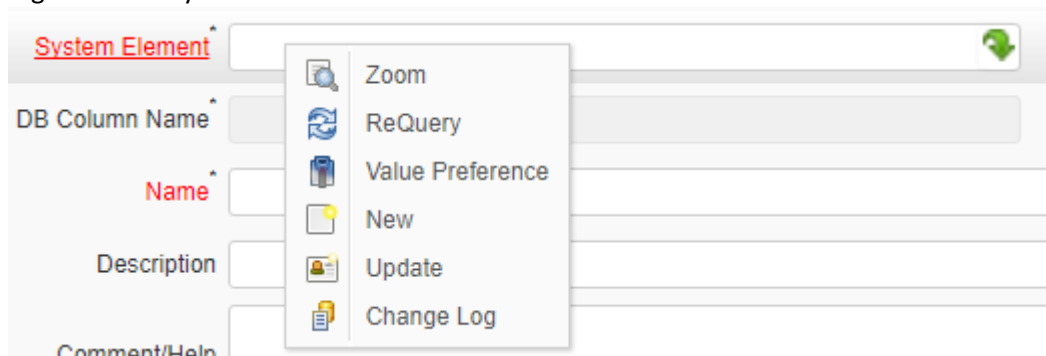
b. Fill the opened window according following picture ([TABLE NAME]=fi_demo:

c. Set **Length** to 36
d. In **Reference** Field select **String**
e. Uncheck **Updatable**

After you create all fields click on the button **Synchronize column.** If you obtain this message, you have created the table:

After this you can add more fields according your requests. You can use following types of **References**:

Name	Description
Account	Account Element
Amount	Number with 4 decimals
Assignment	Resource Assignment
Binary	Binary Data
Button	Command Button - starts a process

Color	Color element
Costs+Prices	Costs + Prices (minimum currency precision but if exists more)
Date	Date mm/dd/yyyy
Date+Time	Date with time
FileName	Local File
FilePath	Local File Path
ID	10 Digit Identifier
Image	Binary Image Data
Integer	10 Digit numeric
List	Reference List
Location (Address)	Location/Address
Locator (WH)	Warehouse Locator Data type
Memo	Large Text Editor - Character String up to 2000 characters
Number	Float Number
Product Attribute	Product Attribute
Quantity	Quantity data type
RowID	Row ID Data Type
Search	Search Field
String	Character String
Table	Table List
Table Direct	Direct Table Access
Text	Character String up to 2000 characters
Text Long	Text (Long) - Text > 2000 characters
Time	Time
URL	URL
Yes-No	CheckBox

How to create a search field to other table (like to Business Partner)

You need to use as system element the **primary key** of the referenced table (for Business Partner it is C_BPartner_ID). After it the **Reference** will be Search. You would need to specify **Dynamic Validation** to eliminate those records you do not want to add (summary or non active records).

How to create list field?

Again you use the **primary key** of the referenced table, but type of **Reference** will be Table Direct. You can also specify **Dynamic Validation**.

What is Dynamic Validation?

It is about creating additional condition. Dynamic validation is set in the window you see after clicking to the link **Dynamic Validation**.

The most common usage of **Validation code** is to formulate SQL condition to filter set of records to be displayed. The example is showing condition *Involve only active and non-summary business partners*.

Example

Create the table that will record all events you communicated with Business Partner. The types of events will be:

- Phone call
- Personal meeting
- Skype call
- Email
- Messenger contact

The recorded information about event will be:

- Date and time of event
- Content of the event
- Leader or responsible person for the event

Solution:

We create two tables:

FI_Type_Events:

- ad_client_id
- ad_org_id
- isactive
- created
- createdby
- updated
- updatedby
- fi_type_events_id [necessary to crate a new element]

- fi_type_events_uu [necessary to crate a new element]
- Name – it must be specified as Identifier and Selection Column, both with Sequence value to 1

FI_Events:

- ad_client_id
- ad_org_id
- isactive
- created
- createdby
- updated
- updatedby
- fi_events_id
- fi_events_uu
- DateEvent [need to be created, reference: Date]
- Description
- Comment
- Business Partner (identification of Business Partner)
- Leader_ID [C_BPartner_ID, Validation IsEmployee = Y] – identification of responsible person
- Event Type [FI_Type_Events_ID, Table Dirrect]