Table of Contents

Part I Introduction 2

Part II Custom Queries 3

Part III Building a query 4

Index 0

1 Introduction

SQL queries can be made to TO3000 Version 9 database by TO3000 user. This feature is designed to be used by system administrators and IT staff since it requires basic knowledge of SQL.

Custom Query in TO3000 Version 9 is an advanced data export option, which can be used to obtain and export any data from TO3000 Version 9 database. Creating new queries requires some basic SQL knowledge, however, almost any TO3000 user can run already existing queries and export the data they gather.

Structured Query Language (SQL) is a language of structured requests. It is intended for working with relational databases, which constitute the sets of interrelated data, stored in tables.

Nowadays SQL is a part of a large number of programs, executed on various types of computers. "Owing to its elegance and machine independence, as well as to the industrial leaders support in relational base technology, SQL was acknowledged the standard language and will keep this position in the foreseeable future."[2000, Mastering SQL, Martin Grubber].

2 Custom Queries

Custom queries controls are located on the **Custom Queries** tab of **Advanced Settings** window of TO3000 Version 9.

To open **Custom Queries** tab, click **Custom Queries** in the left part of the **Advanced Settings** window. This will display the list of custom queries. Double-click any of the queries in the list to open this **Query Builder** window for this query.

Exporting and printing query results

To *export* the data gathered by a certain query, select this query and click **Export Data** button. This will run currently selected query and prompt you to select the format of output file. Select the output file type and proceed with exporting the data

To *print* the resulting data of a query, select it and click **Print Data** button. This will open **Print Preview** window. Click **Print** button to send the query results to chosen printer.

Loading and saving queries

Queries can be exported by saving them into PXQ files. You can also import queries from PXQ files to TO3000 Version 9 database.

To save query in a PXQ file, select the required query and click **Save** button.

To load a query from a PXQ file, click **Load** button and use **Open TO3000 Version 9 Query** window to locate the required PXQ file.

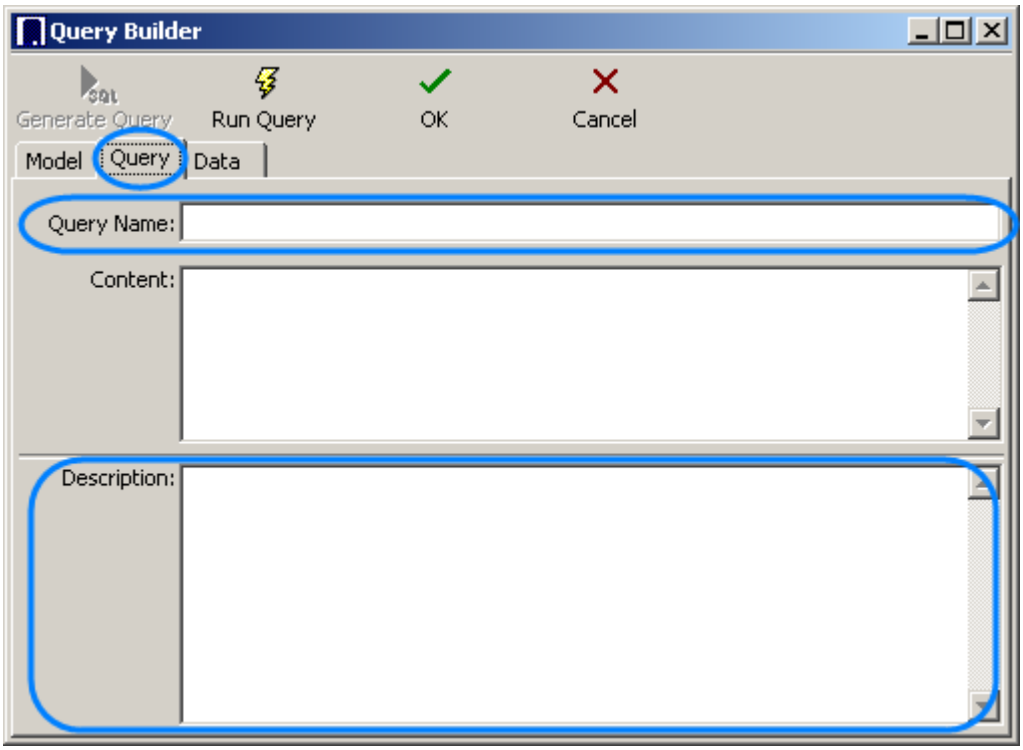
Note: Copying PXQ file to *\Translation Office 3000 Version 9.0\Admin\Queries* will not add this query to the database.

3 Building a query

User queries are built and managed with the help of **Custom Queries** section of **Advanced Settings** tab of the TO3000 Version 9. To begin creating a query click **New** button. This will open **Query Builder** window.

1. General query data

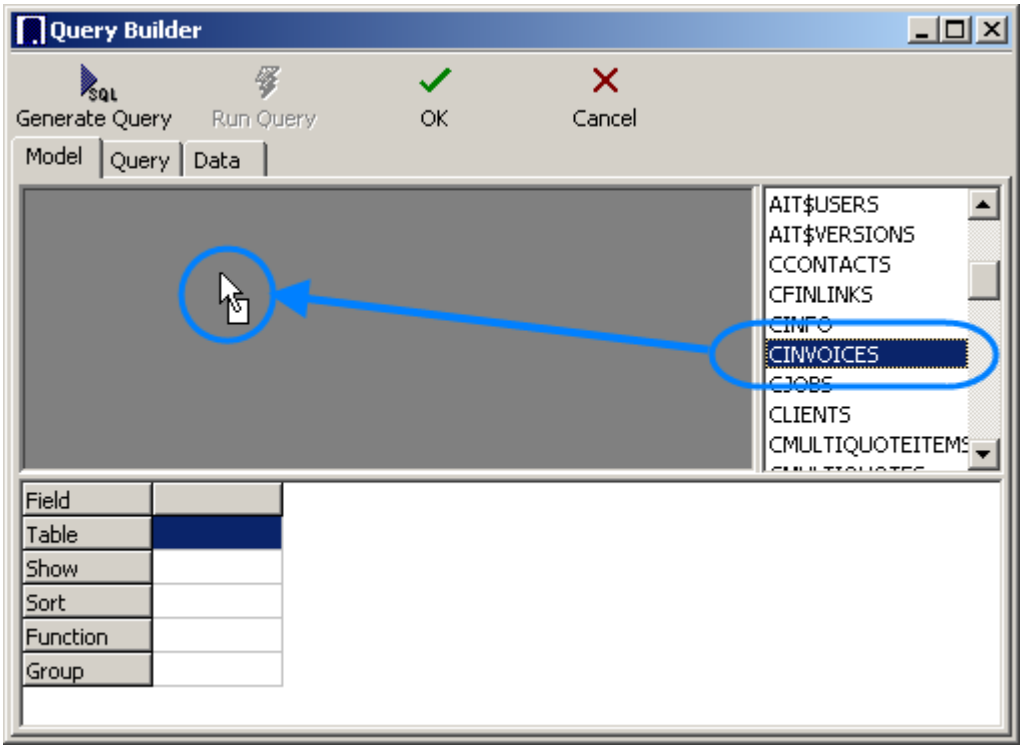
Enter the name of your query in the **Query Name** field, and (optionally) description of the new query in the **Description** field.



2. Adding tables to a custom query

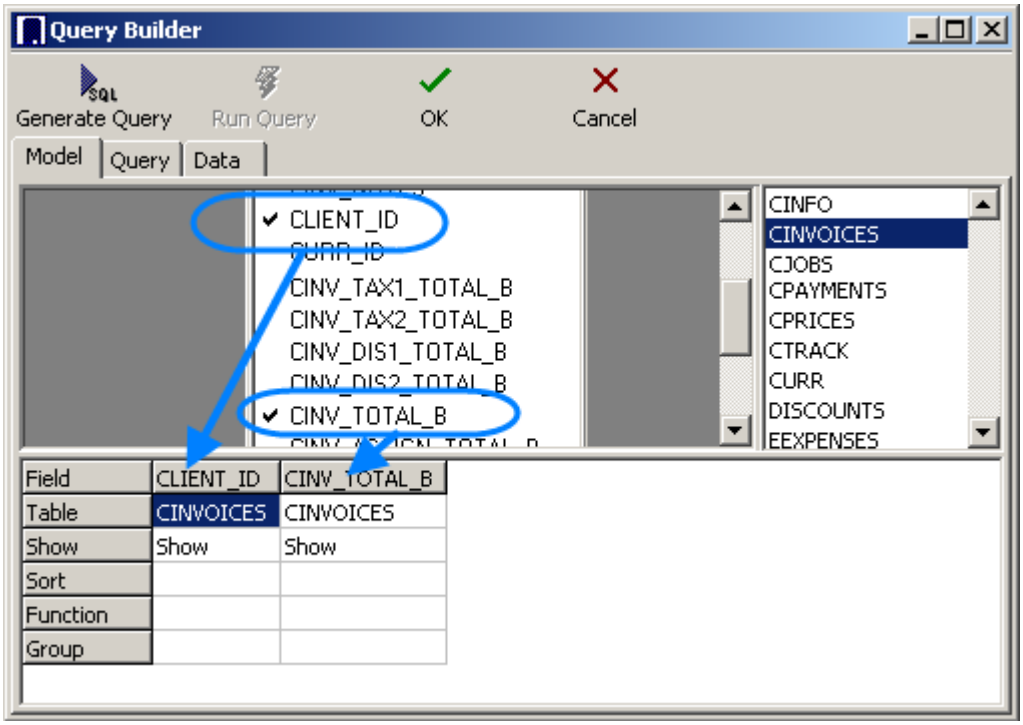
Click **Model** tab.

Database consists of tables, each of which has a number of fields containing certain data. Locate the required tables on the list to the right (in this case – *CINVOICES* table which contains fields with invoices data) and drag them to the gray field of the **Model** tab of **Query Builder** window.



4. Selecting fields to be added to query

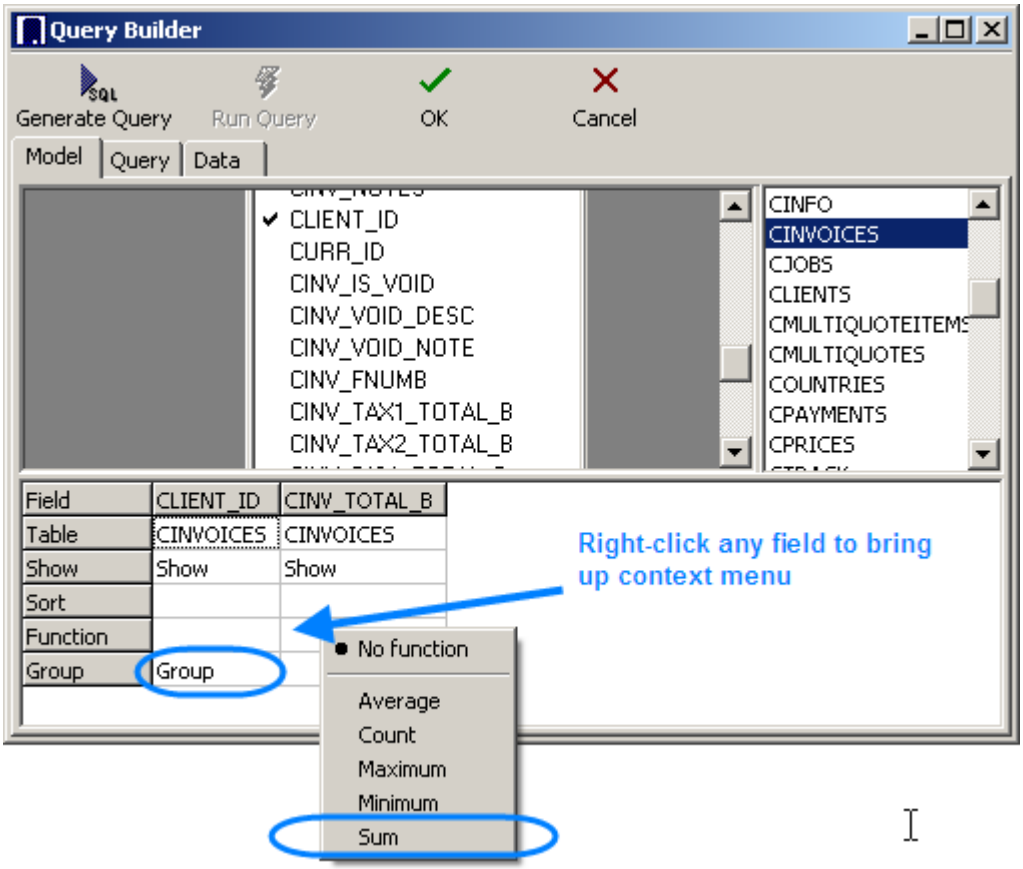
Select fields, which must be added to the new query (for example, fields *CLIENT_ID* and *CINV_TOTAL_B* — client ID and sum of the invoice in basic currency have been selected on the screenshot below) by clicking near the names of theses fields in the table windows. Selected fields will appear in the lower area, which represents the list of all fields added to this query.



5. Grouping data

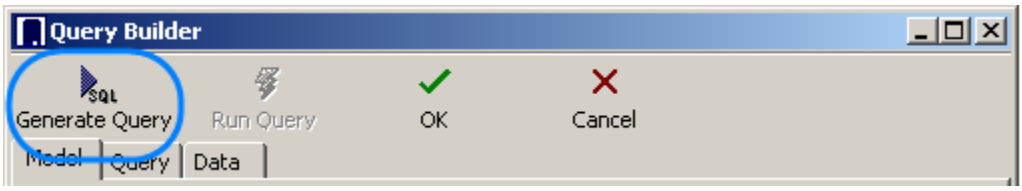
Right-click the Group cell in the table by which data must be grouped. For example, if invoice data is grouped by *CLIENT_ID*, it will be obtained and sorted by clients.

You can also apply function to query. Right-click function row of required column and then click the function which must be applied to that column. For instance Sum function applied to *CINV_TOTAL_B* will return sum of invoice totals.

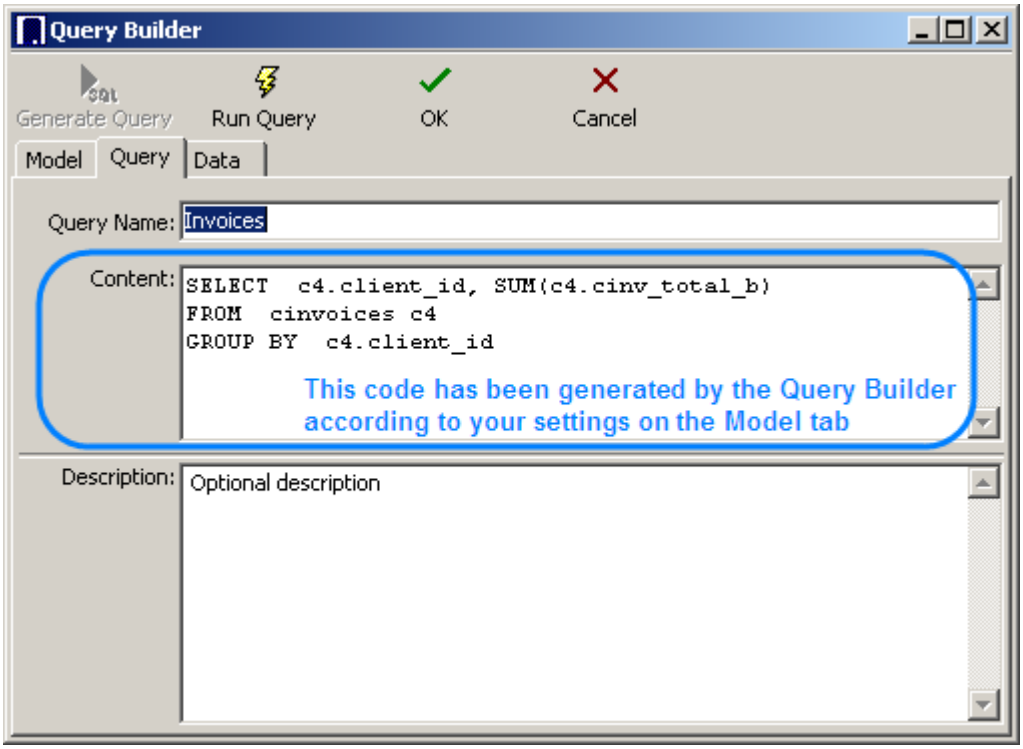


6. Generating query

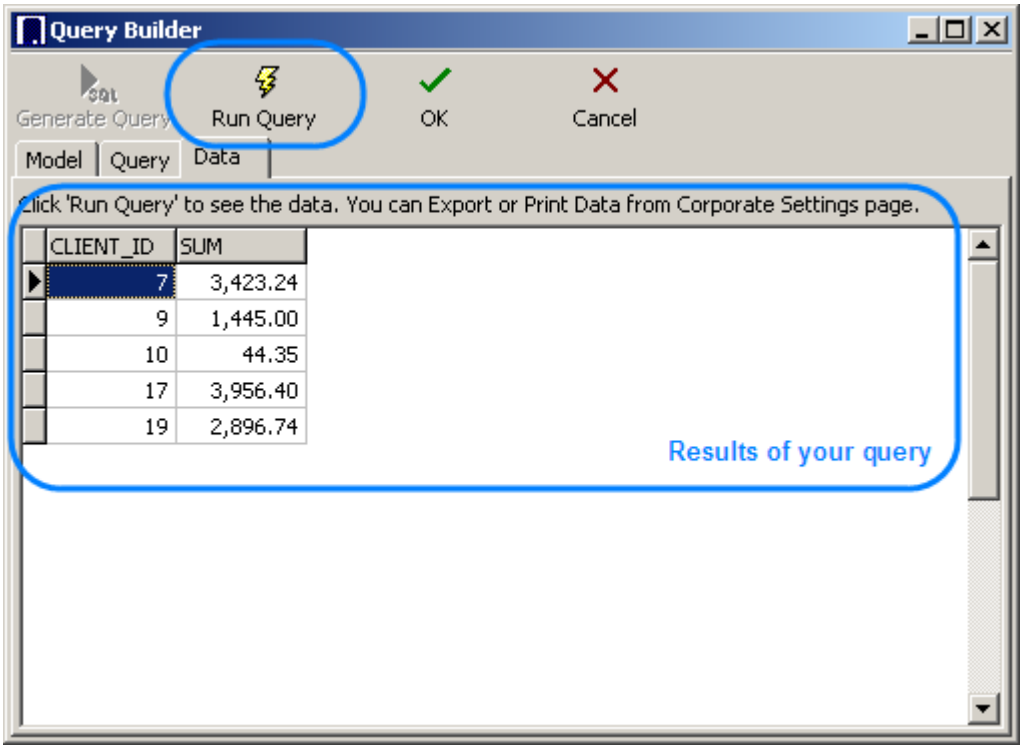
Click the **Generate Query** button.



Newly built SQL-query will appear.



Start executing the query (clicking **Run Query** button from the toolbar in the upper part of the window) and you will see the results.



Note: Although for most basic queries you are not required to write the SQL code of the query

manually, any additional code can be added by SQL-versed users to fully customize their queries. For example, the following strings can be added to this particular query:

```
SELECT FIRST 10 c4.client_id, SUM(c4.cinv_total_b),

 (select client_name from clients where client_id = c4.client_id)

FROM cinvoices c4

GROUP BY c4.client_id

ORDER BY 2 DESC
```

The code above will make your query do the following: Select first ten clients (first 10 client_id), count the sum of invoices for each, and display them in the order of sum reduction (ORDER BY 2 DESC).

Click **Run Query** button once again to see the new result.

Note: If you click **Generate SQL** after editing the query manually the query will be rebuilt based on Model settings and the results of any manual alterations of query code will be lost.

Click **OK** button to save the query under currently entered name.