

# microgen

## *DBCClarity Developer™*



## Product Overview and Examples

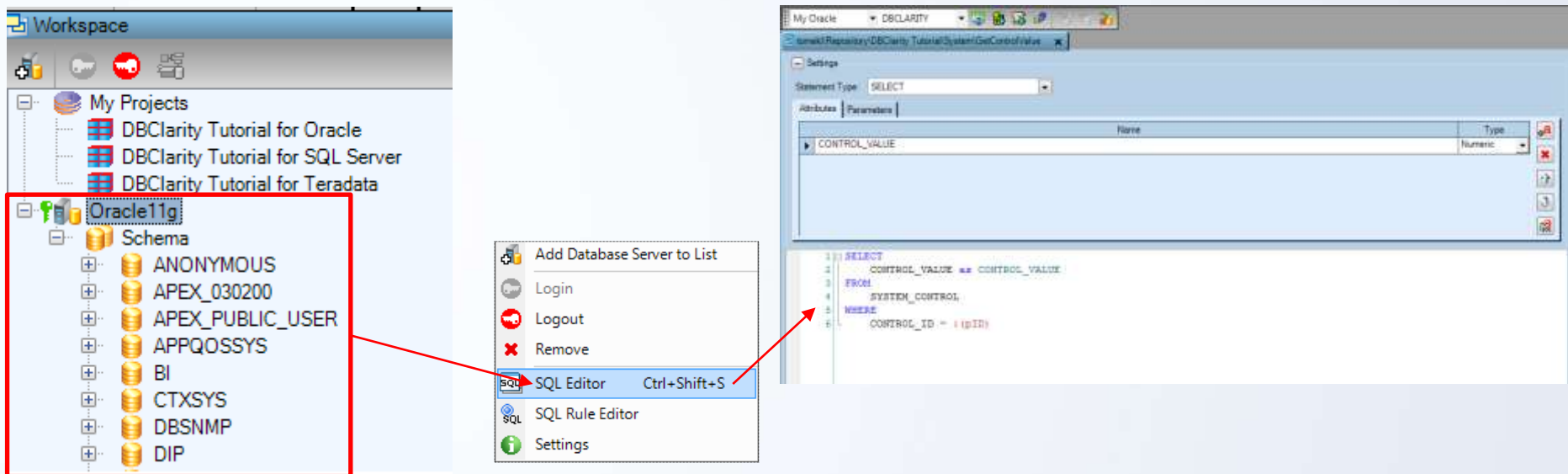
Introducing the SQL Editor

## SQL Editor

- Database Server Interface
- SQL Editor

# Database Server Interface

- Database Server connection in the Workspace navigator
  - Connect to external database
  - View schemas and tables
  - Drag-and-Drop between the database and the Database External Data Format (EDF)
  - Access the SQL Editor



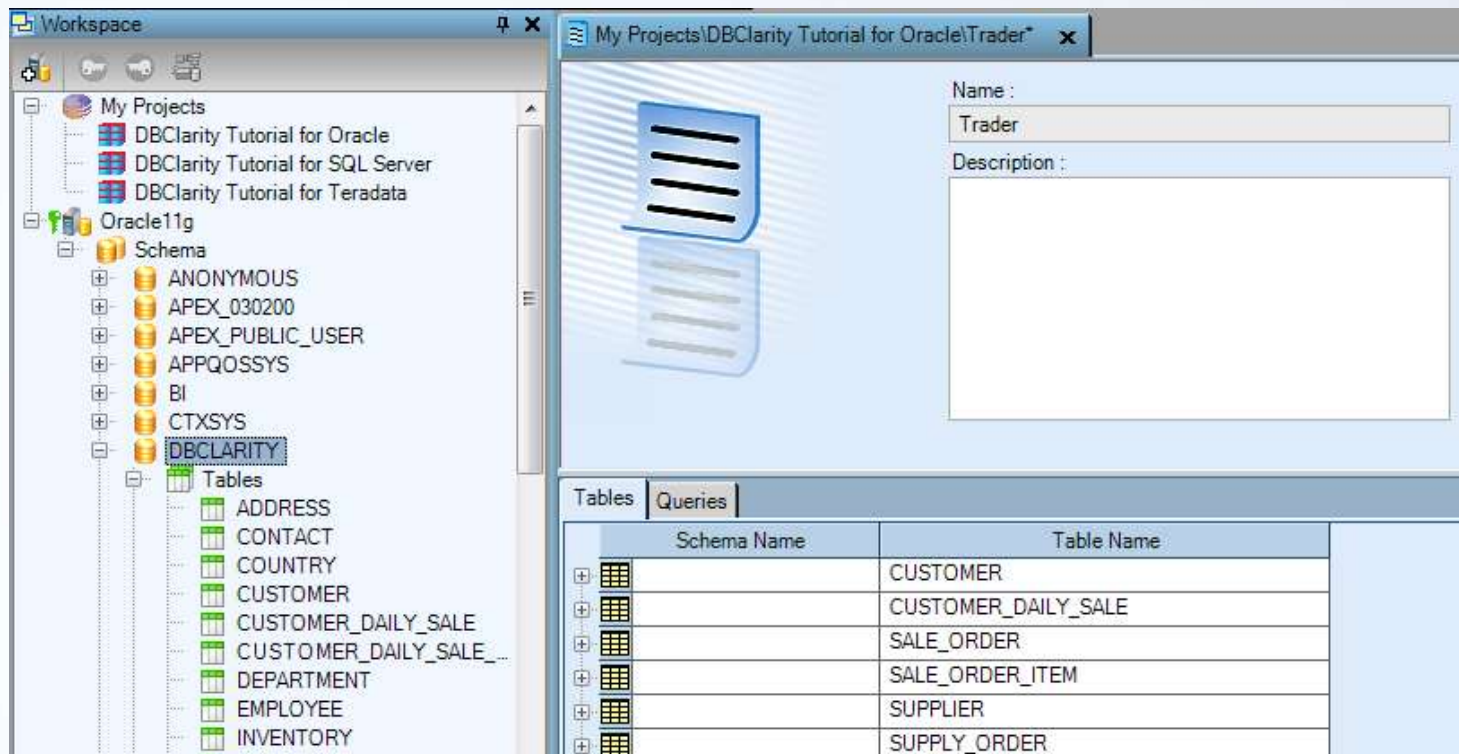
# External Data Format (EDF)

- Defines the structure of external data received from and sent to the database in the format of tables and columns
- Is based on existing database structures
- Is used in database deployment to create database structures

Database Name	Table Name	Column Name	Data Type	Max Length	Precision	Decimal Scale	Nullable	Default Value	Db Role	Sequence	Sequence name	Primary Key	Relation Table	Relation Column
	CUSTOMER													
	CUSTOMER_DAILY_SALE													
	SALE_ORDER	SALE_ORDER_ID	DECIMAL		38	0	<input type="checkbox"/>		ID	<input type="checkbox"/>		<input checked="" type="checkbox"/>		
		CUSTOMER_ID	DECIMAL		38	0	<input type="checkbox"/>		Foreign k			<input type="checkbox"/>	CUSTOMER	CUSTOMER_I
		ORDER_DATE	DATE				<input type="checkbox"/>		Normal			<input type="checkbox"/>		
		ORDER_STATUS	DECIMAL		38	0	<input checked="" type="checkbox"/>		Normal			<input type="checkbox"/>		
		SALESMAN	DECIMAL		38	0	<input checked="" type="checkbox"/>		Normal			<input type="checkbox"/>	EMPLOYEE	EMPLOYEE_I
		TOTAL	DECIMAL		38	0	<input checked="" type="checkbox"/>		Normal			<input type="checkbox"/>		
Constraint Name	Type	Columns	Condition											
PrimaryKey	Primary Key	SALE_ORDER_ID	<input type="checkbox"/>											
Database Name	Table Name													
	SALE_ORDER_ITEM													
	SUPPLIER													
	SUPPLY_ORDER													
	SUPPLY_ORDER_ITEM													
	SUPPLIER_PRODUCT													
	PRODUCT													
	INVENTORY													
	COUNTRY													
	ADDRESS													
	CONTACT													

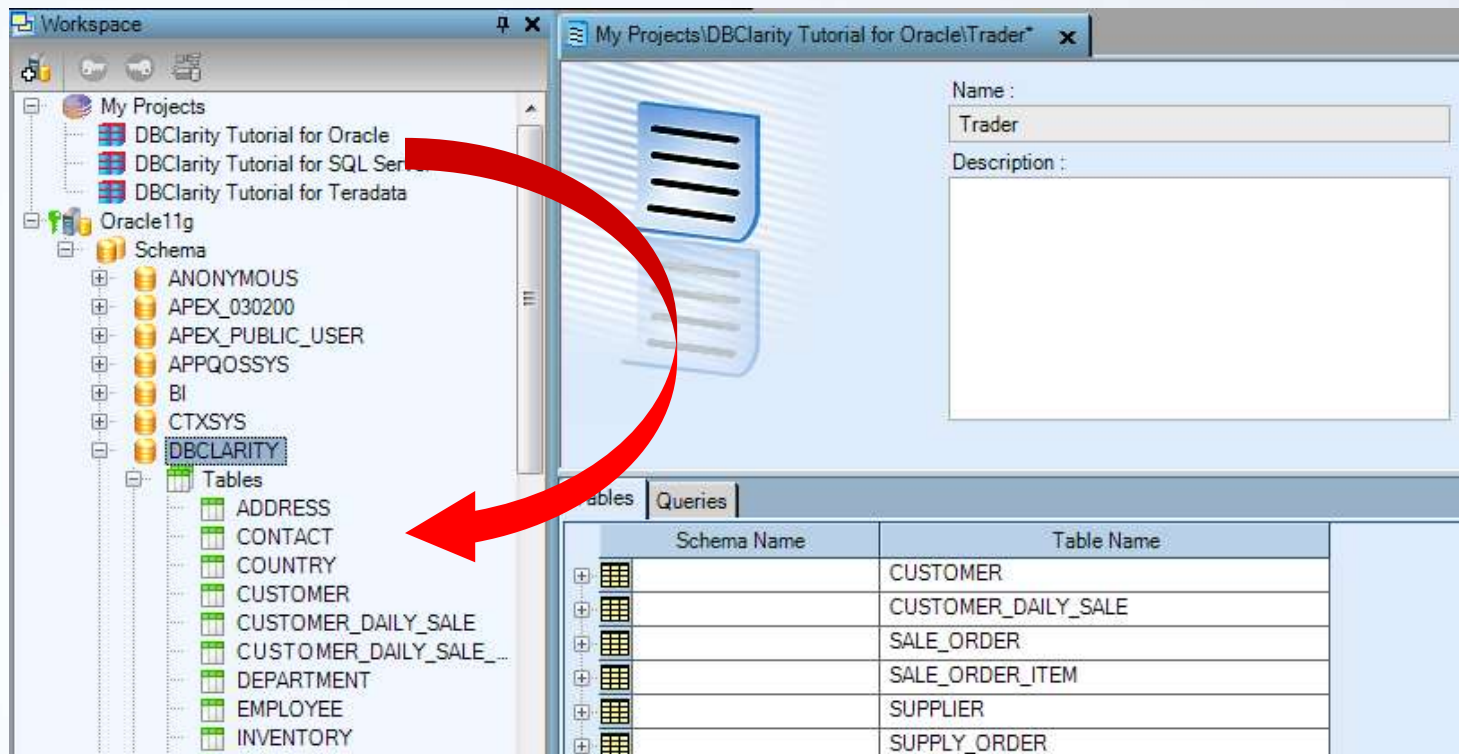
# Database External Data Format

- Create the Database EDF project element based on existing database tables using drag-and-drop from the Database Server Node onto the EDF Editor

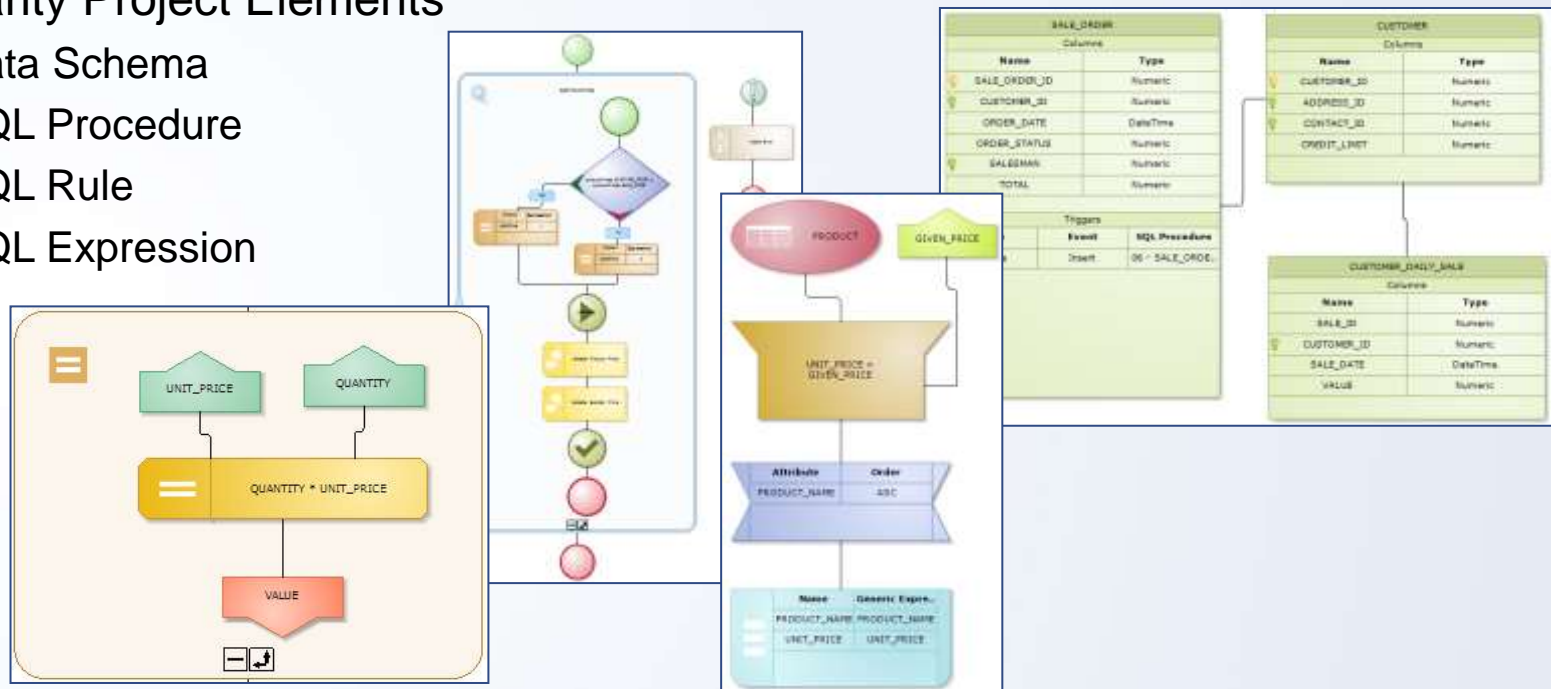


# Database External Data Format

- Create the database structure, tables, procedures, and create or update views, by deploying the database project to the Database Server



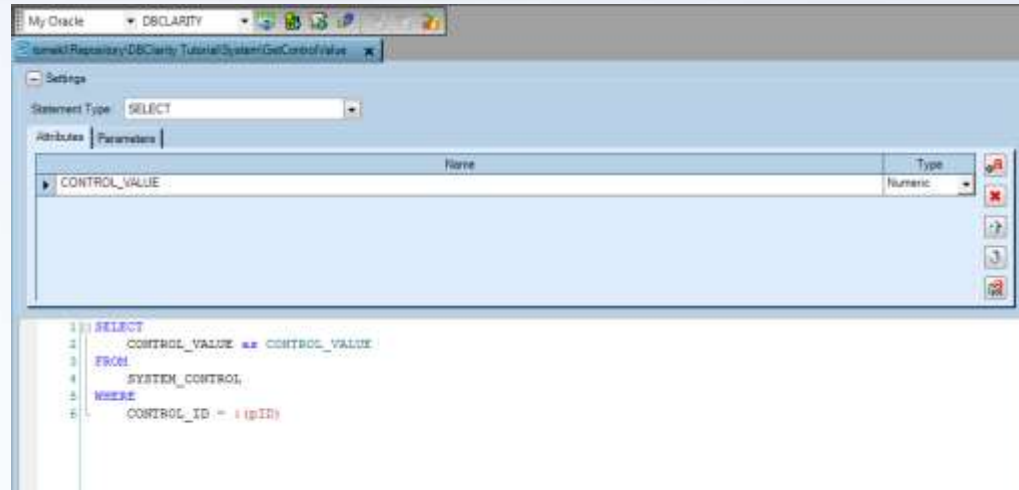
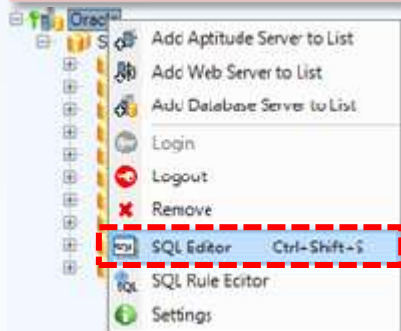
- Database Deployment
  - Generate and execute SQL scripts on the selected database
  - SQL Procedure deployed as a database stored procedure or trigger
  - SQL Rule deployed as a database stored procedure or database view
- DBClarity Project Elements
  - Data Schema
  - SQL Procedure
  - SQL Rule
  - SQL Expression



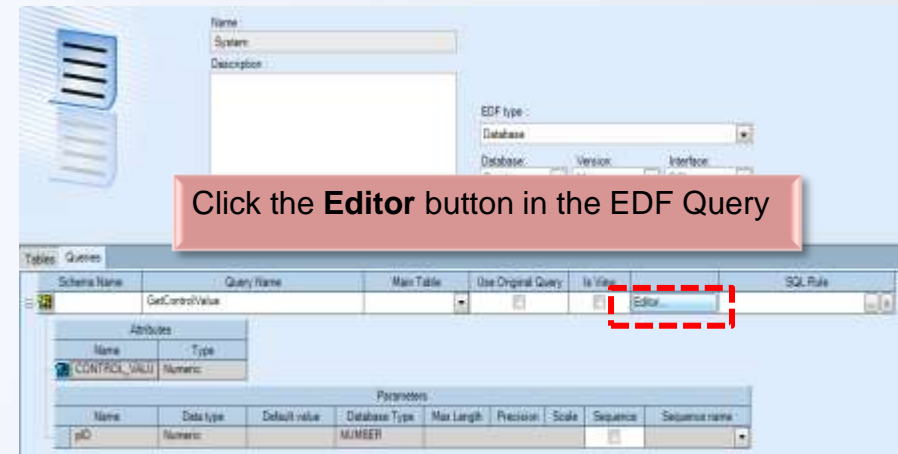
- Database Deployment
  - Generate and execute SQL scripts on the selected database
  - SQL Procedure deployed as a database stored procedure or trigger
  - SQL Rule deployed as a database stored procedure or database view
- Database Stored Procedure
  - Defines a group of SQL statements compiled into a single execution.
  - Provides a consist implementation of operations used in different applications and tasks
- Database Trigger
  - A type of stored procedure executed automatically as a response to events occurring in a database
- Database View
  - A virtual table created in the database showing the result-set of a query defined in the SQL Rule and deployed as a View

# Accessing the SQL Editor

Right click on the **Database Server**



Use the **Database Generator** toolbar in the editors



Click the **Editor** button in the EDF Query

The SQL Editor supports:

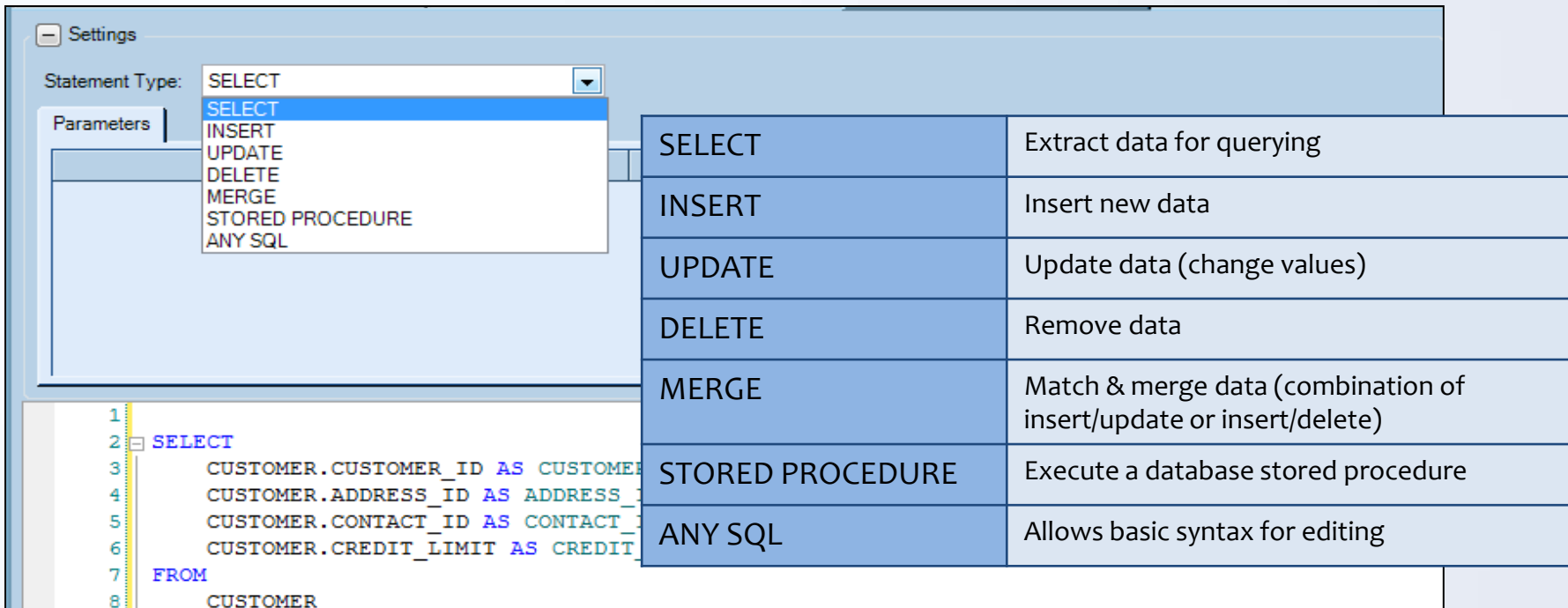
- Syntax highlighting
- Auto complete
- SQL Statement settings
- Statement execution
- Results view
- Add and parse parameters
- Add and parse attributes

The screenshot displays the SQL Editor interface. At the top, the 'Statement Type' is set to 'SELECT'. Below this is a 'Parameters' table with columns for Name, Type, Default Value, Nullable, Max. Prec, and Size. The main editor area shows a SQL query with syntax highlighting. A tooltip is visible over the 'CUSTOMER\_ID' column, indicating it is a NUMBER(0) type from the TRAINING.ADDRESS\_DETAILS table. At the bottom, the 'Results' section shows a table with 7 columns and 5 rows of data.

```
1 ---  
2 --- Table ADDRESS_DETAILS from oracle  
3 ---  
4  
5 SELECT  
6 ADDRESS_DETAILS.CUSTOMER_ID AS CUSTOMER_ID,  
7 ADDRESS_DETAILS ADDRESS1 1,  
8 ADDRESS_DETAILS ADDRESS2 2,  
9 ADDRESS_DETAILS CITY  
10 ADDRESS_DETAILS COUNTRY  
11 ADDRESS_DETAILS CUSTOMER_ID  
12 ADDRESS_DETAILS POSTCODE  
13 FROM  
14 ADDRESS_DETAILS TELEPHONE_NUMBER  
15
```

CUSTOMER_ID	ADDRESS1	ADDRESS2	CITY	COUNTRY	TELEPHONE_NUMBER	POSTCODE
1	High Pl	1	London	Great Britain	44875690567	AAA2
1	Long St	12	Liverpool	Great Britain	44100020002	AAA3
2	Green St	124	London	Great Britain	44654764788	4AAA
3	Pond Ct	45	London	Great Britain	44543688888	AAA7B
3	London St	34	Bristol	Great Britain	44330957564	55AAA
3	Park St	12	Oxford	Great Britain	44561002992	88G5
4	Park Lane	47	Bristol	Great Britain	44666677777	88G8
5	Garden Pl	3	London	Great Britain	44909090909	98G8

- Access and manipulate data in database tables using common SQL commands known as the Statement type in the SQL Editor
- Choose the Statement type and drag-and-drop the database schema (from the Database Server Node) onto the SQL Editor
- Edit the syntax provided automatically, if required

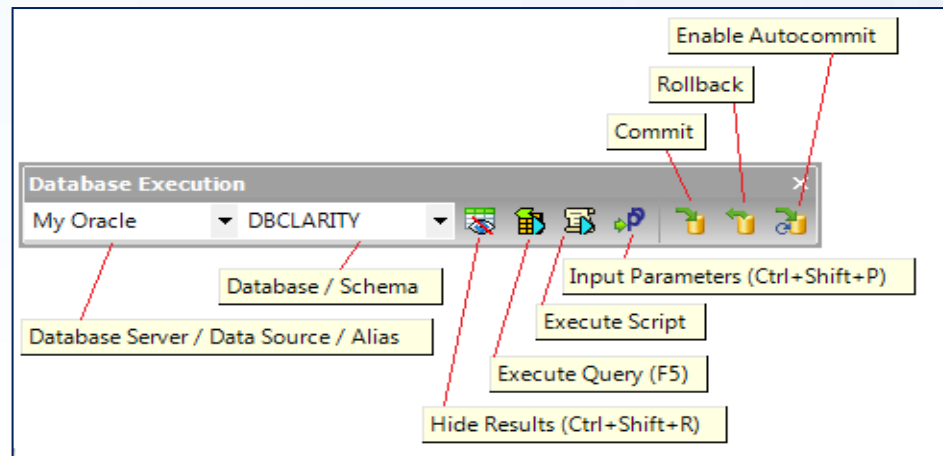





The screenshot shows the SQL Editor interface. On the left, there is a 'Settings' panel with a 'Statement Type' dropdown menu. The dropdown menu is open, showing a list of options: SELECT, INSERT, UPDATE, DELETE, MERGE, STORED PROCEDURE, and ANY SQL. Below the dropdown is a 'Parameters' section. In the main editor area, a SQL query is visible, starting with '1 SELECT' on line 2, followed by columns from the 'CUSTOMER' table: 'CUSTOMER.CUSTOMER\_ID AS CUSTOMER...', 'CUSTOMER.ADDRESS\_ID AS ADDRESS...', 'CUSTOMER.CONTACT\_ID AS CONTACT...', and 'CUSTOMER.CREDIT\_LIMIT AS CREDIT...'. Line 7 starts with 'FROM' and line 8 with 'CUSTOMER'.

SELECT	Extract data for querying
INSERT	Insert new data
UPDATE	Update data (change values)
DELETE	Remove data
MERGE	Match & merge data (combination of insert/update or insert/delete)
STORED PROCEDURE	Execute a database stored procedure
ANY SQL	Allows basic syntax for editing

# SQL Editor – Statement Execution

- Once the SQL script is prepared,  Execute and  Commit to the database
- Choose the database server and database schema



-  Enable Autocommit: Saves data to the selected database automatically
-  Disable Autocommit: Does not save the changes until selected manually
-  Rollback: Will undo the most recent SQL operations executed on the database

# SQL Editor – Results View



- Validate, execute and commit SQL statements directly onto a database and view the results

The screenshot displays the SQL Editor interface. At the top, the 'Statement Type' is set to 'SELECT'. Below it, a 'Parameters' table is shown with one parameter: 'GIVEN\_PRICE' of type 'NUM' with a default value of '38'. The SQL query is as follows:

```
1 SELECT
2   NEWPRODUCTS.PRODUCT_NAME AS PRODUCT_NAME,
3   NEWPRODUCTS.UNIT_PRICE AS UNIT_PRICE
4 FROM
5   NEWPRODUCTS NEWPRODUCTS
6 WHERE
7   NEWPRODUCTS.UNIT_PRICE > : (GIVEN_PRICE)
8 ORDER BY
9   NEWPRODUCTS.PRODUCT_NAME
```

Below the query, the status bar shows 'Line: 1/9 Column: 1', 'Selection: 0', 'Insert', and '[Read Only]'. The 'Results' section contains a table with the following data:

PRODUCT_NAME	UNIT_PRICE
KINDLE	152
LAPTOP	650
iPAD	400
iPOD	200

- SQL scripts generated in Microgen DBClarity can be exported and imported for reuse
- In the SQL Editor:
  - Export to File 
  - Import from File 
- Microgen DBClarity supports two file types:
  - .sql files can be used in Microgen DBClarity's SQL Editor and third party editors
  - .aptsql files can only be used in Microgen DBClarity

# Database External Data Format

- EDF Queries perform pre-processing directly on the database to simplify processes required in Microgen DBClarity
- The Database EDF has extended support for SQL Statements:
  - Use the SQL Editor to manually define queries
  - Choose a defined SQL Rule project element as the query

Open SQL Editor or choose a SQL Rule

Database Name	Query Name	Main Table	Use Original Query	Is View	SQL Rule
	GetControlValue		<input type="checkbox"/>	<input type="checkbox"/>	Editor... <input type="button" value="..."/> <input type="button" value="&gt;"/>

Attributes	
Name	Type
CONTROL_VALU	Numeric

Parameters								
Name	Data type	Default value	Database Type	Max Length	Precision	Scale	Sequence	Sequence name
pID	Numeric		NUMBER				<input type="checkbox"/>	

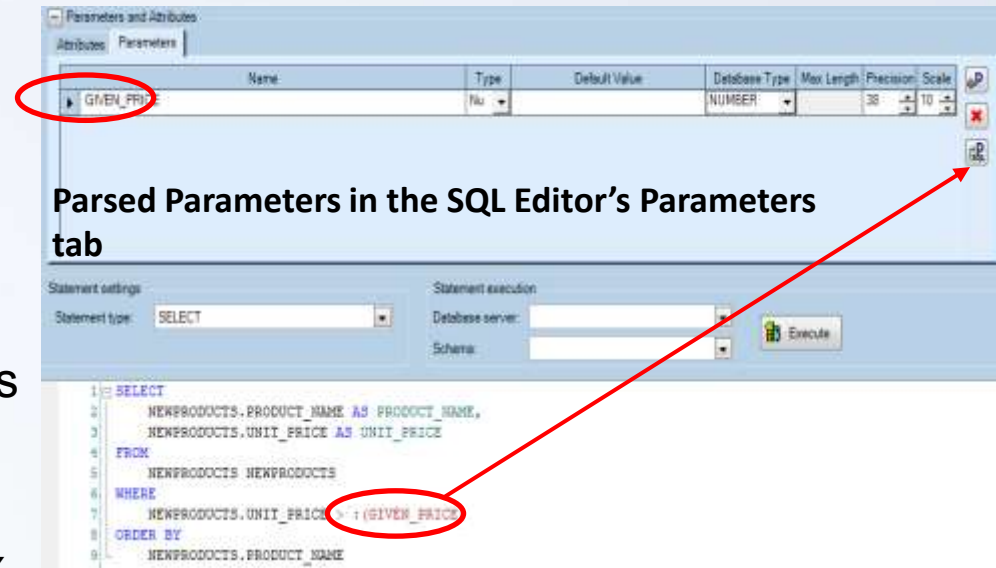
Database Name	Query Name	Main Table	Use Original Query	Is View	SQL Rule
	GetProducts		<input type="checkbox"/>	<input type="checkbox"/>	Editor... <input type="button" value="..."/> <input type="button" value="&gt;"/> 01 - SELECT

Attributes	
Name	Type
PRODUCT_NAME	String
UNIT_PRICE	Numeric

Parameters								
Name	Data type	Default value	Database Type	Max Length	Precision	Scale	Sequence	Sequence name
GIVEN_PRICE	Numeric		DECIMAL		38	0	<input type="checkbox"/>	

# SQL Editor – Parameters

- Used for filtering
- If defined in a SQL Rule, then automatically parsed to the EDF-level when the rule is chosen as the query
- Can be added in the SQL Editor's Parameter tab and be available for use in the syntax
- Parameters defined in the syntax manually can be parsed



Database Name	Query Name	Main Table	Use Original Query	Is View	SQL Rule
	GetProducts		<input type="checkbox"/>	<input type="checkbox"/>	01 - SELECT

Attributes	
Name	Type
PRODUCT_NAME	String
UNIT_PRICE	Numeric

Parameters									
Name	Data type	Default value	Database Type	Max Length	Precision	Scale	Sequence	Sequence name	
GIVEN_PRICE	numeric		DECIMAL	38	0		<input type="checkbox"/>		

**Parsed Parameters in the EDF Query**

# SQL Editor - Attributes

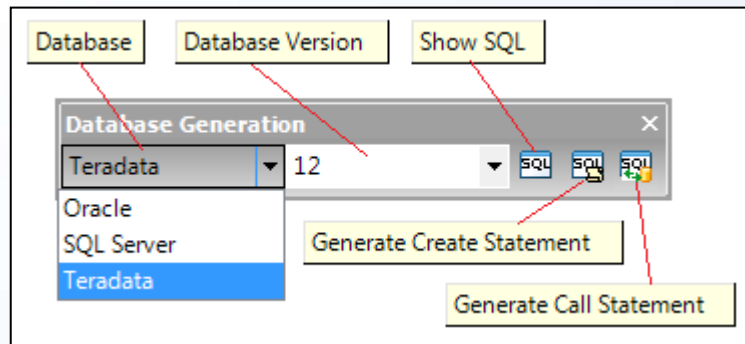
- If defined in a query are returned as a result of query execution and mapped to Data Object attributes in the Mapping
- If defined in a SQL Rule, are automatically parsed to the EDF-level when the rule is chosen as the query
- Can be created in the SQL Editor's Attributes tab and used in the syntax
- Can be parsed when defined in the syntax manually



Database Name	Query Name	Main Table	Use Original Query	Is View	Editor...	SQL Rule		
	GetProducts		<input type="checkbox"/>	<input type="checkbox"/>	Editor...	01 - SELECT		
Attributes								
Name	Type							
PRODUCT_NAME	String							
UNIT_PRICE	Numeric							
Parameters								
Name	Data type	Default value	Database Type	Max Length	Precision	Scale	Sequence	Sequence name
GIVEN_PRICE	Numeric		DECIMAL		38	0	<input type="checkbox"/>	

**Parsed Attributes in the EDF Query**

- The Database Generation Toolbar is available in SQL Rule and SQL Procedure diagrams
- Use the toolbar to
  - Generate code specific for the chosen Database and Database Version
  - Show SQL to view the SQL syntax behind the SQL Rule design (read-only)
  - Generate Create Statement for SQL Rules and SQL Procedures
  - Generate Call Statement for SQL Procedures



# More Information

- Download the product and find more information, including product workshops and training videos, visit the Microgen website [www.microgen.com/dbclarity](http://www.microgen.com/dbclarity)
- Send your product questions and feedback to the Microgen team at [dbclarity.feedback@microgen.com](mailto:dbclarity.feedback@microgen.com)
- Contact Microgen Support for all DBClarity Developer related support queries at [dbclarity.support@microgen.com](mailto:dbclarity.support@microgen.com)
- Follow DBClarity Developer on Twitter @**MCGN\_DBClarity**

This document is designed to provide a training overview of Microgen DBClarity Developer™. Information in this document is subject to change without notice and does not represent a commitment on the part of Microgen.

The information contained in this document is proprietary and confidential to Microgen plc and must not, therefore, be disclosed to any third-party without the express written permission of Microgen. In addition, no part of this document may be reproduced or transmitted in any form or by any means electronic or mechanical including photocopying, recording or information storage and retrieval systems, for any purpose other than the recipient's personal use without the express written permission of Microgen.