📊 DiagramER

Database Entity Relationship Diagram Tool

Getting Started Guide - v1.0

Quick Navigation

🎯 Getting Started

What is DiagramER?

DiagramER is a modern web-based tool for designing and visualizing database schemas. It allows you to create Entity-Relationship (ER) diagrams by adding tables, defining columns with various data types, and establishing relationships between tables. Once complete, you can export your diagram in multiple formats including JSON and SQL.

Key Features

- ✓ Interactive canvas for drag-and-drop table creation
- ✓ Support for multiple SQL data types
- ✓ Visual relationship lines between tables
- ✓ Real-time JSON schema generation
- ✓ SQL script generation from diagrams
- ✓ Export diagrams as images or JSON files
- ✓ Load previously saved diagrams
- ✓ Add descriptive text labels anywhere on the canvas
- ✓ Customizable column properties (Primary Key, Nullable, Default Values)

📝 Creating Your First Diagram

Follow these simple steps to start building your database diagram:

1 Launch DiagramER

Open the DiagramER application in your web browser. You'll see a clean interface with a toolbar on the left and a blank canvas on the right.

2 Add Your First Table

Click the **Add Table** button in the left panel. A new table will appear in the center of the canvas with a default name (e.g., "Table1").

3 Customize the Table

Double-click on the table title to rename it. Use meaningful names that represent the entity (e.g., "Users", "Products", "Orders"). Press Enter or click away to save the name.

4 Add Columns

Click the **+ Add Column** button at the bottom of each table to define database columns.

🗄️ Working with Tables

Moving Tables

Click on the table header (the title area) and drag it to reposition the table on the canvas. This helps you organize your diagram layout for better visualization.

Resizing Tables

Look for the small resize handle in the bottom-right corner of each table. Drag it to adjust the table's width and height according to your preference.

Deleting Tables

Click the ✕ button in the top-right corner of a table to remove it. Any relationships connected to this table will also be removed.

> 💡 **Pro Tip**
>
> Organize related tables together on the canvas for better visual clarity. You can rearrange them at any time.

➕ Adding Columns

Columns are the fields within each table. Here's how to work with them:

Column Properties

Property	Description	Example
Name	The column name in your database	UserID, EmailAddress
DataType	The SQL data type (int, varchar, datetime, etc.)	int, varchar, datetime
Length	Maximum length (for varchar and similar types)	50, 255
Default Value	Default value if not provided	GETDATE(), 0, 'Unknown'
Null	Can the column accept NULL values?	Yes, No
PK (Primary Key)	Is this column a primary key?	Yes, No

Steps to Add a Column:

1. Click **+ Add Column** at the bottom of a table
2. Enter the column name in the first field
3. Select the data type from the dropdown
4. Set the length (if applicable)
5. Enter a default value (optional)
6. Mark if it's nullable or a primary key

Editing Columns

Click on any column field to edit it. Changes are automatically saved to the JSON output.

Deleting Columns

Click the ✕ button on the right side of any column row to remove it.

Reordering Columns

Select a row and use the up (▲) and down (▼) arrow buttons at the bottom of the table to rearrange column order. Click on a column row to select it first.

> ⚠️ **Important**
>
> Ensure that each table has at least one Primary Key column to establish relationships properly. When setting a relation both columns must have the same datatype defined.

🔗 Creating Relationships

Understanding Relationships

Relationships define how tables connect to each other. They show foreign key constraints and data dependencies. For example, an "Orders" table might have a relationship to the "Users" table via a UserId column.

Creating a Relationship

Step-by-step:

1. In the source table, locate the column that will act as the foreign key
2. Look for the drag handle (●) on the left side of that column
3. Click and drag from that handle to a column in the target table
4. Release it over the handle (●) of the target column to create the relationship
5. A line will appear connecting the two columns

Viewing Relationship Details

Hover over relationship lines to see the connection details. The line shows the cardinality (one-to-many, many-to-many) of the relationship.

Removing Relationships

Click on a relationship line to select it, then press Delete or click the delete button.

📋 Managing Your Diagram

Adding Text Labels

Use the **Add Text Label** button to add descriptive notes anywhere on your canvas. This is useful for adding comments or explaining specific parts of your schema. Double-click a label to edit it, and click the ✕ to delete it.

Canvas Navigation

- **Pan:** Click and drag on empty canvas space to move around
- **Zoom:** Use the lower right buttons (+) or (-) to zoom in or zoom out
- **Reset:** Click the (%) button to return to the default zoom level

JSON Schema Preview

The left panel shows a real-time JSON representation of your diagram. This is useful for understanding the structure and can be used by developers for integration.

💾 Exporting & Sharing

Copy JSON to Clipboard

> 📋 **Copy**
> Copies the JSON schema to your clipboard. Paste it into your code or documentation.

Download JSON File

> ⬇️ **Download**
> Saves your diagram as a `.json` file. Use this to backup or share your design with team members.

Load JSON File

> ⬆️ **Load**
> Opens a previously saved JSON file to restore your diagram. Perfect for continuing work on existing designs.

Export Diagram as Image

> 💾 **Export Diagram**
> Exports your current diagram as an image (PNG/JPG) or as PDF suitable for documentation and presentations.

Generate SQL Script

> 🗄️ **Generate SQL**
> Creates a SQL CREATE TABLE script based on your diagram. This script can be directly executed in your database to create the actual schema.

💡 Tips & Tricks

Best Practices

- ✓ Use clear, descriptive column names
- ✓ Always define a Primary Key for each table
- ✓ Group related tables together visually
- ✓ Use text labels to document complex relationships
- ✓ Regularly save your work by downloading the JSON file
- ✓ Review generated SQL before implementing in production

Common Use Cases

1. Database Design: Plan your database schema before development begins

2. Documentation: Export diagrams to include in project documentation

3. Team Collaboration: Share JSON files with team members for review and feedback

4. Code Generation: Use the SQL script to quickly set up your database

5. Architecture Planning: Visualize complex data relationships before implementation

Troubleshooting

Q: Why can't I create a relationship?

A: Ensure both columns exist and that you're dragging from a column with the drag handle (●) over the handle on a target column.

Q: How do I clear the entire canvas?

A: Delete tables individually by clicking the × button. You can also load an empty diagram or refresh the page.

Q: Can I edit the generated SQL before creating the database?

A: Yes! The generated SQL is just a starting point. You can copy it and modify it as needed before execution.

Q: Can I import an existing database schema?

A: Yes, DiagramER allows you to download a JSON file, so you can load it later and continue working or modifying your diagram.

Getting Help

For issues, feature requests, or suggestions, visit the About section and send an email via the contact button at the bottom of the page.

DiagramER - Database Entity Relationship Diagram Tool

Developed with ❤️ by Migra | Powered by Blazor and .NET 10