

teknologi
informasi
UNIVERSITAS MUHAMMADIYAH YOGYAKARTA

Modul Praktikum

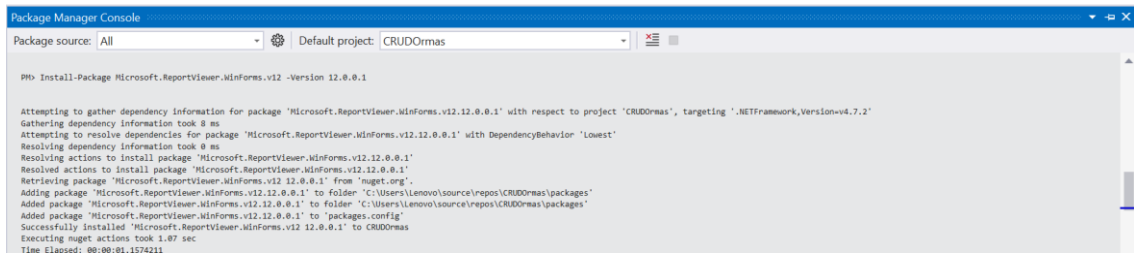
Pengembangan Aplikasi Basisdata

Apriliya Kurnianti S.T., M.Eng

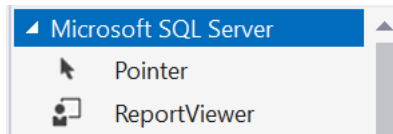


Daftar Isi

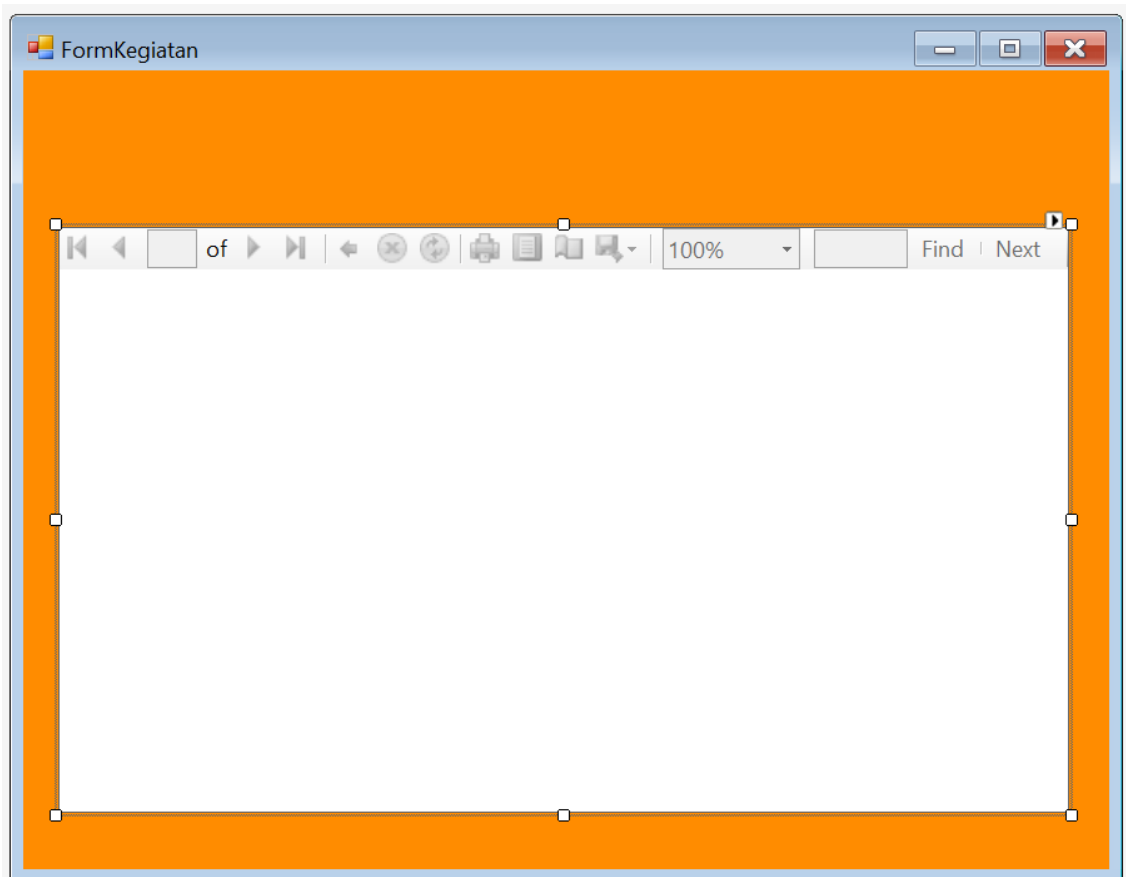
Report & Export Data (CSV, PDF)	1
Tujuan Praktikum	1
1. Menampilkan laporan dari database menggunakan ReportViewer.....	1
2. Meng-export data ke format CSV dan PDF menggunakan ReportViewer.	1
Alat dan Bahan	1
Langkah-Langkah Praktikum	1



c. Pada bagian toolbox akan muncul tampilan baru sebagai berikut :



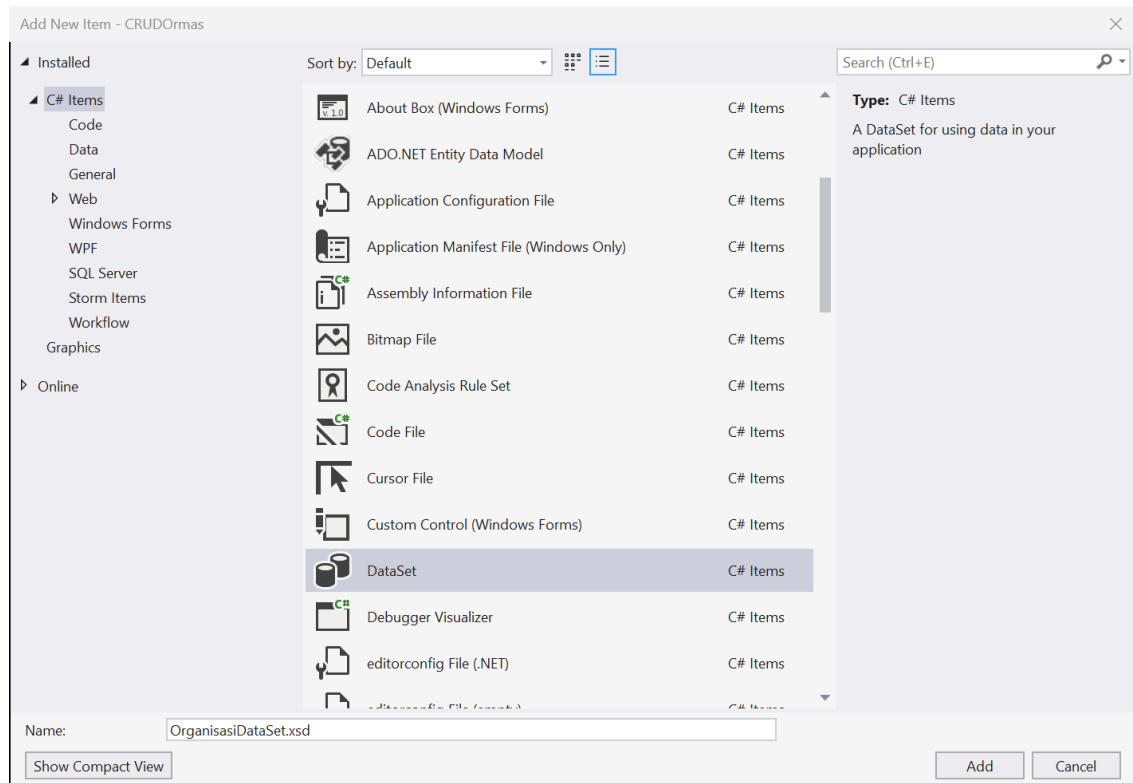
3. Buat form baru :
 - a. Tambahkan ReportViewer ke form Anda.
 - b. Buka Toolbox, lalu cari ReportViewer di kategori Reporting.
 - c. Tarik dan lepas ReportViewer ke form Anda.



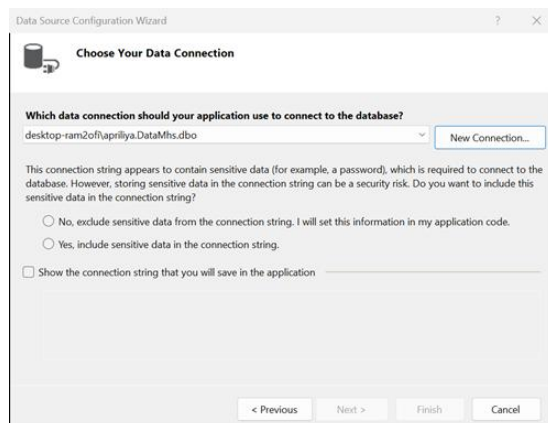
4. Buat dataset
 - a. Klik kanan pada proyek di Solution Explorer.

- b. Pilih Add > New Item.
- c. Pilih Data dari kategori item.
- d. Pilih Dataset dan beri nama seperti OrganisasiDataSet.xsd.

Dataset ini akan digunakan untuk mengonfigurasi struktur data yang digunakan dalam laporan.



- e. Di jendela Dataset Designer, tambahkan TableAdapter untuk tabel yang ingin Anda masukkan ke laporan.
- f. Buat koneksi database:
 - Pada wizard yang muncul, pilih New Connection.



- Isi connection string sesuai dengan database Anda dan klik Next.

- Pada bagian ‘Choose a Command Type’ pilih ‘Use SQL Statement’

- Tambahkan Query berikut :

TableAdapter Configuration Wizard

Enter a SQL Statement
The TableAdapter uses the data returned by this statement to fill its DataTable.

Type your SQL statement or use the Query Builder to construct it. What data should be loaded into the table?

What data should be loaded into the table?

```

SELECT
    o.ID_Organisasi,
    o>NamaOrganisasi,
    k.Jenis,
    k.Jumlah,
    k.Tanggal
FROM
    Organisasi o
JOIN
    Keuangan k ON o.ID_Organisasi = k.ID_Organisasi;

```

Advanced Options... Query Builder...

< Previous Next > Finish Cancel

- Pilih Query Builder lalu pilih Execute Query

Query Builder

o k

* (All Columns)
☒ ID_Organisasi
☒ NamaOrganisasi
☐ Deskripsi
☐ TahunBerdiri

* (All Columns)
☐ ID_Keuangan
☐ ID_Organisasi
☒ Jenis
☒ Jumlah

Column	Alias	Table	Outp...	Sort Type	Sort Order	Filter	Or
Jumlah		k	<input checked="" type="checkbox"/>				
Tanggal		k	<input checked="" type="checkbox"/>				

```

SELECT o.ID_Organisasi, o>NamaOrganisasi, k.Jenis, k.Jumlah, k.Tanggal
FROM
    Organisasi AS o INNER JOIN
    Keuangan AS k ON o.ID_Organisasi = k.ID_Organisasi

```

ID_Organisasi	NamaOrganisasi	Jenis	Jumlah	Tanggal
1	KMTI	Pemasukan	5000000.00	5/14/2025 12:3...
14	KMTE	Pemasukan	5000000.00	5/14/2025 3:42...

1 of 3 Cell is Read Only.

Execute Query OK Cancel

- g. Setelah muncul query,lalu pilih next.

TableAdapter Configuration Wizard

Enter a SQL Statement

The TableAdapter uses the data returned by this statement to fill its DataTable.

Type your SQL statement or use the Query Builder to construct it. What data should be loaded into the table?

What data should be loaded into the table?

```
SELECT o.ID_Organisasi, o>NamaOrganisasi, k.Jenis, k.Jumlah, k.Tanggal
FROM Organisasi AS o INNER JOIN
Keuangan AS k ON o.ID_Organisasi = k.ID_Organisasi
```

Advanced Options... Query Builder...

< Previous Next > Finish Cancel

h. Pilih next lagi

TableAdapter Configuration Wizard

Choose Methods to Generate

The TableAdapter methods load and save data between your application and the database.

Which methods do you want to add to the TableAdapter?

☒ **Fill a DataTable**

Creates a method that takes a DataTable or DataSet as a parameter and executes the SQL statement or SELECT stored procedure entered on the previous page.

Method name: Fill

☒ **Return a DataTable**

Creates a method that returns a new DataTable filled with the results of the SQL statement or SELECT stored procedure entered on the previous page.

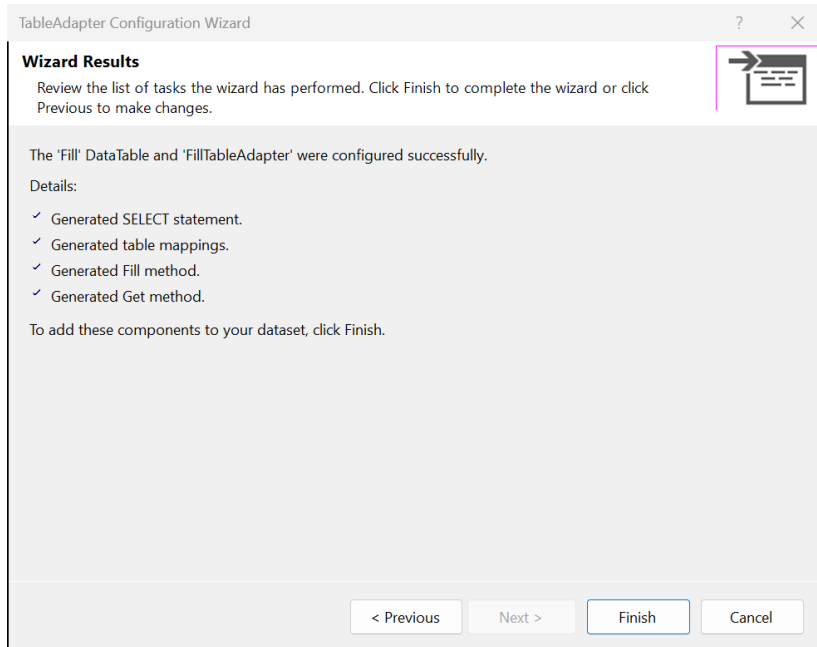
Method name: GetData

☐ **Create methods to send updates directly to the database (GenerateDBDirectMethods)**

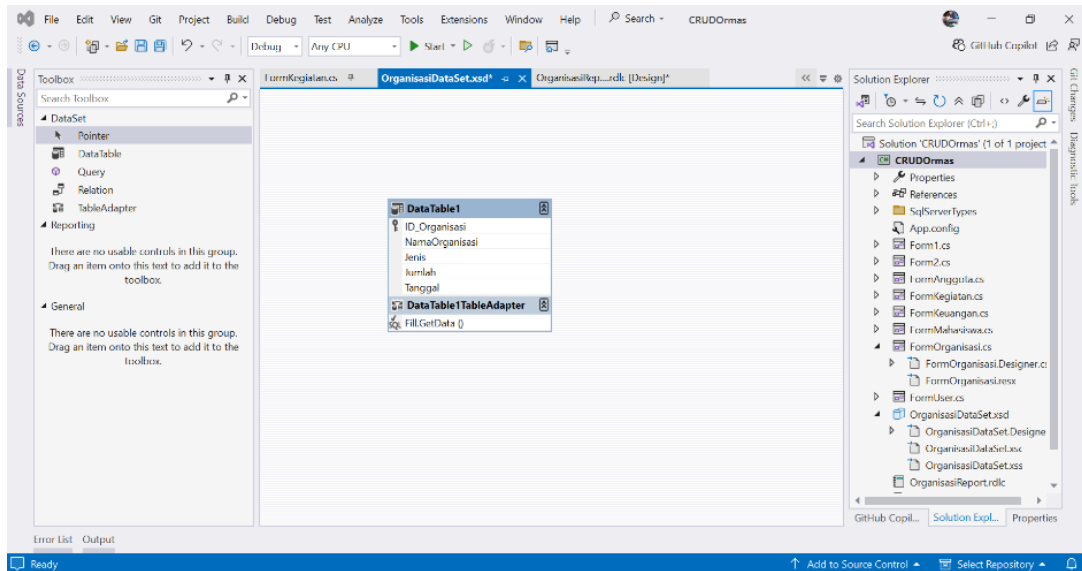
Creates Insert, Update, and Delete methods that can be called to send individual row changes directly to the database.

< Previous Next > Finish Cancel

i. Jika sudah sukses semua, pilih finish

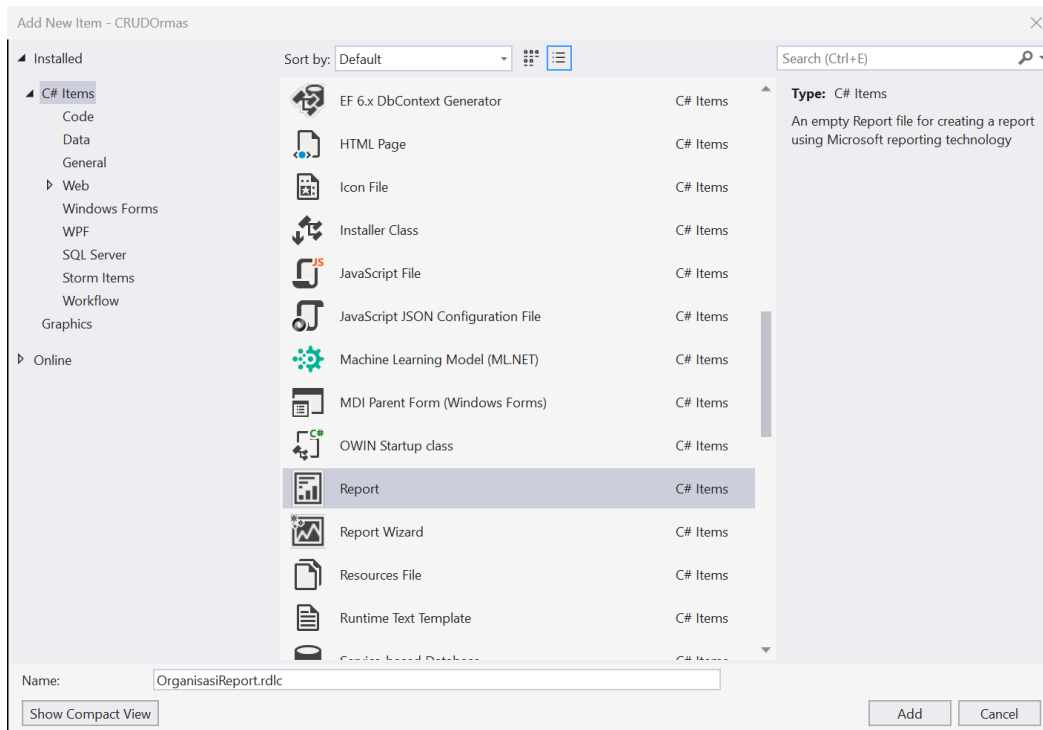


j. Pada tab desainer akan muncul tampilan sebagai berikut:



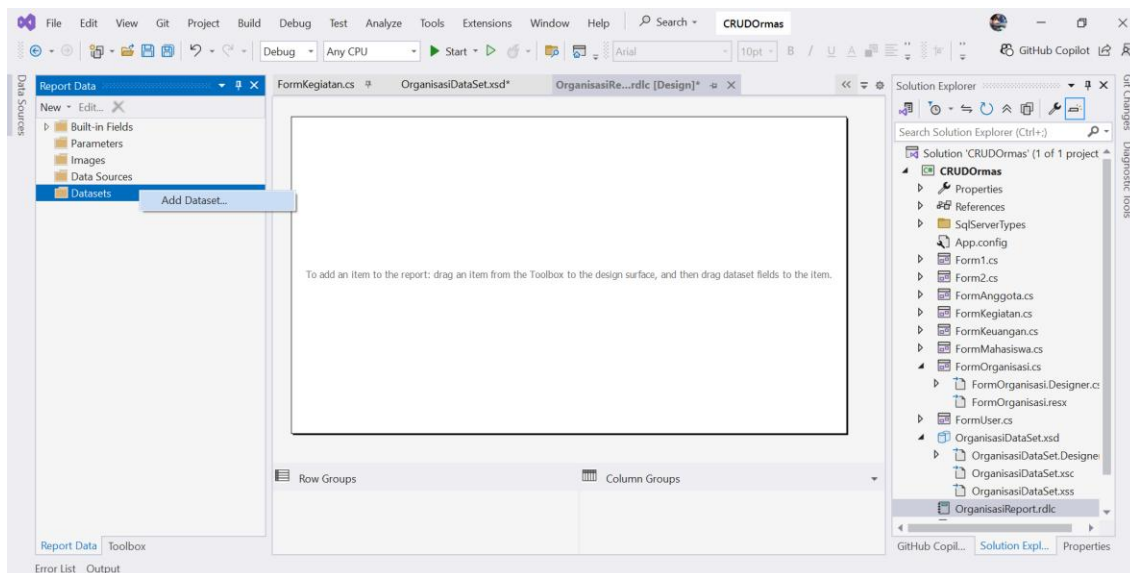
5. Buat Laporan (RDLC)

- Klik kanan pada proyek Anda di Solution Explorer.
- Pilih Add > New Item > Pilih Report dan beri nama, misalnya OrganisasiReport.rdlc.

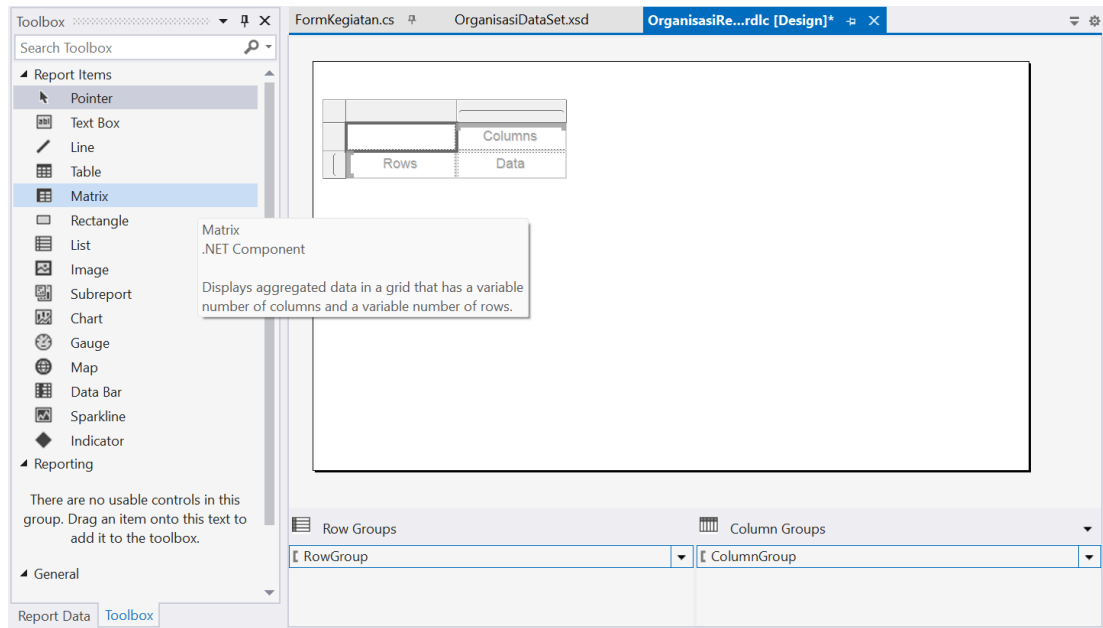


6. Menambahkan DataSet ke Report

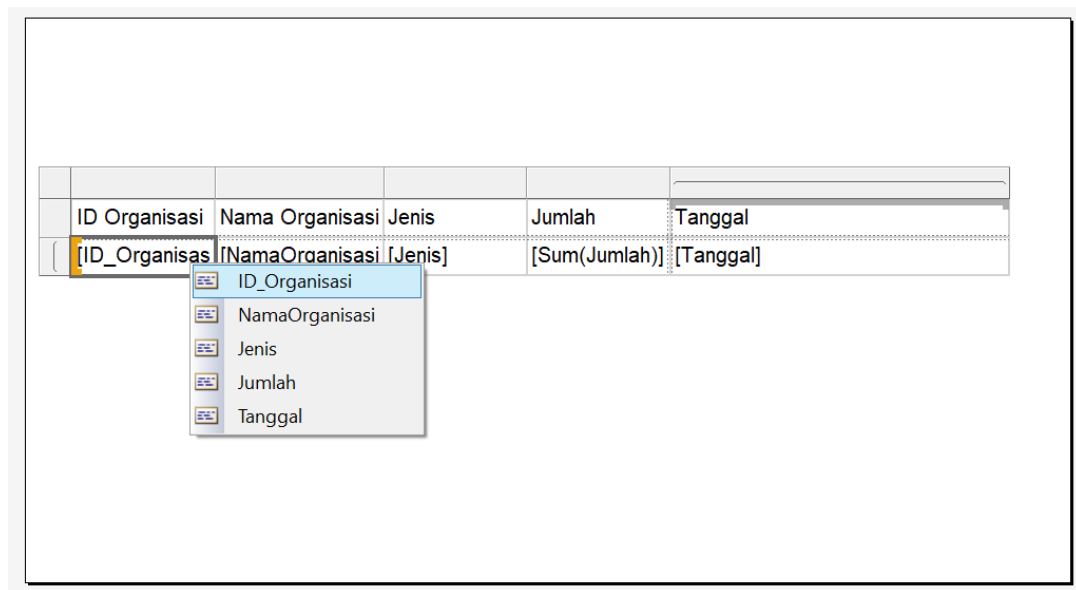
- Buka OrganisasiReport.rdlc
- Setelah file .rdlc terbuka, buka tab Report Data di sebelah kiri.
- Klik kanan pada Datasets dan pilih Add Dataset.



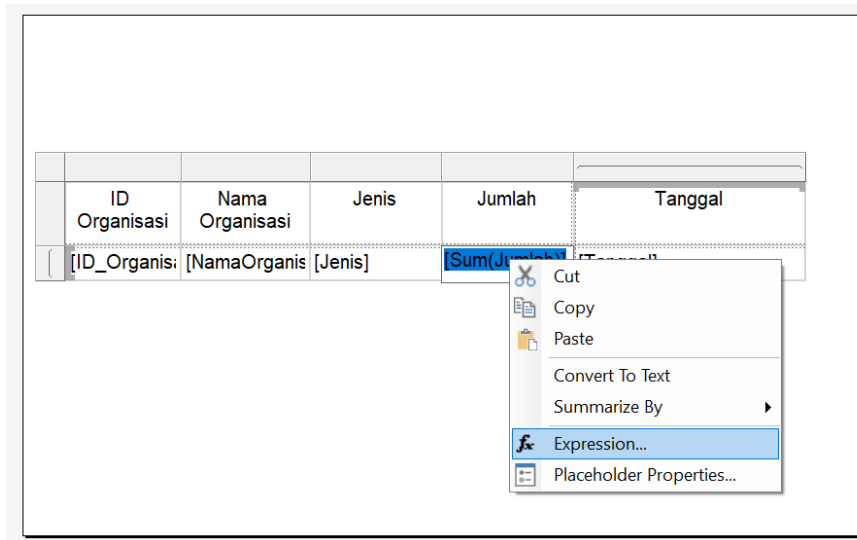
- Pilih Dataset yang sudah Anda buat sebelumnya (misalnya OrganisasiDataSet).



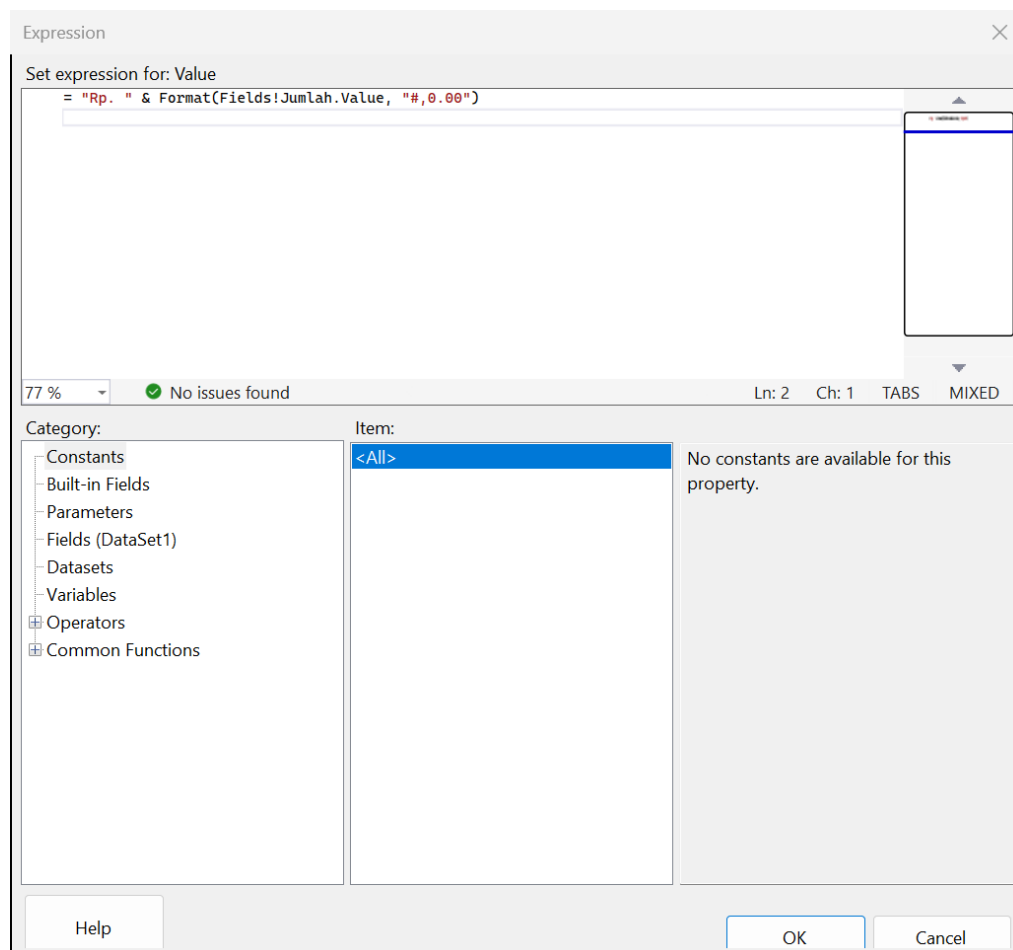
- Pada baris pertama (Header Row), tambahkan Judul sesuai dengan nama kolom anda lalu tautkan masing-masing kolom.



- Jika ingin kolom jumlah berubah menjadi format rupiah, lakukan hal berikut :



- Pada Expression Editor, Gunakan fungsi FormatCurrency untuk mengubah angka menjadi format mata uang.



8. Menghubungkan Data ke ReportViewer

Setelah desain laporan selesai, langkah berikutnya adalah menghubungkan laporan dengan ReportViewer pada form Anda.

9. Code untuk Mengisi ReportViewer

Berikut adalah kode lengkap untuk menghubungkan ReportViewer dengan Dataset dan menampilkan data di dalam laporan:

```
using System;
using System.Data;
using System.Data.SqlClient;
using System.Windows.Forms;
using Microsoft.Reporting.WinForms;

namespace CRUDOrmas
{
    4 references
    public partial class FormKegiatan : Form
    {
        1 reference
        public FormKegiatan()
        {
            InitializeComponent();
        }

        1 reference
        private void FormKegiatan_Load(object sender, EventArgs e)
        {
            // Setup ReportViewer data
            SetupReportViewer();
            // Refresh report to display data
            this.reportViewer1.RefreshReport();
        }

        private void SetupReportViewer()
        {
            // Connection string to your database
            string connectionString = "Data Source=DESKTOP-RAM2OFI\\APRILIYA;Initial Catalog=OrganisasiMahasiswa;Integrated Security=True";

            // SQL query to retrieve the required data from the database
            string query = @"
                SELECT
                    o.ID_Organisasi,
                    o>NamaOrganisasi,
                    k.Jenis,
                    k.Jumlah,
                    k.Tanggal
                FROM
                    Organisasi o
                JOIN
                    Keuangan k ON o.ID_Organisasi = k.ID_Organisasi";

            // Create a DataTable to store the data
            DataTable dt = new DataTable();

            // Use SqlDataAdapter to fill the DataTable with data from the database
            using (SqlConnection conn = new SqlConnection(connectionString))
            {
                SqlDataAdapter da = new SqlDataAdapter(query, conn);
                da.Fill(dt);
            }
        }
    }
}
```

```

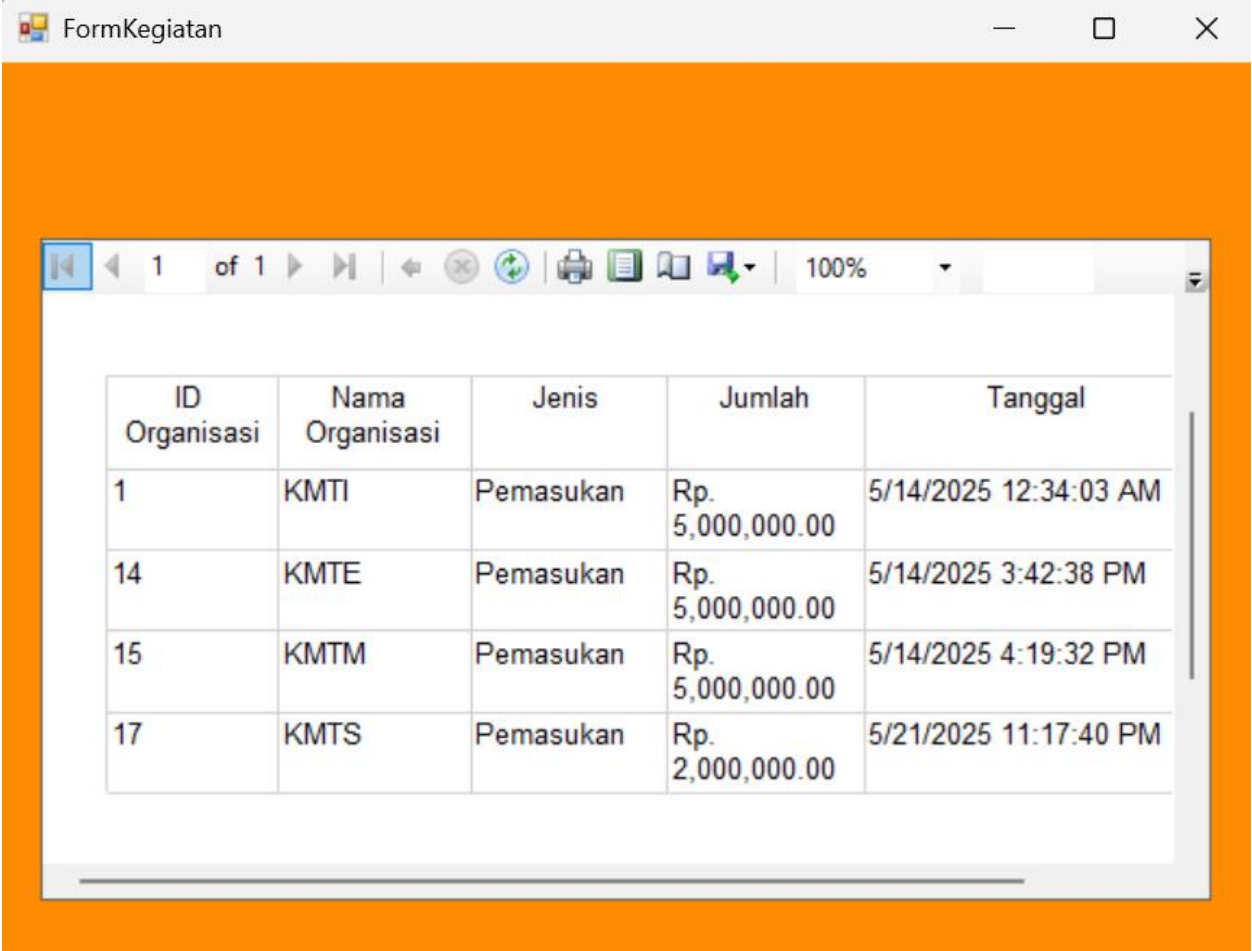
    // Create a ReportDataSource
    ReportDataSource rds = new ReportDataSource("DataSet1", dt); // Make sure "DataSet1" matches your RDLC dataset name

    // Clear any existing data sources and add the new data source
    reportViewer1.LocalReport.DataSources.Clear();
    reportViewer1.LocalReport.DataSources.Add(rds);

    // Set the path to the report (.rdlc file)
    // Change this to the actual path of your RDLC file
    reportViewer1.LocalReport.ReportPath = @"C:\Users\Lenovo\source\repos\CRUDOrmas\CRUDOrmas\OrganisasiReport.rdlc";
    // Refresh the ReportViewer to show the updated report
    reportViewer1.RefreshReport();
}
}
}

```

10. hasil report ketika dijalankan



ID Organisasi	Nama Organisasi	Jenis	Jumlah	Tanggal
1	KMTI	Pemasukan	Rp. 5,000,000.00	5/14/2025 12:34:03 AM
14	KMTE	Pemasukan	Rp. 5,000,000.00	5/14/2025 3:42:38 PM
15	KMTM	Pemasukan	Rp. 5,000,000.00	5/14/2025 4:19:32 PM
17	KMTS	Pemasukan	Rp. 2,000,000.00	5/21/2025 11:17:40 PM

Penjelasan:

Penjelasan untuk Code FormKegiatan:

Berikut adalah penjelasan tentang fungsi-fungsi utama dalam kode FormKegiatan yang mengimplementasikan ReportViewer untuk menampilkan laporan dari database:

1. Deklarasi dan Inisialisasi Form

```
public FormKegiatan()
{
    InitializeComponent();
}
```

- Konstruktor FormKegiatan digunakan untuk menginisialisasi komponen-komponen form, seperti ReportViewer dan kontrol lainnya.
- InitializeComponent() adalah metode yang otomatis dihasilkan oleh Visual Studio untuk menginisialisasi komponen form.

2. FormKegiatan_Load

```
private void FormKegiatan_Load(object sender, EventArgs e)
{
    // Setup ReportViewer data
    SetupReportViewer();
    // Refresh report to display data
    this.reportViewer1.RefreshReport();
}
```

- FormKegiatan_Load adalah event handler yang dijalankan saat form Kegiatan pertama kali dimuat.
- Di sini, metode SetupReportViewer() dipanggil untuk menyiapkan ReportViewer dengan data dari database dan kemudian me-refresh ReportViewer dengan this.reportViewer1.RefreshReport(), yang akan menampilkan laporan yang telah dipersiapkan.

3. SetupReportViewer

```
private void SetupReportViewer()
{
    // Connection string to your database
    string connectionString = "Data Source=DESKTOP-RAM20FI\\APRILIYA;Initial Catalog=OrganisasiMahasiswa;Integrated Security=True";
}
```

- SetupReportViewer() adalah metode utama yang menghubungkan aplikasi dengan SQL Server dan mempersiapkan data yang akan ditampilkan di ReportViewer.
- connectionString berisi informasi tentang cara aplikasi terhubung ke database OrganisasiMahasiswa, seperti alamat server, nama database, dan informasi autentikasi.


```
// SQL query to retrieve the required data from the database
string query = @"
    SELECT
        o.ID_Organisasi,
        o>NamaOrganisasi,
        k.Jenis,
        k.Jumlah,
        k.Tanggal
    FROM
        Organisasi o
    JOIN
        Keuangan k ON o.ID_Organisasi = k.ID_Organisasi";
```

- query adalah query SQL yang mengambil data dari dua tabel (Organisasi dan Keuangan) yang di-join berdasarkan kolom ID_Organisasi.
- Data yang diambil adalah ID_Organisasi, NamaOrganisasi, Jenis (dari tabel Keuangan), Jumlah, dan Tanggal.

```
// Create a DataTable to store the data
DataTable dt = new DataTable();
```

- DataTable dt adalah tempat untuk menyimpan data hasil query yang akan digunakan oleh ReportViewer.

```
// Use SqlDataAdapter to fill the DataTable with data from the database
using (SqlConnection conn = new SqlConnection(connectionString))
{
    SqlDataAdapter da = new SqlDataAdapter(query, conn);
    da.Fill(dt);
}
```

- SqlDataAdapter digunakan untuk menjalankan query SQL dan mengisi DataTable dengan data yang diambil dari database.
- conn adalah koneksi ke database menggunakan connectionString.

```
// Create a ReportDataSource
ReportDataSource rds = new ReportDataSource("DataSet1", dt); // Make sure "DataSet1" matches your RDLC dataset name
```

- ReportDataSource adalah objek yang menghubungkan ReportViewer dengan DataTable yang berisi data.
- "DataSet1" adalah nama dataset yang digunakan di dalam RDLC. Pastikan ini sesuai dengan nama dataset yang Anda tentukan dalam laporan RDLC.

```
// Clear any existing data sources and add the new data source
reportViewer1.LocalReport.DataSources.Clear();
reportViewer1.LocalReport.DataSources.Add(rds);
```

- Sebelum menambahkan data baru, kita menghapus sumber data yang lama dengan DataSources.Clear().

- Kemudian, kita menambahkan ReportDataSource yang baru, yang berisi data yang telah diisi di DataTable.

```
// Set the path to the report (.rdlc file)
// Change this to the actual path of your RDLC file
reportViewer1.LocalReport.ReportPath = @"C:\Users\Lenovo\source\repos\CRUDOrmas\CRUDOrmas\OrganisasiReport.rdlc";
```

- ReportPath menentukan lokasi dari file RDLC yang digunakan untuk menampilkan laporan. Pastikan bahwa path ini benar dan file RDLC Anda ada di lokasi yang tepat.

```
// Refresh the ReportViewer to show the updated report
reportViewer1.RefreshReport();
```

- RefreshReport() memaksa ReportViewer untuk memuat ulang laporan dengan data yang baru diisi.