Report

Report yg kita hasilkan merupakan sebuah file PDF yg dihasilkan oleh bagian BackEnd, jadi kita tidak mempunyai control di bagian FrontEnd Blazor, melainkan kita hanya mempunyai API Controller yg menghasilkan file export berupa PDF yang dapat dipanggil dari Blazor maupun dari mobile.

# Library

Untuk library kita memakai 2 Library

1. R\_ReportFastReportBack
   1. Berisi library untuk kebutuhan reporting yang terdiri dari beberapa class
      1. R\_ReportFastReportBackClass
         1. Merupakan Class yg dipakai di bagian controller untuk menghasilkan report
      2. R\_ReportUtility
         1. Merupakan class Utility helper untuk Report
   2. Dependency Package
      1. FastReport.Core
2. R\_Cache
   1. Berisi Library yg dipakai untuk melakukan penyimpanan sementara yang membantu kita dalam mekanisme untuk menghasilkan sebuah report.
   2. Dependency Package
      1. Microsoft.Extensions.Caching.StackExchangeRedis
   3. Dependency Software
      1. Redis Server

# Design Report

Secara design report, sebenarnya banyak type datasource yang dapat kita pakai, tetapi kita memilih menggunakan DTO sebagai penampung data karena pada sebuah DTO dapat mempunyai beberapa entity baik berupa one to one relation maupun berupa Parent Child relation.

## Persyaratan Design

Karena kita menggunakan DTO sebagai data source, maka pada waktu kita melakukan design maka kita harus mempersiapkan 2 project

### Common Project

Merupakan project C# dengan framework standard 2.1 yang akan berisi DTO dan project ini dipisah berdasarkan nama program report, misalkan LMR00100Common

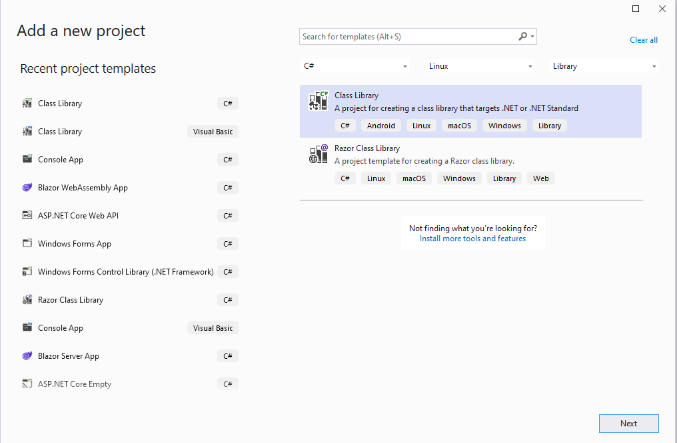
DTO tsb ada di bagian Common untuk mempersiapkan kalau sewaktu2 kita mau mengembangkan Control di bagian FrontEnd, saat ini hanya dipakai di bagian backend dan di bagian Design

### Design Project

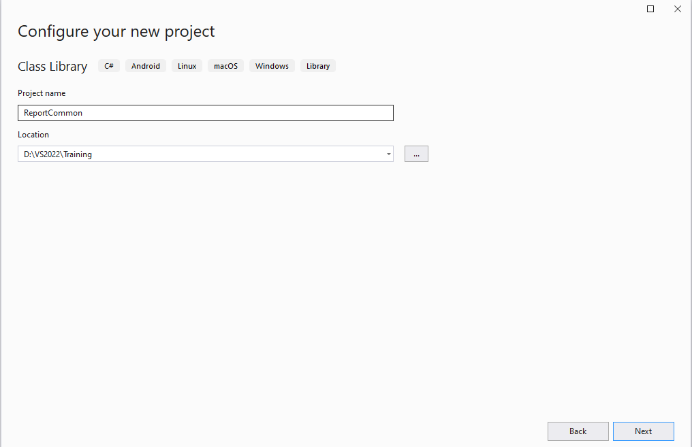
Untuk melakukan men-design sebuah report dengan datasoure DTO, maka kita perlu membuat sebuah winform project yang nantinya akan berisi tombol2 untuk menjalankan design report dengan melakukan inject DTO dari Common Project, sehingga report yg akan kita design dapat mempunyai datasource DTO yg kita sediakan saat menjalankan design report.

# Pembuatan Common Project

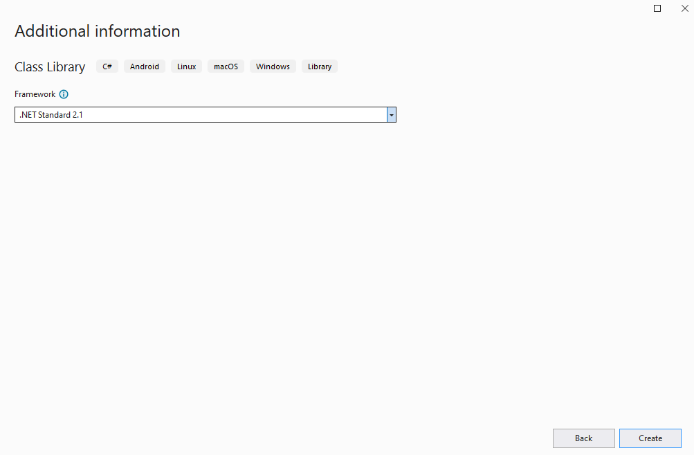
Kita membuat common project dengan menggunakan framework standard 2.1, misalkan dalam hal ini kita buat project ReportCommon dengan menggunakan Class Library Project



Dengan nama ReportCommon

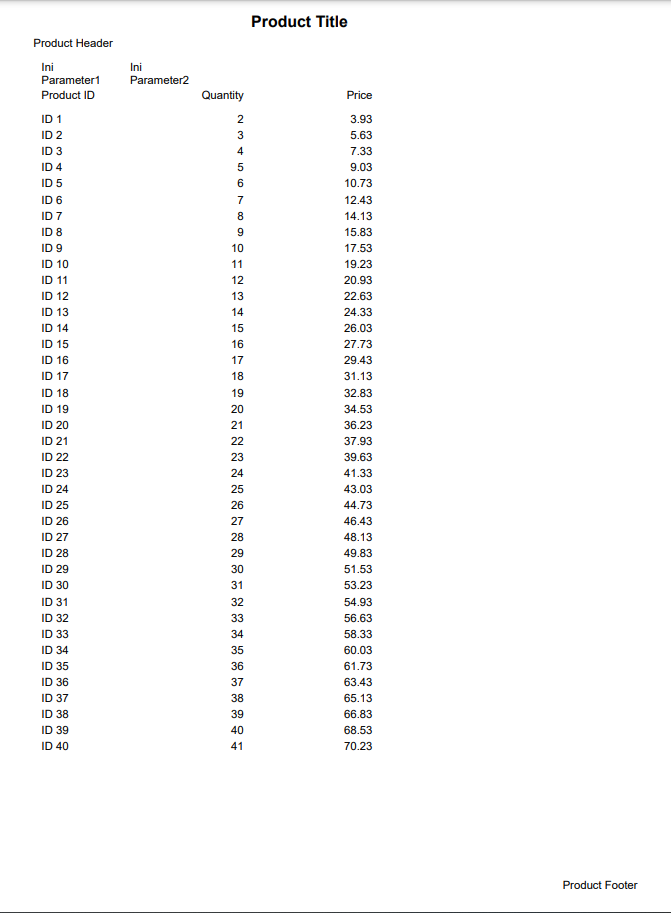


Menggunakan framework Standard 2.1



## Pembuatan Report Single Entity

Untuk membuat sebuah report yg berisi single entity seperti gambar di bawah ini



Data

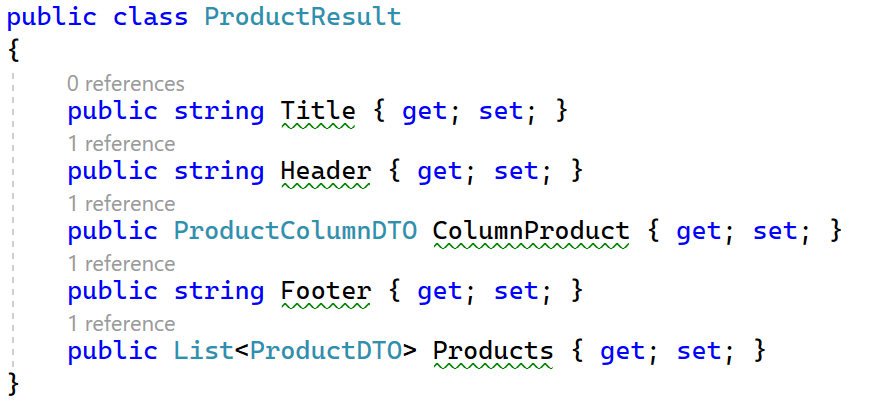
Footer

ColumnDTO

Parameter

Header

Title



Data

Footer

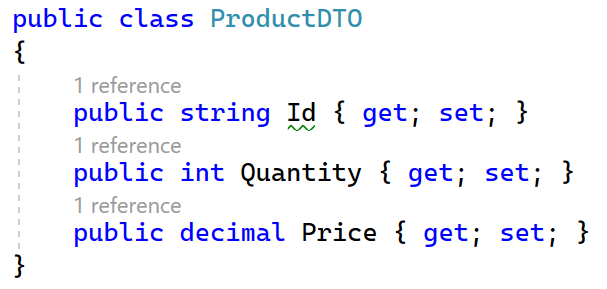
ColumnDTO

Title

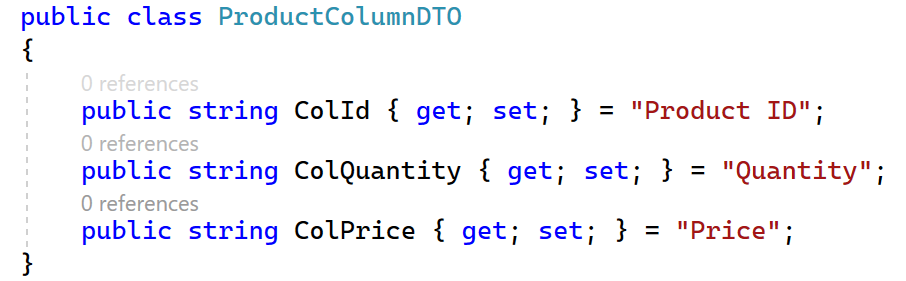
Header

Dalam contoh diatas DTO kita adalah ProductResult yg berisi beberapa entity dan entity2 tsb dipakai supaya semua entity ini tidak diisi di bagian report tapi dapat kita assign saat pengisian data, misalkan dengan menggunakan culture, maka kita dapat mengganti title, header, footer, column sesuai dengan culture nya.

Data kita beri nama Products yg berupa List dari class ProductDTO



Column kita beri nama ColumProduct yg berupa class ProductColumnDTO

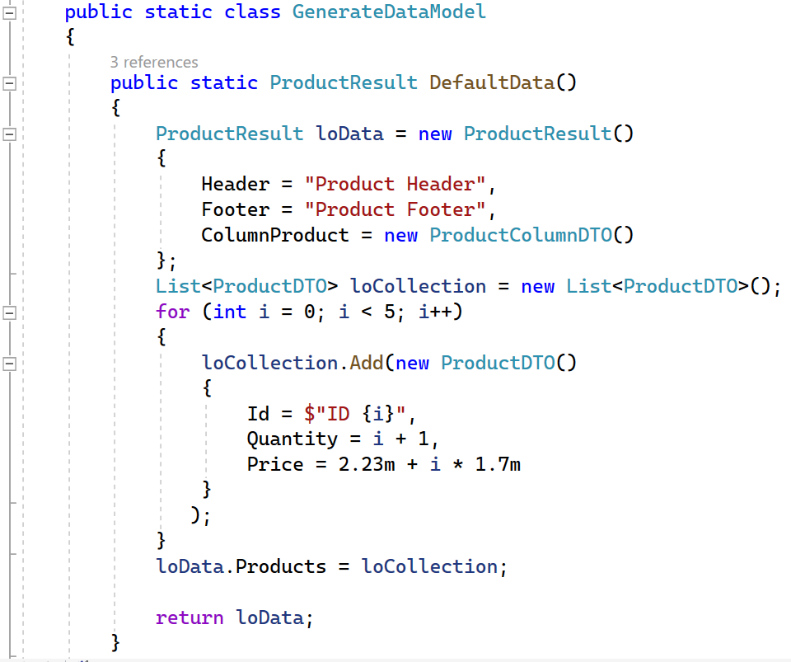


Untuk melakukan design sebuah report, terkadang kita memerlukan sekumpulan data yang nanti nya dapat kita test behavior nya saat melakukan design, untuk hal ini kita dapat membuat suatu static class dan static method yg dapat kita panggil untuk mengisi default data sebelum kita menjalankan report design.

## Pengisian Default Data

Untuk mengisi Default Data, maka kita dapat membuat sebuah static class dan static method yang menghasilkan DTO yg akan kita pakai, dalam hal ini menghasilkan ProductResult

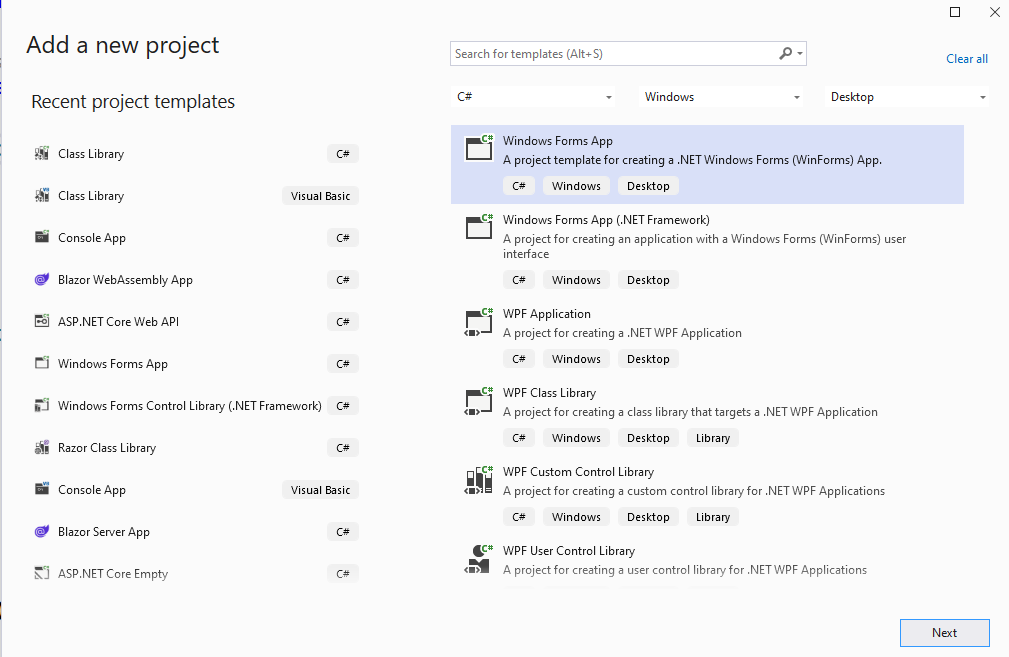
Contoh dibawah ini kita membuat sebuah **static class GenerateDataModel** dengan sebuah **static method DefaultData** yg **menghasilkan ProductResult** Object



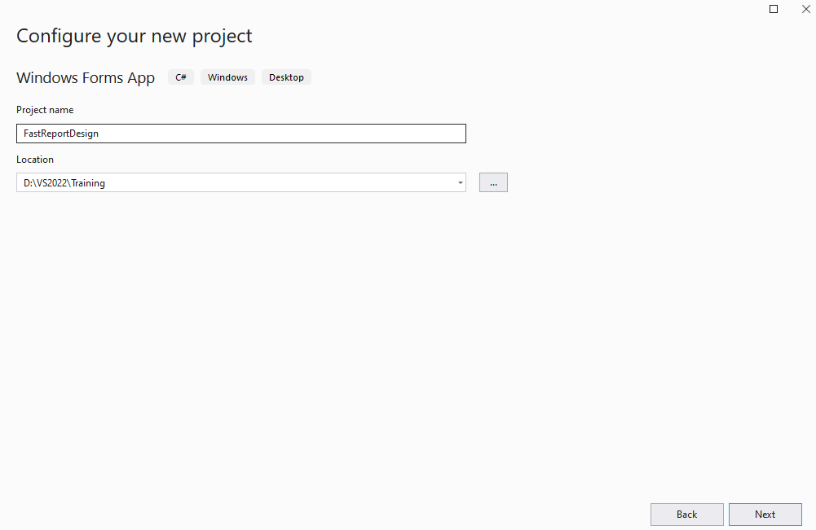
# Pembuatan Design Project

Supaya kita dapat melakukan injection DTO kita sebelum kita membuat design report, maka kita perlu membuat sebuah project WinForm, melalui project ini kita dapat membuat beberapa design report yang kita pisahkan dengan tombol, jadi kita hanya membuat 1 project designer untuk melakukan running design untuk semua program report kita.

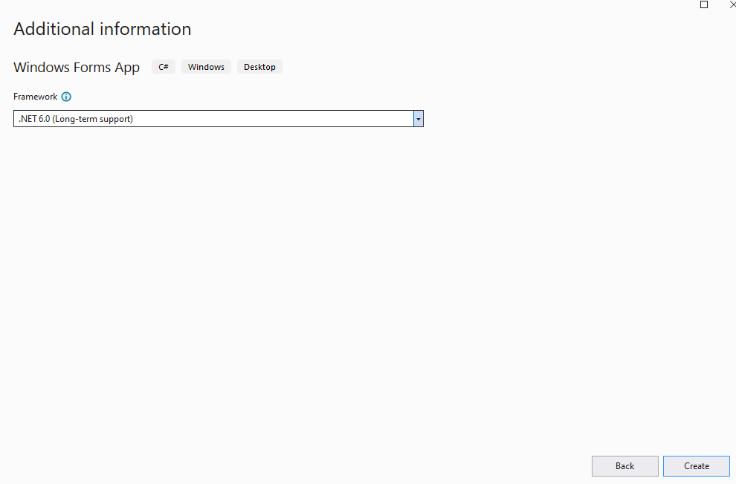
Kita membuat design project dengan menggunakan project Desktop WinForm



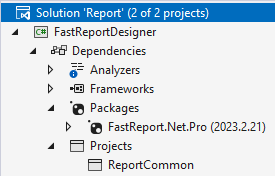
Dengan nama FastReportDesign



Dengan framework NET 6.0

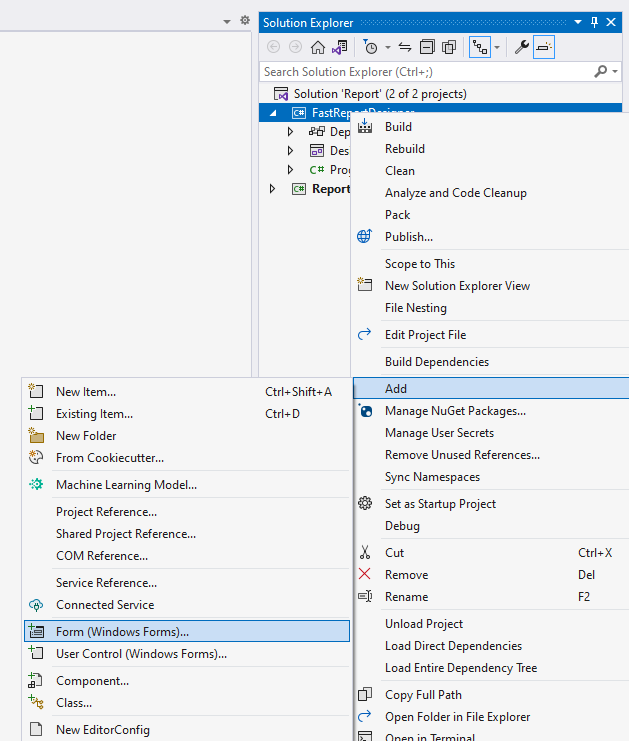


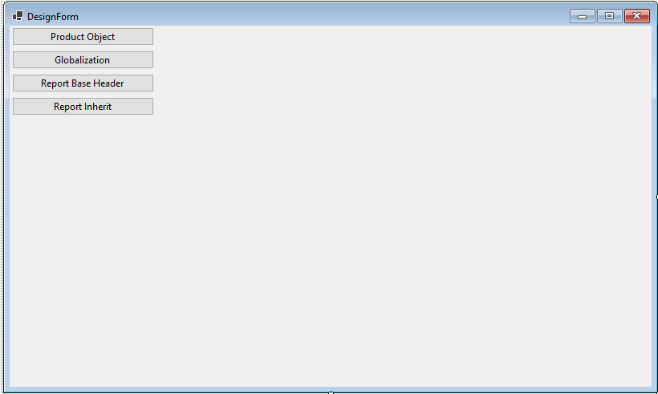
Untuk menjalankan design report, kita memerlukan package FastReport.Net.Pro yg ada di Realta Nuget, juga kita memerlukan referensi Common Project (masing2 report program).



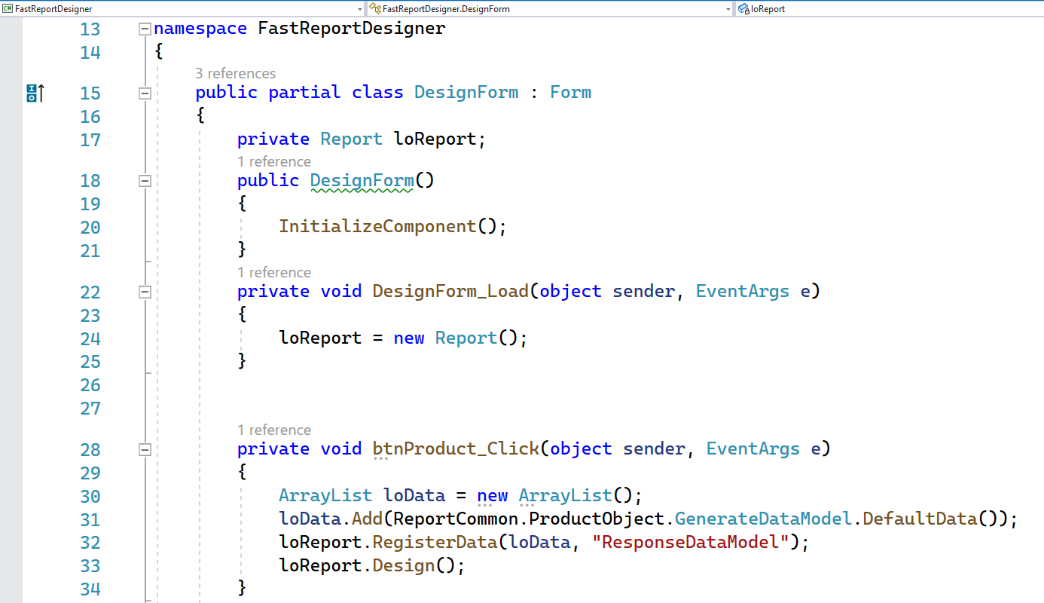
Dalam contoh kita ini, kita gabungkan beberapa project common report menjadi 1 project dan hanya dibedakan oleh namespace.

Tetapi nanti secara real nya kita pisahkan Common Project nya per report untuk kebutuhan pemisahan source supaya dapat ditangani oleh masing2 developer.

Tambahkan Windows Form

Report akan dipisah2kan oleh tombol

Dalam contoh ini kita akan membuat report untuk DTO ProductResult

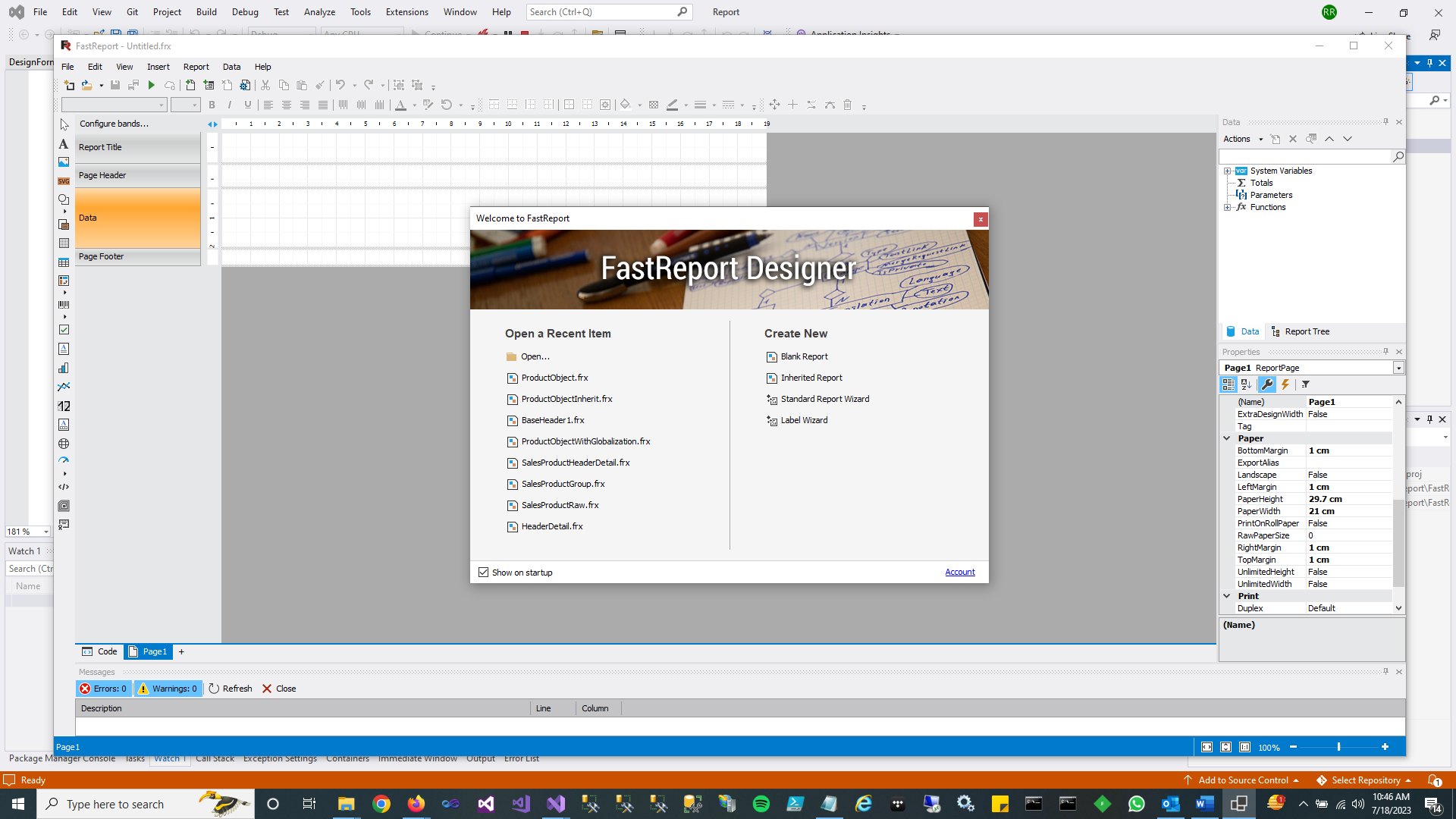


## Langkah2 menjalankan report designer

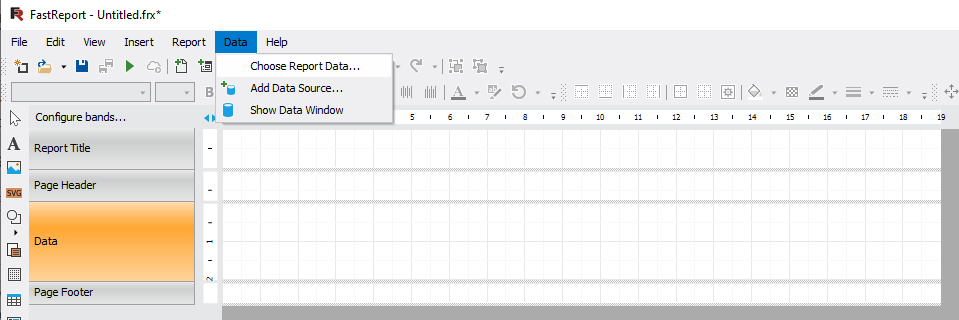
1. Buat variable class loReport yang merupakan type FastReport.Report
2. Saat Form Load, maka kita meng-instantiate loReport
3. Untuk method Product\_Click
   1. Siapkan variable loData dengan type ArrayList
   2. Tambahkan loData dengan ProductResult yg didapatkan dari static method DefaultData yang ada di dalam static class GenerateDataModel
   3. Register ArrayList dengan suatu nama, dalam hal ini **“ResponseDataModel”**, ingat nama dari register ini, nantinya akan muncul **sebagai nama data di report**
   4. Jalankan method Design pada variable loReport.

## Langkah2 Membuat Report di dalam report designer

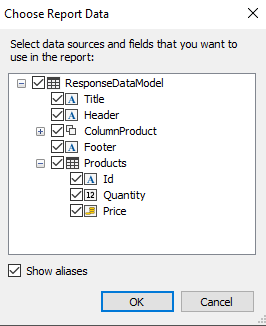
Setelah kita meng-click tombol untuk menjalankan report designer, maka akan tampil Report designer seperti gambar dibawah ini.



Karena ini report baru, maka kita pilih bagian Create New dan pilih Blank Report.



Pilih bagian menu Data dan bagian item Choose Report Data

Buka semua nya dan check semua checkbox dan click OK, hal ini kita lakukan karena data yg kita butuhkan ada di semua property dari DTO tsb

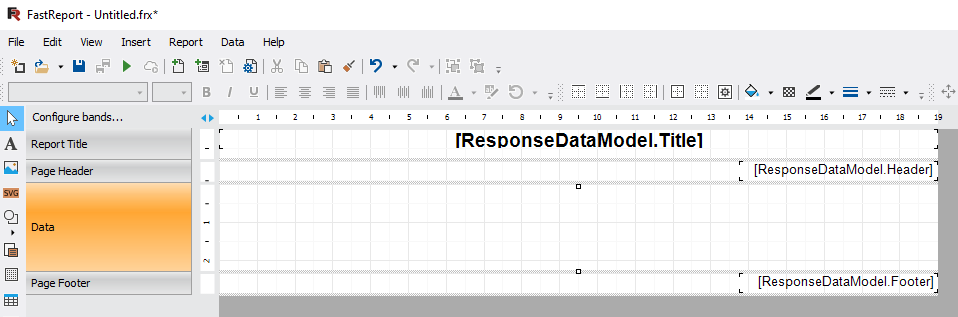
Maka pada bagian data di sebelah kanan akan muncul structure DTO kita dengan nama yg kita berikan waktu kita jalankan method RegisterData di [langkah2 menjalankan report designer](#_Langkah2_menjalankan_report)

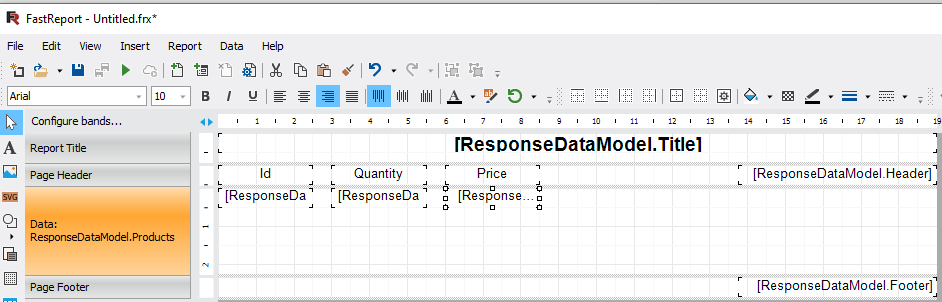
Kalau kita lihat empty report secara default sudah mempunyai beberapa band

1. Report Title
   1. Hanya muncul 1x dalam report
2. Page Header
   1. Muncul di atas di tiap halaman
3. Data
   1. Muncul berulang terhadap data yg di-assign
4. Page Footer
   1. Muncul dibawah di tiap halaman

Untuk Band, kita dapat membaca pada PDF FRNetUserManual bagian Bands in Designer hal 13.

Kita dapat melakukan drag drop semua property dari data ke tempatnya masing2

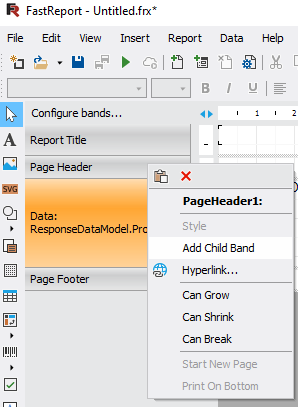
Contoh property yg di drag drop selain data

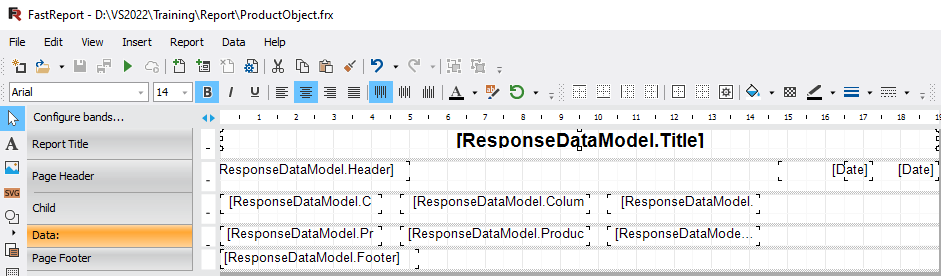


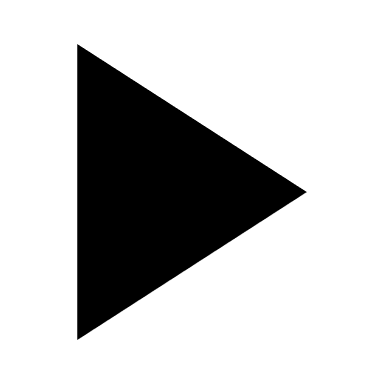
Contoh Data di drag drop, ada 2 hal yg harus diperhatikan

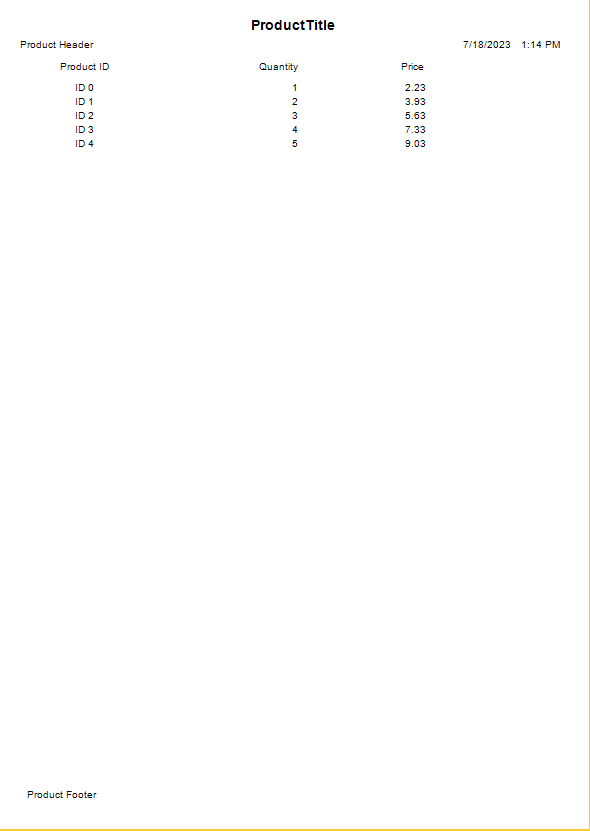
1. Muncul nama property di bagian page header
   1. Property ini akan kita hapus dan akan kita ganti dengan property ColumnProduct
2. Muncul nama property dan namespace di bagian band data
   1. Property pada band data ini sangat penting untuk menandakan kapan pengulangan data akan muncul pada report

Supaya kita dapat meletakkan ColumnProduct pada band tersendiri, maka kita dapat menambahkan Child Band pada Page Header Band

Cara menambahkan Child Band, dengan memunculkan menu popup pada Band saat kita click kanan atau kalau kita mau mengatur keseluruhan Band dapat kita atur dengan menu Configure Band y gada diatas Report Title

Setelah penempatan semua property pada report designer

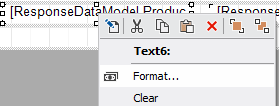
Kalau kita preview (click ) maka hasilnya akan terlihat spt di bawah ini.



### Format dan Font

Kalau kita lihat untuk price dan quantity ditampilkan dalam bentuk yg berbeda, hal ini dikarenakan ada pengaturan default dengan format currency pada decimal type dan number pada integer type.

Cara melakukan format per masing2 object munculkan menu popup dengan click kanan pada object yg akan di format

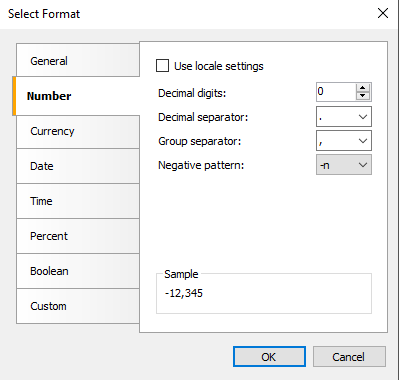


Pengaturan format untuk tiap2 data.

#### Format Data Integer

Untuk Data integer, maka kita lakukan

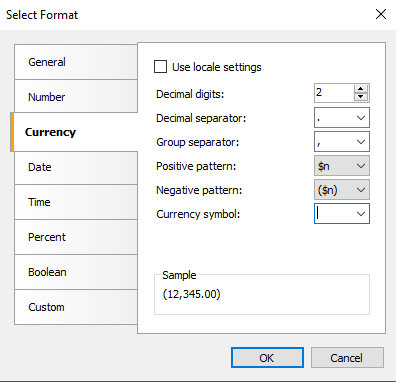
1. Uncheck Use locale settings
2. Decimal digits menjadi 0



#### Format Data Decimal

Untuk data decimal

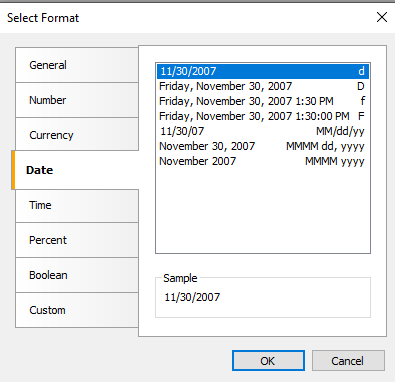
1. Uncheck Use locale settings
2. Biarkan decimal digit
3. Hapus Currency symbol



#### Format Data Date

Untuk data date

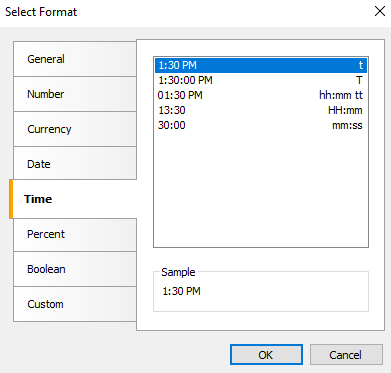
1. Pilih Date



#### Format data time

Untuk data time

1. Pilih Time

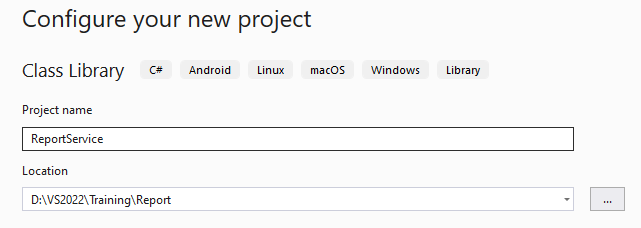


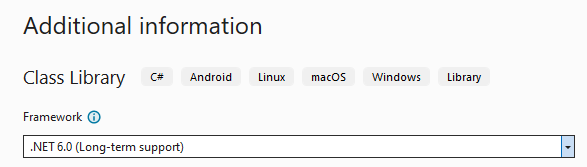
# Pembuatan Service Project

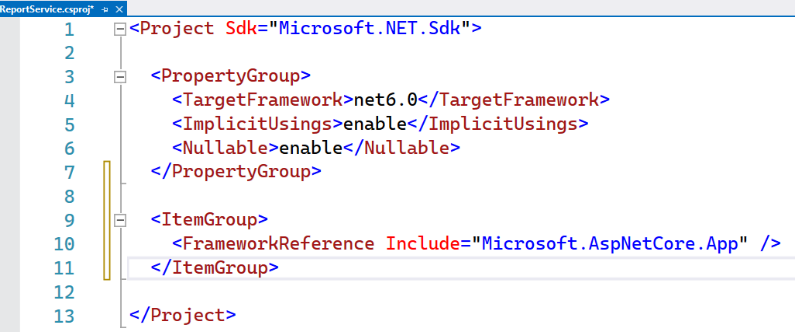
Sesuai dengan standard pembuatan Service Project

A screenshot of a computer

Description automatically generatedPilih type Class Library

Berikan nama untuk project dalam hal ini ReportService

Framework Net 6.0

Edit project dan tambahkan <FrameworkReference Include="Microsoft.AspNetCore.App" />

A screenshot of a computer

Description automatically generated

## Reference Assembly

1. R\_ApiCommon
   1. Untuk Exception framework
2. R\_ApiBackEnd
   1. Untuk R\_BackGlobalVar
3. R\_Cache
   1. Untuk penyimpanan parameter ke temporary storage
4. R\_CommonFrontBackAPI
   1. Untuk menghasilkan Result GUID, hasil dari penyimpanan parameter
5. R\_ReportFastReportBack
   1. Reporting Class

## Reference Project

1. ReportCommon
   1. DTO
   2. Interface

## Pembuatan Controller

Buat Controller dengan nama ProductObjectController

Setting untuk anotasi Class

Inherits ControllerBase

Implements IProductObject

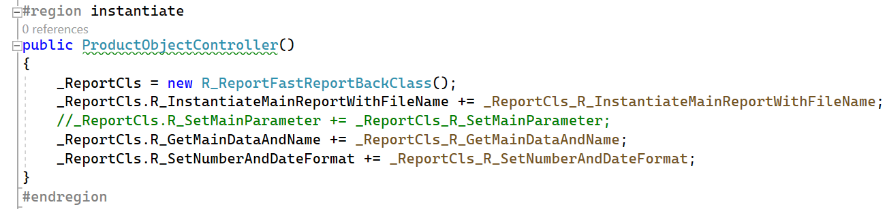
Variable Class \_ReportCls

Variable Class \_AllProductParameter

### Instantiate Controller

Saat melakukan Instantiate Controller, maka kita dapat melakukan

1. Meng-instantiate variable \_ReportCls
2. Menambahkan event handler report

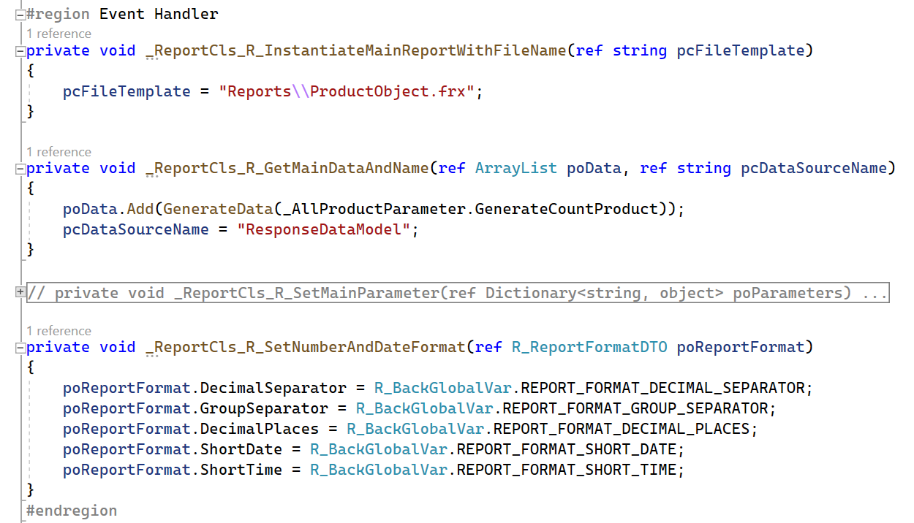


### Event Handler Report

Ada 4 macam Event Handler

Untuk Instantiate tergantung dari permintaan saat menjalankan method R\_GetStreamReport

1. R\_InstantiateMainReportWithFileName
   1. Parameter string yg kita isi dengan lokasi dari file template report
2. R\_InstantiateMainReportWithByte
   1. Parameter Byte() yg kita isi dengan file template report yg sudah menjadi byte array
3. R\_SetMainParameter
   1. Parameter Dictionary<string,object> untuk mengisi report yg mempunyai parameter
4. R\_GetMainDataAndName
   1. Parameter ArrayList yg dipakai untuk memasukkan DTO dari report
   2. Parameter DataSourceName yg diisi dengan nama registrasi datasource saat pembuatan report
5. R\_SetNumberAndDateFormat
   1. Parameter ReportFormat yg diisi dengan backend global variable untuk urusan report



### Cara kerja Report pada API

Untuk Report di API, setiap mau mendapatkan report kita membagi nya menjadi 2 langkah

1. Post
   1. AllProductPost
   2. Merupakan method Post yg dipakai untuk mengirimkan parameter dari Body, method ini akan menyimpan parameter ke dalam repository sementara memakai Redis Service
   3. Sebelum menyimpan di Redis, maka parameter yg berupa object tsb akan diserialize terlebeih dahulu dengan bantuan R\_NetCoreUtility.R\_SerializeObjectToByte<T>
   4. Method ini akan mengembalikan R\_DownloadFileResultDTO yg berisi Guid yg akan kita masukkan sebagai parameter method pasangan yg bagian Get
   5. Repository Sementara ini akan menyimpan datasecara default hanya dalam 1 menit
2. Get
   1. AllProductGet
   2. Pada method ini akan mengambil parameter yg diperlukan dari temporary storage Redis berdasarkan parameter Guid
   3. Lalu parameter tsb akan di-deserialize dengan bantuan R\_NetCoreUtility.R\_DeserializeObjectFromByte<T> dan akan di-assign ke variable class untuk parameter
   4. Variable class untuk parameter tsb nantinya akan dipakai pada saat event R\_GetMainDataAndName
   5. Pada method ini, kita akan memanggil method R\_GetStreamReport y gada pada class report.



# Redis

Redis service dipakai sebagai temporary storage untuk menyimpan parameter report

## Menjalankan Redis

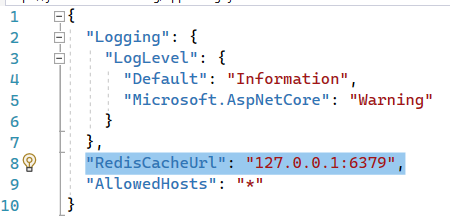
Karena kita memakai windows dalam sample ini, untuk menjalankan Redis

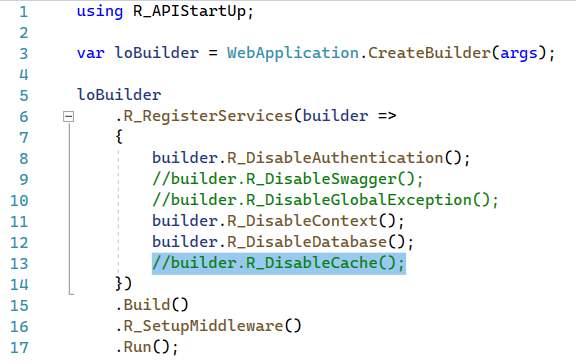
1. unzip Redis-x64-3.0.504.zip
2. Run redis-server.exe

## Setting Kebutuhan Redis

Untuk kebutuhan Redis ini kita melakukan 2 hal **pada API Project**

1. Setting di appsettings.json
2. Setting di program.cs

Setting pada appsettings.json

Untuk Cache disable ditutup

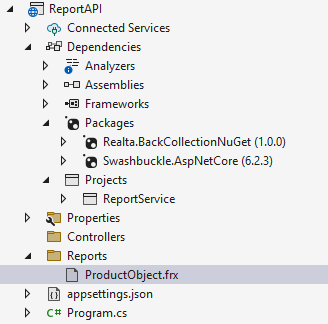
Dalam contoh ini kita tidak memakai context, nantinya kalau terhubung dengan front, kita harus memakai context karena semua setting report akan di supply dari front melalui context

# Penggabungan Project API

Untuk menggabungkan project API

1. Assembly memakai semua Library Back
2. Reference Project ReportService
   1. Saat real nantinya akan berupa reference assembly
3. Buat Directory Reports
   1. Masukkan hasil report designer dalam directory report
   2. Setting Copy Always

A screenshot of a computer

Description automatically generated

# Note untuk report diluar basic report

## Globalization

1. Penambahan Resources
2. Globalization harus nya di-Assign lewat R\_BackGlobalVar

## Inherit

1. Untuk Header
   1. Dibuatkan DTO dan DTOResult untuk Header
   2. Dibuatkan Report Header yg berisi DTOResult Header
      1. Untuk Page keseluruhan harus 2 pass setting
   3. Header hanya berisi Header Page
2. Untuk Report
   1. Dibuatkan DTO yg meng-inherit DTOResult Header
   2. Report harus meng-inherit Report Header
   3. Untuk Header jangan diubah posisi nya
   4. Untuk Data pastikan di band data untuk menggunakan data (dalam hal ini ProductObjectData) bukan dataheader

## ReportHeaderDetail

1. Didalam CategoryDTO ada property Products yg merupakan collection ProductDTO
2. Category ada di dalam band Data
   1. Product ada di dalam Band Child Data didalam Band Data Category
   2. Property Start New Page
   3. Property Keep Detail

## Flat/Raw Data

1. Untuk Group dan Matrix memerlukan record header detail yg flat, biasanya diproduksi dari SP
2. Untuk membuat Data flat menjadi Header Detail Object, maka kita dapat melakukan LINQ Group
3. Kita dapat membuat suatu variable class type untuk membuat beberapa method Get dalam sebuah controller
   1. Di Set di masing2 method Get
   2. Dipakai di bagian R\_InstantiateMainReportWithFileName untuk memisahkan template report
   3. Dipakai di bagian R\_InstantiateMainReportWithFileName untuk mendapatkan data yg sesuai