Utility

Untuk Library ada 2 utility class

1. R\_Utility
   1. Merupakan bagian assembly R\_APICommon yg ada di bagian Back
   2. Menggunakan .NET Core 6.0
   3. Merupakan Utility yg dipakai oleh .NET framework untuk meng-akomodasi Old Class yg dapat kita gunakan saat kita upgrade class yg lama dari tehnologi .Net Framewok ke .NET Core.
2. R\_NetCoreUtility
   1. Merupakan bagian assembly R\_CommonFrontBackAPI yg ada di bagian Common
   2. Menggunakan .NET Standard2.1
   3. Merupakan utility tambahan untuk .NET Core, dimana hanya meng-cover Serialization File dan Object karena untuk convertion collection kita dapat menggunakan LINQ

# R\_Utility

R\_Utility merupakan library yg dipakai sebagai kompatibilitas saat memakai class lama sebagai bisnis proses yg dipanggil oleh controller, Utility tsb adanya di bagian back dan mempunyai beberapa fungsi

## Object Converter

### Fungsi IList< T> R\_ConvertCollectionToCollection< F, T>( IList< F> poList)

1. Merupakan fungsi untuk mengubah sebuah collection dari Type data F menjadi sebuah collection dengan type data T
2. Konversi yg dilakukan adalah dengan nama field yg sama
3. Parameter
   1. poList
      1. Type IList<F>
4. Return
   1. Type IList<T>

### Fungsi T R\_ConvertObjectToObject<F,T>(F poPar)

1. Merupakan fungsi untuk melakukan konversi dari sebuah object type data F ke object dg type data T
2. Konversi yg dilakukan adalah dengan nama field yg sama
3. Parameter
   1. poPar
      1. Type F
4. Return
   1. Type T

### Fungsi T R\_MapObjectToObject<F,T>(F poPar, string pcTargetFormat)

1. Merupakan fungsi untuk melakukan konversi dari sebuah object type data F ke object dg type data T
2. Konversi yg dilakukan adalah dengan nama field yg berbeda dengan target tetapi perbedaan nama field nya berbeda secara keseluruhan dengan suatu custom format
3. Parameter
   1. poPar
      1. Type F
   2. pcTargetFormat
      1. Type String
      2. Custom Format dari field
      3. Misalkan “\_{0}” berarti nama field XXX akan menjadi \_XXX
4. Return
   1. Type T

## Table Converter

### Fungsi DataTable R\_ConvertTo<T>(IList<T> poList, bool plAcceptChanges = true, Dictionary<string, string> poMapping = null)

1. Merupakan fungsi yg melakukan konversi dari Collection dengan Type T ke sebuah datatable
2. Disediakan juga Mapping yg digunakan untuk melakukan mapping name field dari field object ke Table column name
3. Parameter
   1. poList
      1. Type IList<T>
   2. plAcceptChanges
      1. Type Boolean
      2. Default true
      3. Mengindikasikan datatable yg ter-create sudah merupakan final changes
   3. poMapping
      1. Type Dictionary<string,string>
      2. Bersifat optional
      3. Merupakan mapping dari nama field yg ada di class T menjadi nama column di table
4. Return
   1. Type DataTable

### Fungsi IList<T> R\_ConvertTo<T>(DataTable poTable)

1. Merupakan fungsi yg melakukan konversi dari data table ke sebuah collection Type T
2. Parameter
   1. poTable
      1. Type DataTable
3. Return
   1. Type IList<T>

## Serialize dan Deserialize

### Fungsi byte[] Serialize(object Obj)

1. Merupakan fungsi untuk melakukan serialisasi sebuah object menjadi Byte array, biasanya konversi ini dilakukan untuk men-transfer object menjadi data
2. Parameter
   1. Obj
      1. Type Object
      2. Merupakan Object yg akan di-convert
   2. Return’
      1. Type Byte array

### Fungsi object Deserialize(byte[] SerializedData)

1. Merupakan fungsi untuk melakukan Deserialisasi dari data berupa Byte array menjadi sebuah object
2. Parameter
   1. SerializedData
      1. Type Byte array
      2. Data yg akan di-convert menjadi object
3. Return
   1. Type Object

### Fungsi object Clone(object poObject)

1. Merupakan fungsi untuk melakukan duplikasi object
2. Parameter
   1. poObject
   2. Type Object
3. Return
   1. Type Object

### Fungsi string ObjectToXML(object poObject)

1. Merupakan fungsi untuk melakukan serialisasi sebuah object menjadi XML string, biasanya konversi ini dilakukan untuk men-transfer object menjadi data berupa string
2. Parameter
   1. poObj
      1. Type Object
      2. Merupakan Object yg akan di-convert
   2. Return’
   3. Type String

### Fungsi XMLToObject<T>(ByVal pcString As String) As T

1. Merupakan fungsi untuk melakukan Deserialisasi dari data berupa xml string menjadi sebuah object
2. Parameter
   1. SerializedData
      1. Type Byte array
   2. Data yg akan di-convert menjadi object
3. Return
   1. Type Object

### Fungsi byte[] GetByteFromFile(string pcFile)

1. Merupakan fungsi untuk melakukan serialisasi sebuah file menjadi Byte array, biasanya konversi ini dilakukan untuk men-transfer file menjadi data
2. Parameter
   1. pcFile
      1. Type string
      2. Merupakan File yg mau di convert
   2. Return’
      1. Type Byte array

### Fungsi void ConvertByteToFile(string pcFile, byte[] pbData)

1. Merupakan fungsi untuk melakukan Deserialisasi dari data berupa Byte array menjadi sebuah File
2. Parameter
   1. pcFile
      1. Type String
      2. Path dari File hasil
   2. pbData
      1. Type Byte array
      2. Data yg akan di-convert menjadi file

## Encrypt dan Decrypt

### Fungsi string Encrypt(string pcData, string pcKey)

1. Merupakan fungsi untuk melakukan encryption sebuah data dengan menggunakan Salt Key
2. Parameter
   1. pcData
      1. Type string
      2. Merupakan data yg akan di encrypt
   2. pcKey
      1. Type string
      2. Merupakan string yg menjadi Salt Key
   3. Return
      1. Type string

### Fungsi string Decrypt(string pcData, string pcKey)

1. Merupakan fungsi untuk melakukan decription sebuah data dengan menggunakan Salt Key
2. Parameter
   1. pcData
      1. Type string
      2. Merupakan data yg akan di decrypt
   2. pcKey
      1. Type string
      2. Merupakan string yg menjadi Salt Key
   3. Return
      1. Type string

# R\_NetCoreUtility

## Serialize dan Deserialize

Merupakan utility yang ada di common, utility ini hanya ber-fokus pada Serialisasi dan Deserialisasi untuk kebutuhan transfer batch dan Data yg berupa Byte Array dari Front ke Back.

Untuk perubahan collection ke type collection yg berbeda, kita dapat menggunakan LINQ.

### Fungsi byte[] R\_SerializeObjectToByte<T>(T poObject)

1. Merupakan fungsi untuk melakukan serialisasi sebuah object menjadi Byte array, biasanya konversi ini dilakukan untuk men-transfer object menjadi data
2. Parameter
   1. poObject
      1. Type T
      2. Merupakan Object yg akan di-convert
   2. Return
      1. Type Byte array

### Fungsi T R\_DeserializeObjectFromByte<T>(byte[] poData)

1. Merupakan fungsi untuk melakukan Deserialisasi dari data berupa Byte array menjadi sebuah object
2. Parameter
   1. poData
      1. Type Byte array
   2. Data yg akan di-convert menjadi object
3. Return
   1. Type T

### Fungsi T R\_CloneObject<T>(T poObject)

1. Merupakan fungsi untuk melakukan duplikasi object
2. Parameter
   1. poObject
   2. Type T
3. Return
   1. Type T

### Fungsi byte[] R\_SerializeFileToByte(string pcFile)

1. Merupakan fungsi untuk melakukan serialisasi sebuah file menjadi Byte array, biasanya konversi ini dilakukan untuk men-transfer file menjadi data
2. Parameter
   1. pcFile
      1. Type string
      2. Merupakan File yg mau di convert
   2. Return’
      1. Type Byte array

### Fungsi void R\_DeserializeFileFromByte(string pcFile, byte[] pbData)

1. Merupakan fungsi untuk melakukan Deserialisasi dari data berupa Byte array menjadi sebuah File
2. Parameter
   1. pcFile
      1. Type String
      2. Path dari File hasil
   2. pbData
      1. Type Byte array
      2. Data yg akan di-convert menjadi file