

Aplikacja do planowania posiłków

Generated by Doxygen 1.13.2

1 Namespace Index	1
1.1 Package List	1
2 Hierarchical Index	3
2.1 Class Hierarchy	3
3 Class Index	5
3.1 Class List	5
4 Namespace Documentation	7
4.1 Adapters Namespace Reference	7
4.2 Adapters.FileSaveAdapter Namespace Reference	7
4.3 Adapters.JsonAdapter Namespace Reference	7
4.4 DB Namespace Reference	8
4.5 Meal Namespace Reference	8
4.6 Product Namespace Reference	8
4.7 Program Namespace Reference	8
4.8 Recipe Namespace Reference	9
4.9 Recipe.Commands Namespace Reference	9
4.10 ShoppingList Namespace Reference	9
4.11 ShoppingList.Decorators Namespace Reference	9
5 Class Documentation	11
5.1 Recipe.Commands.AddIngredientCommand Class Reference	11
5.1.1 Detailed Description	11
5.1.2 Constructor & Destructor Documentation	11
5.1.2.1 AddIngredientCommand()	11
5.1.3 Member Function Documentation	12
5.1.3.1 Execute()	12
5.1.3.2 Undo()	12
5.2 ShoppingList.Decorators.CategorizedDisplayDecorator Class Reference	12
5.2.1 Detailed Description	13
5.2.2 Constructor & Destructor Documentation	13
5.2.2.1 CategorizedDisplayDecorator()	13
5.2.3 Member Function Documentation	13
5.2.3.1 DisplayProducts()	13
5.3 Recipe.Commands.CommandManager Class Reference	13
5.3.1 Detailed Description	14
5.3.2 Member Function Documentation	14
5.3.2.1 ExecuteCommand()	14
5.4 DB.ConnectionManager Class Reference	14
5.4.1 Detailed Description	15
5.4.2 Member Function Documentation	15
5.4.2.1 getConnection< T >()	15

5.4.2.2 getInstance()	16
5.5 DB.Database< T > Class Template Reference	16
5.5.1 Detailed Description	16
5.5.2 Member Function Documentation	17
5.5.2.1 getInstance()	17
5.6 DB.DatabaseConnection< T > Class Template Reference	17
5.6.1 Detailed Description	17
5.6.2 Member Function Documentation	18
5.6.2.1 AddRecord()	18
5.6.2.2 DatabaseConnection()	18
5.6.2.3 DeleteRecord()	18
5.6.2.4 GetAllRecords()	19
5.6.2.5 GetLastRecord()	19
5.6.2.6 GetRecordById()	19
5.6.2.7 GetRecordByName()	19
5.6.2.8 ShowAllRecords()	20
5.6.2.9 UpdateRecord()	20
5.7 Recipe.Commands.DeleteIngredientCommand Class Reference	20
5.7.1 Detailed Description	21
5.7.2 Constructor & Destructor Documentation	21
5.7.2.1 DeleteIngredientCommand()	21
5.7.3 Member Function Documentation	21
5.7.3.1 Execute()	21
5.7.3.2 Undo()	21
5.8 Adapters.FileSaveAdapter.FileSaveManager Class Reference	21
5.8.1 Detailed Description	22
5.8.2 Member Function Documentation	22
5.8.2.1 SaveToFile()	22
5.9 Recipe.Commands.ICommand Interface Reference	22
5.9.1 Detailed Description	23
5.9.2 Member Function Documentation	23
5.9.2.1 Execute()	23
5.9.2.2 Undo()	23
5.10 DB.IDatabaseConnection< T > Interface Template Reference	23
5.10.1 Detailed Description	23
5.10.2 Member Function Documentation	24
5.10.2.1 AddRecord()	24
5.10.2.2 DeleteRecord()	24
5.10.2.3 GetAllRecords()	24
5.10.2.4 GetLastRecord()	25
5.10.2.5 GetRecordById()	25
5.10.2.6 GetRecordByName()	25

5.10.2.7 ShowAllRecords()	25
5.10.2.8 UpdateRecord()	25
5.11 Adapters.FileSaveAdapter.IFileSaveAdapter Interface Reference	26
5.11.1 Detailed Description	26
5.11.2 Member Function Documentation	26
5.11.2.1 SaveToFile()	26
5.12 Adapters.JsonAdapter.IJsonAdapter< T > Interface Template Reference	27
5.12.1 Detailed Description	27
5.12.2 Member Function Documentation	27
5.12.2.1 Deserialize()	27
5.12.2.2 Serialize()	27
5.13 Recipe.Ingredient Class Reference	28
5.13.1 Detailed Description	28
5.13.2 Constructor & Destructor Documentation	28
5.13.2.1 Ingredient()	28
5.14 Recipe.IngredientList Class Reference	29
5.14.1 Detailed Description	29
5.14.2 Member Function Documentation	29
5.14.2.1 Add()	29
5.14.2.2 GetEnumerator()	30
5.14.2.3 GetTotalCarbohydrates()	30
5.14.2.4 GetTotalFat()	30
5.14.2.5 GetTotalProtein()	30
5.14.2.6 Remove()	30
5.15 Adapters.JsonAdapter.IngredientListAdapter Class Reference	31
5.15.1 Detailed Description	31
5.15.2 Member Function Documentation	31
5.15.2.1 Deserialize()	31
5.15.2.2 Serialize()	32
5.16 Adapters.JsonAdapter.IngredientListConverter Class Reference	32
5.16.1 Detailed Description	32
5.16.2 Member Function Documentation	32
5.16.2.1 Read()	32
5.16.2.2 Write()	33
5.17 ShoppingList.IShoppingList Interface Reference	33
5.17.1 Detailed Description	34
5.17.2 Member Function Documentation	34
5.17.2.1 AddProduct()	34
5.17.2.2 DisplayProducts()	34
5.17.2.3 GenerateFromRecipe()	34
5.17.2.4 GetItems()	35
5.17.2.5 MarkAsAdded()	35

5.17.2.6 RemoveProduct()	35
5.18 Adapters.FileSaveAdapter.JsonFileSaveAdapter Class Reference	35
5.18.1 Detailed Description	36
5.18.2 Member Function Documentation	36
5.18.2.1 SaveToFile()	36
5.19 ShoppingList.Decorators.MarkAsAddedDecorator Class Reference	36
5.19.1 Detailed Description	37
5.19.2 Constructor & Destructor Documentation	37
5.19.2.1 MarkAsAddedDecorator()	37
5.19.3 Member Function Documentation	37
5.19.3.1 DisplayProducts()	37
5.19.3.2 MarkAsAdded()	37
5.20 Meal.Meal Class Reference	38
5.20.1 Detailed Description	39
5.21 Meal.MealHistory Class Reference	39
5.21.1 Detailed Description	39
5.21.2 Member Function Documentation	39
5.21.2.1 Add()	39
5.21.2.2 Get()	40
5.21.2.3 Remove()	40
5.22 Meal.Meal.MealMemento Class Reference	40
5.22.1 Detailed Description	41
5.22.2 Constructor & Destructor Documentation	41
5.22.2.1 MealMemento()	41
5.23 Product.Product Class Reference	41
5.23.1 Detailed Description	42
5.23.2 Constructor & Destructor Documentation	42
5.23.2.1 Product()	42
5.23.3 Member Function Documentation	43
5.23.3.1 ToString()	43
5.24 Adapters.JsonAdapter.ProductAdapter Class Reference	43
5.24.1 Detailed Description	43
5.24.2 Member Function Documentation	43
5.24.2.1 Deserialize()	43
5.24.2.2 Serialize()	44
5.25 Adapters.JsonAdapter.ProductConverter Class Reference	44
5.25.1 Detailed Description	44
5.25.2 Member Function Documentation	45
5.25.2.1 Read()	45
5.25.2.2 Write()	46
5.26 Program.Program Class Reference	46
5.26.1 Detailed Description	46

5.27 Recipe.Recipe Class Reference	47
5.27.1 Detailed Description	47
5.27.2 Member Function Documentation	47
5.27.2.1 ToString()	47
5.28 Recipe.RecipeBuilder Class Reference	48
5.28.1 Detailed Description	48
5.28.2 Member Function Documentation	48
5.28.2.1 AddIngredient()	48
5.28.2.2 AddInstructionStep()	48
5.28.2.3 AddName()	49
5.28.2.4 AddTotalTime()	49
5.28.2.5 Build()	49
5.28.2.6 RemoveIngredient()	49
5.29 DB.Record< T > Class Template Reference	50
5.29.1 Detailed Description	50
5.29.2 Member Function Documentation	50
5.29.2.1 Record()	50
5.29.2.2 ToString()	51
5.30 ShoppingList.ShoppingItem Class Reference	51
5.30.1 Detailed Description	51
5.30.2 Constructor & Destructor Documentation	51
5.30.2.1 ShoppingItem()	51
5.31 ShoppingList.ShoppingList Class Reference	52
5.31.1 Detailed Description	52
5.31.2 Member Function Documentation	52
5.31.2.1 AddProduct()	52
5.31.2.2 GenerateFromRecipe()	53
5.31.2.3 RemoveProduct()	53
5.32 ShoppingList.Decorators.ShoppingListDecorator Class Reference	53
5.32.1 Detailed Description	54
5.32.2 Constructor & Destructor Documentation	54
5.32.2.1 ShoppingListDecorator()	54
5.32.3 Member Function Documentation	54
5.32.3.1 AddProduct()	54
5.32.3.2 DisplayProducts()	55
5.32.3.3 GenerateFromRecipe()	55
5.32.3.4 GetItems()	55
5.32.3.5 MarkAsAdded()	55
5.32.3.6 RemoveProduct()	56
5.33 Adapters.FileSaveAdapter.TxtFileSaveAdapter Class Reference	56
5.33.1 Detailed Description	56
5.33.2 Member Function Documentation	56

5.33.2.1 SaveToFile()	56
---------------------------------	----

Index	59
--------------	-----------

Chapter 1

Namespace Index

1.1 Package List

Here are the packages with brief descriptions (if available):

Adapters	7
Adapters.FileSaveAdapter	7
Adapters.JsonAdapter	7
DB	8
Meal	8
Product	8
Program	8
Recipe	9
Recipe.Commands	9
ShoppingList	9
ShoppingList.Decorators	9

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Recipe.Commands.CommandManager	13
DB.ConnectionManager	14
DB.Database< T >	16
Adapters.FileSaveAdapter.FileSaveManager	21
Recipe.Commands.ICommand	22
Recipe.Commands.AddIngredientCommand	11
Recipe.Commands.DeleteIngredientCommand	20
DB.IDatabaseConnection< T >	23
DB.DatabaseConnection< T >	17
Adapters.FileSaveAdapter.IFileSaveAdapter	26
Adapters.FileSaveAdapter.JsonFileSaveAdapter	35
Adapters.FileSaveAdapter.TxtFileSaveAdapter	56
Adapters.JsonAdapter.IJsonAdapter< T >	27
Adapters.JsonAdapter.IngredientListAdapter	31
Adapters.JsonAdapter.ProductAdapter	43
Recipe.Ingredient	28
Recipe.IngredientList	29
ShoppingList.IShoppingList	33
ShoppingList.Decorators.ShoppingListDecorator	53
ShoppingList.Decorators.CategorizedDisplayDecorator	12
ShoppingList.Decorators.MarkAsAddedDecorator	36
ShoppingList.ShoppingList	52
JsonConverter	
Adapters.JsonAdapter.IngredientListConverter	32
Adapters.JsonAdapter.ProductConverter	44
Meal.Meal	38
Meal.MealHistory	39
Meal.Meal.MealMemento	40
Product.Product	41
Program.Program	46
Recipe.Recipe	47
Recipe.RecipeBuilder	48
DB.Record< T >	50
ShoppingList.ShoppingItem	51

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Recipe.Commands.AddIngredientCommand	
Komenda dodająca składnik do przepisu	11
ShoppingList.Decorators.CategorizedDisplayDecorator	
Dekorator, który wyświetla produkty na liście zakupów według kategorii	12
Recipe.Commands.CommandManager	
Zarządza wykonywaniem, cofaniem i ponownym wykonywaniem komend	13
DB.ConnectionManager	
Zarządza połączeniami do bazy danych, implementując wzorzec Singleton	14
DB.Database< T >	
Klasa reprezentująca bazę danych, implementująca wzorzec Multiton	16
DB.DatabaseConnection< T >	
Klasa reprezentująca połączenie z bazą danych	17
Recipe.Commands.DeleteIngredientCommand	
Komenda usuwająca składnik z przepisu	20
Adapters.FileSaveAdapter.FileSaveManager	
Zarządza zapisywaniem danych do plików za pomocą różnych adapterów w zależności od rozszerzenia pliku	21
Recipe.Commands.ICommand	
Definiuje interfejs dla komend, które mogą być wykonane i cofnięte	22
DB.IDatabaseConnection< T >	
Interfejs definiujący metody połączenia z bazą danych	23
Adapters.FileSaveAdapter.IFileSaveAdapter	
Interfejs definiujący metodę zapisywania danych do pliku	26
Adapters.JsonAdapter.IJsonAdapter< T >	
Interfejs definiujący metody serializacji i deserializacji danych JSON	27
Recipe.Ingredient	
Reprezentuje składnik przepisu, zawierający produkt i jego ilość	28
Recipe.IngredientList	
Reprezentuje listę składników przepisu	29
Adapters.JsonAdapter.IngredientListAdapter	
Adapter do serializacji i deserializacji listy składników przepisu	31
Adapters.JsonAdapter.IngredientListConverter	
Konwerter JSON dla obiektu Recipe.IngredientList	32
ShoppingList.IShoppingList	
Definiuje metody dla listy zakupów	33

Adapters.FileSaveAdapter.JsonFileSaveAdapter	
Adapter do zapisywania danych w formacie JSON	35
ShoppingList.Decorators.MarkAsAddedDecorator	
Dekorator, który oznacza produkt jako dodany do koszyka i wyświetla listę zakupów według kategorii	36
Meal.Meal	
Reprezentuje posiłek, zawierający nazwę, datę, typ posiłku oraz listę przepisów	38
Meal.MealHistory	
Reprezentuje historię posiłków, przechowującą memento posiłków	39
Meal.Meal.MealMemento	
Reprezentuje memento posiłku, zawierające nazwę, datę i typ posiłku	40
Product.Product	
Reprezentuje produkt z nazwą, kategorią, jednostką miary oraz wartościami odżywczymi . . .	41
Adapters.JsonAdapter.ProductAdapter	
Adapter do serializacji i deserializacji obiektu Product.Product	43
Adapters.JsonAdapter.ProductConverter	
Konwerter JSON dla obiektu Product.Product	44
Program.Program	
Główna klasa programu	46
Recipe.Recipe	
Reprezentuje przepis kulinarny, zawierający nazwę, listę składników, instrukcje, czas przygotowania oraz wartości odżywcze	47
Recipe.RecipeBuilder	
Klasa budowniczego dla obiektu Recipe , umożliwiającą stopniowe dodawanie składników, instrukcji i innych właściwości	48
DB.Record< T >	
Klasa reprezentująca rekord w bazie danych	50
ShoppingList.ShoppingItem	
Reprezentuje pozycję na liście zakupów, która zawiera produkt i jego ilość	51
ShoppingList.ShoppingList	
Reprezentuje listę zakupów, która może przechowywać produkty i ich ilości	52
ShoppingList.Decorators.ShoppingListDecorator	
Abstrakcyjna klasa dekoratora dla listy zakupów, implementująca interfejs IShoppingList	53
Adapters.FileSaveAdapter.TxtFileSaveAdapter	
Adapter do zapisywania danych w formacie tekstowym	56

Chapter 4

Namespace Documentation

4.1 Adapters Namespace Reference

4.2 Adapters.FileSaveAdapter Namespace Reference

Classes

- class [FileSaveManager](#)
Zarządza zapisywaniem danych do plików za pomocą różnych adapterów w zależności od rozszerzenia pliku.
- interface [IFileSaveAdapter](#)
Interfejs definiujący metodę zapisywania danych do pliku.
- class [JsonFileSaveAdapter](#)
Adapter do zapisywania danych w formacie JSON.
- class [TxtFileSaveAdapter](#)
Adapter do zapisywania danych w formacie tekstowym.

4.3 Adapters.JsonAdapter Namespace Reference

Classes

- interface [IJsonAdapter< T >](#)
Interfejs definiujący metody serializacji i deserializacji danych JSON.
- class [IngredientListAdapter](#)
Adapter do serializacji i deserializacji listy składników przepisu.
- class [IngredientListConverter](#)
Konwerter JSON dla obiektu [Recipe.IngredientList](#).
- class [ProductAdapter](#)
Adapter do serializacji i deserializacji obiektu [Product.Product](#).
- class [ProductConverter](#)
Konwerter JSON dla obiektu [Product.Product](#).

4.4 DB Namespace Reference

Classes

- class [ConnectionManager](#)
Zarządza połączeniami do bazy danych, implementując wzorzec Singleton.
- class [Database< T >](#)
Klasa reprezentująca bazę danych, implementująca wzorzec Multiton.
- class [DatabaseConnection< T >](#)
Klasa reprezentująca połączenie z bazą danych.
- interface [IDatabaseConnection< T >](#)
Interfejs definiujący metody połączenia z bazą danych.
- class [Record< T >](#)
Klasa reprezentująca rekord w bazie danych.

4.5 Meal Namespace Reference

Classes

- class [Meal](#)
Reprezentuje posiłek, zawierający nazwę, datę, typ posiłku oraz listę przepisów.
- class [MealHistory](#)
Reprezentuje historię posiłków, przechowującą mementa posiłków.

4.6 Product Namespace Reference

Classes

- class [Product](#)
Reprezentuje produkt z nazwą, kategorią, jednostką miary oraz wartościami odżywczymi.

4.7 Program Namespace Reference

Classes

- class [Program](#)
Główna klasa programu.

4.8 Recipe Namespace Reference

Classes

- class [Ingredient](#)
Reprezentuje składnik przepisu, zawierający produkt i jego ilość.
- class [IngredientList](#)
Reprezentuje listę składników przepisu.
- class [Recipe](#)
Reprezentuje przepis kulinarny, zawierający nazwę, listę składników, instrukcje, czas przygotowania oraz wartości odżywcze.
- class [RecipeBuilder](#)
Klasa budowniczego dla obiektu [Recipe](#), umożliwiającą stopniowe dodawanie składników, instrukcji i innych właściwości.

4.9 Recipe.Commands Namespace Reference

Classes

- class [AddIngredientCommand](#)
Komenda dodająca składnik do przepisu.
- class [CommandManager](#)
Zarządza wykonywaniem, cofaniem i ponownym wykonywaniem komend.
- class [DeleteIngredientCommand](#)
Komenda usuwająca składnik z przepisu.
- interface [ICommand](#)
Definiuje interfejs dla komend, które mogą być wykonane i cofnięte.

4.10 ShoppingList Namespace Reference

Classes

- interface [IShoppingList](#)
Definiuje metody dla listy zakupów.
- class [ShoppingItem](#)
Reprezentuje pozycję na liście zakupów, która zawiera produkt i jego ilość.
- class [ShoppingList](#)
Reprezentuje listę zakupów, która może przechowywać produkty i ich ilości.

4.11 ShoppingList.Decorators Namespace Reference

Classes

- class [CategorizedDisplayDecorator](#)
Dekorator, który wyświetla produkty na liście zakupów według kategorii.
- class [MarkAsAddedDecorator](#)
Dekorator, który oznacza produkt jako dodany do koszyka i wyświetla listę zakupów według kategorii.
- class [ShoppingListDecorator](#)
Abstrakcyjna klasa dekoratora dla listy zakupów, implementująca interfejs [IShoppingList](#).

Chapter 5

Class Documentation

5.1 Recipe.Commands.AddIngredientCommand Class Reference

Komenda dodająca składnik do przepisu.

Public Member Functions

- [AddIngredientCommand](#) ([RecipeBuilder](#) recipeBuilder, [Product.Product](#) product, double amount)
Inicjalizuje nową instancję klasy [AddIngredientCommand](#) z podanym budowniczym przepisu, produktem i ilością.
- void [Execute](#) ()
Wykonuje komendę dodania składnika.
- void [Undo](#) ()
Cofa wykonanie komendy dodania składnika.

5.1.1 Detailed Description

Komenda dodająca składnik do przepisu.

5.1.2 Constructor & Destructor Documentation

5.1.2.1 AddIngredientCommand()

```
Recipe.Commands.AddIngredientCommand.AddIngredientCommand (  
    RecipeBuilder recipeBuilder,  
    Product.Product product,  
    double amount)
```

Inicjalizuje nową instancję klasy [AddIngredientCommand](#) z podanym budowniczym przepisu, produktem i ilością.

Parameters

<i>recipeBuilder</i>	Budowniczy przepisu.
<i>product</i>	Produkt do dodania.
<i>amount</i>	Ilość produktu.

5.1.3 Member Function Documentation

5.1.3.1 Execute()

```
void Recipe.Commands.AddIngredientCommand.Execute ()
```

Wykonuje komendę dodania składnika.

Implements [Recipe.Commands.ICommand](#).

5.1.3.2 Undo()

```
void Recipe.Commands.AddIngredientCommand.Undo ()
```

Cofa wykonanie komendy dodania składnika.

Implements [Recipe.Commands.ICommand](#).

The documentation for this class was generated from the following file:

- AddIngredientCommand.cs

5.2 ShoppingList.Decorators.CategorizedDisplayDecorator Class Reference

Dekorator, który wyświetla produkty na liście zakupów według kategorii.

Public Member Functions

- [CategorizedDisplayDecorator](#) ([IShoppingList](#) shoppingList)
Inicjalizuje nową instancję klasy [CategorizedDisplayDecorator](#) z podaną listą zakupów.
- override void [DisplayProducts](#) ()
Wyświetla produkty na liście zakupów według kategorii.

Public Member Functions inherited from [ShoppingList.Decorators.ShoppingListDecorator](#)

- [ShoppingListDecorator](#) ([IShoppingList](#) shoppingList)
Inicjalizuje nową instancję klasy [ShoppingListDecorator](#) z podaną listą zakupów.
- virtual void [AddProduct](#) ([Product.Product](#) product, int quantity)
Dodaje produkt do listy zakupów z określoną ilością.
- virtual void [GenerateFromRecipe](#) ([Recipe.Recipe](#) recipe)
Generuje pozycje listy zakupów na podstawie podanego przepisu.
- virtual void [RemoveProduct](#) ([Product.Product](#) product)
Usuwa produkt z listy zakupów.
- virtual void [MarkAsAdded](#) ([Product.Product](#) product)
Oznacza produkt jako dodany do koszyka.
- virtual List< [ShoppingItem](#) > [GetItems](#) ()
Zwraca listę pozycji na liście zakupów.

Additional Inherited Members

Protected Attributes inherited from [ShoppingList.Decorators.ShoppingListDecorator](#)

- [IShoppingList](#) shoppingList

5.2.1 Detailed Description

Dekorator, który wyświetla produkty na liście zakupów według kategorii.

5.2.2 Constructor & Destructor Documentation

5.2.2.1 CategorizedDisplayDecorator()

```
ShoppingList.Decorators.CategorizedDisplayDecorator.CategorizedDisplayDecorator (  
    IShoppingList shoppingList)
```

Inicjalizuje nową instancję klasy [CategorizedDisplayDecorator](#) z podaną listą zakupów.

Parameters

<i>shoppingList</i>	Lista zakupów do dekorowania.
---------------------	-------------------------------

5.2.3 Member Function Documentation

5.2.3.1 DisplayProducts()

```
override void ShoppingList.Decorators.CategorizedDisplayDecorator.DisplayProducts () [virtual]
```

Wyświetla produkty na liście zakupów według kategorii.

Reimplemented from [ShoppingList.Decorators.ShoppingListDecorator](#).

The documentation for this class was generated from the following file:

- CategorizedDisplayDecorator.cs

5.3 Recipe.Commands.CommandManager Class Reference

Zarządza wykonywaniem, cofaniem i ponownym wykonywaniem komend.

Public Member Functions

- void [ExecuteCommand](#) ([ICommand](#) command)
Wykonuje podaną komendę i dodaje ją do stosu cofania.
- void **Undo** ()
Cofa ostatnio wykonaną komendę.
- void **Redo** ()
Ponownie wykonuje ostatnio cofniętą komendę.

5.3.1 Detailed Description

Zarządza wykonywaniem, cofaniem i ponownym wykonywaniem komend.

5.3.2 Member Function Documentation

5.3.2.1 ExecuteCommand()

```
void Recipe.Commands.CommandManager.ExecuteCommand (  
    ICommand command)
```

Wykonuje podaną komendę i dodaje ją do stosu cofania.

Parameters

<i>command</i>	Komenda do wykonania.
----------------	-----------------------

The documentation for this class was generated from the following file:

- CommandManager.cs

5.4 DB.ConnectionManager Class Reference

Zarządza połączeniami do bazy danych, implementując wzorzec Singleton.

Public Member Functions

- DatabaseConnection< T > [getConnection](#)< T > (string name)
Zwraca połączenie do bazy danych o podanej nazwie.

Static Public Member Functions

- static [ConnectionManager](#) getInstance ()
Zwraca jedyną instancję klasy [ConnectionManager](#).

5.4.1 Detailed Description

Zarządza połączeniami do bazy danych, implementując wzorzec Singleton.

5.4.2 Member Function Documentation

5.4.2.1 getConnection< T >()

```
DatabaseConnection< T > DB.ConnectionManager.getConnection< T > (  
    string name)
```

Zwraca połączenie do bazy danych o podanej nazwie.

Template Parameters

<i>T</i>	Typ danych, z którymi będzie pracować połączenie.
----------	---

Parameters

<i>name</i>	Nazwa bazy danych.
-------------	--------------------

Returns

Połączenie do bazy danych typu [DatabaseConnection<T>](#).

5.4.2.2 getInstance()

```
static ConnectionManager DB.ConnectionManager.getInstance () [static]
```

Zwraca jedną instancję klasy [ConnectionManager](#).

Returns

Instancja klasy [ConnectionManager](#).

The documentation for this class was generated from the following file:

- ConnectionManager.cs

5.5 DB.Database< T > Class Template Reference

Klasa reprezentująca bazę danych, implementująca wzorzec Multiton.

Static Public Member Functions

- static Database< T > [getInstance](#) (string key)
Zwraca instancję bazy danych dla podanego klucza.

Public Attributes

- readonly List< Record< T > > **records**
Lista przechowująca rekordy.
- int **nextId** = 1
Licznik do generowania unikalnych ID.
- int **dbId** = 0
ID bazy danych.

5.5.1 Detailed Description

Klasa reprezentująca bazę danych, implementująca wzorzec Multiton.

Template Parameters

T	Typ danych przechowywanych w bazie danych.
----------	--

5.5.2 Member Function Documentation

5.5.2.1 getInstance()

```
static Database< T > DB.Database< T >.getInstance (
    string key) [static]
```

Zwraca instancję bazy danych dla podanego klucza.

Parameters

key	Klucz identyfikujący instancję bazy danych.
------------	---

Returns

Instancja bazy danych typu [Database<T>](#).

The documentation for this class was generated from the following file:

- Database.cs

5.6 DB.DatabaseConnection< T > Class Template Reference

Klasa reprezentująca połączenie z bazą danych.

Public Member Functions

- [DatabaseConnection](#) (string name)
Inicjalizuje nową instancję klasy [DatabaseConnection<T>](#).
- int [AddRecord](#) (T obj)
Dodaje nowy rekord do bazy danych.
- Record< T >? [GetRecordById](#) (int id)
Pobiera rekord z bazy danych na podstawie ID.
- Record< T >? [GetRecordByName](#) (string name)
Pobiera rekord z bazy danych na podstawie nazwy.
- void [UpdateRecord](#) (int id, T obj)
Aktualizuje istniejący rekord w bazie danych.
- void [DeleteRecord](#) (int id)
Usuwa rekord z bazy danych.
- List< Record< T > > [GetAllRecords](#) ()
Pobiera wszystkie rekordy z bazy danych.
- T? [GetLastRecord](#) ()
Pobiera ostatni rekord z bazy danych.
- void [ShowAllRecords](#) ()
Wyświetla wszystkie rekordy z bazy danych.

5.6.1 Detailed Description

Klasa reprezentująca połączenie z bazą danych.

Template Parameters

<i>T</i>	Typ danych przechowywanych w bazie danych.
----------	--

5.6.2 Member Function Documentation

5.6.2.1 AddRecord()

```
int DB.DatabaseConnection< T >.AddRecord (  
    T obj)
```

Dodaje nowy rekord do bazy danych.

Parameters

<i>obj</i>	Obiekt do dodania.
------------	--------------------

Returns

ID dodanego rekordu lub -1, jeśli rekord o danej nazwie już istnieje.

Implements [DB.IDatabaseConnection< T >](#).

5.6.2.2 DatabaseConnection()

```
DB.DatabaseConnection< T >.DatabaseConnection (  
    string name)
```

Inicjalizuje nową instancję klasy [DatabaseConnection<T>](#).

Parameters

<i>name</i>	Nazwa bazy danych.
-------------	--------------------

5.6.2.3 DeleteRecord()

```
void DB.DatabaseConnection< T >.DeleteRecord (  
    int id)
```

Usuwa rekord z bazy danych.

Parameters

<i>id</i>	ID rekordu do usunięcia.
-----------	--------------------------

Implements [DB.IDatabaseConnection< T >](#).

5.6.2.4 GetAllRecords()

```
List< Record< T > > DB.DatabaseConnection< T >.GetAllRecords ()
```

Pobiera wszystkie rekordy z bazy danych.

Returns

Lista wszystkich rekordów.

Implements [DB.IDatabaseConnection< T >](#).

5.6.2.5 GetLastRecord()

```
T? DB.DatabaseConnection< T >.GetLastRecord ()
```

Pobiera ostatni rekord z bazy danych.

Returns

Ostatni rekord lub domyślną wartość typu *T*.

Implements [DB.IDatabaseConnection< T >](#).

5.6.2.6 GetRecordById()

```
Record< T >? DB.DatabaseConnection< T >.GetRecordById (  
    int id)
```

Pobiera rekord z bazy danych na podstawie ID.

Parameters

<i>id</i>	ID rekordu do pobrania.
-----------	-------------------------

Returns

Rekord typu [Record<T>](#) lub null, jeśli rekord nie istnieje.

Implements [DB.IDatabaseConnection< T >](#).

5.6.2.7 GetRecordByName()

```
Record< T >? DB.DatabaseConnection< T >.GetRecordByName (  
    string name)
```

Pobiera rekord z bazy danych na podstawie nazwy.

Parameters

<i>name</i>	Nazwa rekordu do pobrania.
-------------	----------------------------

Returns

Rekord typu [Record<T>](#) lub null, jeśli rekord nie istnieje.

Implements [DB.IDatabaseConnection< T >](#).

5.6.2.8 ShowAllRecords()

```
void DB.DatabaseConnection< T >.ShowAllRecords ()
```

Wyświetla wszystkie rekordy z bazy danych.

Implements [DB.IDatabaseConnection< T >](#).

5.6.2.9 UpdateRecord()

```
void DB.DatabaseConnection< T >.UpdateRecord (  
    int id,  
    T obj)
```

Aktualizuje istniejący rekord w bazie danych.

Parameters

<i>id</i>	ID rekordu do aktualizacji.
<i>obj</i>	Zaktualizowany obiekt.

Implements [DB.IDatabaseConnection< T >](#).

The documentation for this class was generated from the following file:

- DatabaseConnection.cs

5.7 Recipe.Commands.DeleteIngredientCommand Class Reference

Komenda usuwająca składnik z przepisu.

Public Member Functions

- [DeleteIngredientCommand](#) ([RecipeBuilder](#) recipeBuilder, [Product.Product](#) product)
Inicjalizuje nową instancję klasy [DeleteIngredientCommand](#) z podanym budowniczym przepisem i produktem.
- void [Execute](#) ()
Wykonuje komendę usunięcia składnika.
- void [Undo](#) ()
Cofa wykonanie komendy usunięcia składnika.

5.7.1 Detailed Description

Komenda usuwająca składnik z przepisu.

5.7.2 Constructor & Destructor Documentation

5.7.2.1 DeleteIngredientCommand()

```
Recipe.Commands.DeleteIngredientCommand.DeleteIngredientCommand (  
    RecipeBuilder recipeBuilder,  
    Product.Product product)
```

Inicjalizuje nową instancję klasy [DeleteIngredientCommand](#) z podanym budowniczym przepisu i produktem.

Parameters

<i>recipeBuilder</i>	Budowniczy przepisu.
<i>product</i>	Produkt do usunięcia.

5.7.3 Member Function Documentation

5.7.3.1 Execute()

```
void Recipe.Commands.DeleteIngredientCommand.Execute ()
```

Wykonuje komendę usunięcia składnika.

Implements [Recipe.Commands.ICommand](#).

5.7.3.2 Undo()

```
void Recipe.Commands.DeleteIngredientCommand.Undo ()
```

Cofa wykonanie komendy usunięcia składnika.

Implements [Recipe.Commands.ICommand](#).

The documentation for this class was generated from the following file:

- DeleteIngredientCommand.cs

5.8 Adapters.FileSaveAdapter.FileSaveManager Class Reference

Zarządza zapisywaniem danych do plików za pomocą różnych adapterów w zależności od rozszerzenia pliku.

Public Member Functions

- void [SaveToFile](#) (string filePath, object data)
Zapisuje określone dane do pliku w podanej ścieżce pliku.

5.8.1 Detailed Description

Zarządza zapisywaniem danych do plików za pomocą różnych adapterów w zależności od rozszerzenia pliku.

5.8.2 Member Function Documentation

5.8.2.1 SaveToFile()

```
void Adapters.FileSaveAdapter.FileSaveManager.SaveToFile (  
    string filePath,  
    object data)
```

Zapisuje określone dane do pliku w podanej ścieżce pliku.

Parameters

<i>filePath</i>	Ścieżka pliku, do którego mają zostać zapisane dane.
<i>data</i>	Dane do zapisania w pliku.

Exceptions

<i>NotSupportedException</i>	Rzucany, gdy format pliku nie jest obsługiwany.
------------------------------	---

The documentation for this class was generated from the following file:

- FileSaveManager.cs

5.9 Recipe.Commands.ICommand Interface Reference

Definiuje interfejs dla komend, które mogą być wykonane i cofnięte.

Public Member Functions

- void [Execute](#) ()
Wykonuje komendę.
- void [Undo](#) ()
Cofa wykonanie komendy.

5.9.1 Detailed Description

Definiuje interfejs dla komend, które mogą być wykonane i cofnięte.

5.9.2 Member Function Documentation

5.9.2.1 Execute()

```
void Recipe.Commands.ICommand.Execute ()
```

Wykonuje komendę.

Implemented in [Recipe.Commands.AddIngredientCommand](#), and [Recipe.Commands.DeleteIngredientCommand](#).

5.9.2.2 Undo()

```
void Recipe.Commands.ICommand.Undo ()
```

Cofa wykonanie komendy.

Implemented in [Recipe.Commands.AddIngredientCommand](#), and [Recipe.Commands.DeleteIngredientCommand](#).

The documentation for this interface was generated from the following file:

- ICommand.cs

5.10 DB.IDatabaseConnection< T > Interface Template Reference

Interfejs definiujący metody połączenia z bazą danych.

Public Member Functions

- int [AddRecord](#) (T obj)
Dodaje nowy rekord do bazy danych.
- void [UpdateRecord](#) (int id, T obj)
Aktualizuje istniejący rekord w bazie danych.
- void [DeleteRecord](#) (int id)
Usuwa rekord z bazy danych.
- Record< T >? [GetRecordById](#) (int id)
Pobiera rekord z bazy danych na podstawie ID.
- Record< T >? [GetRecordByName](#) (string name)
Pobiera rekord z bazy danych na podstawie nazwy.
- List< Record< T > > [GetAllRecords](#) ()
Pobiera wszystkie rekordy z bazy danych.
- T? [GetLastRecord](#) ()
Pobiera ostatni rekord z bazy danych.
- void [ShowAllRecords](#) ()
Wyświetla wszystkie rekordy z bazy danych.

5.10.1 Detailed Description

Interfejs definiujący metody połączenia z bazą danych.

Template Parameters

<i>T</i>	Typ danych przechowywanych w bazie danych.
----------	--

5.10.2 Member Function Documentation

5.10.2.1 AddRecord()

```
int DB.IDatabaseConnection< T >.AddRecord (  
    T obj)
```

Dodaje nowy rekord do bazy danych.

Parameters

<i>obj</i>	Obiekt do dodania.
------------	--------------------

Returns

ID dodanego rekordu.

Implemented in [DB.DatabaseConnection< T >](#).

5.10.2.2 DeleteRecord()

```
void DB.IDatabaseConnection< T >.DeleteRecord (  
    int id)
```

Usuwa rekord z bazy danych.

Parameters

<i>id</i>	ID rekordu do usunięcia.
-----------	--------------------------

Implemented in [DB.DatabaseConnection< T >](#).

5.10.2.3 GetAllRecords()

```
List< Record< T > > DB.IDatabaseConnection< T >.GetAllRecords ()
```

Pobiera wszystkie rekordy z bazy danych.

Returns

Lista wszystkich rekordów.

Implemented in [DB.DatabaseConnection< T >](#).

5.10.2.4 GetLastRecord()

```
T? DB.IDatabaseConnection< T >.GetLastRecord ()
```

Pobiera ostatni rekord z bazy danych.

Returns

Ostatni rekord lub domyślną wartość typu *T*.

Implemented in [DB.DatabaseConnection< T >](#).

5.10.2.5 GetRecordById()

```
Record< T >? DB.IDatabaseConnection< T >.GetRecordById (  
    int id)
```

Pobiera rekord z bazy danych na podstawie ID.

Parameters

<i>id</i>	ID rekordu do pobrania.
-----------	-------------------------

Returns

Rekord typu [Record<T>](#) lub null, jeśli rekord nie istnieje.

Implemented in [DB.DatabaseConnection< T >](#).

5.10.2.6 GetRecordByName()

```
Record< T >? DB.IDatabaseConnection< T >.GetRecordByName (  
    string name)
```

Pobiera rekord z bazy danych na podstawie nazwy.

Parameters

<i>name</i>	Nazwa rekordu do pobrania.
-------------	----------------------------

Returns

Rekord typu [Record<T>](#) lub null, jeśli rekord nie istnieje.

Implemented in [DB.DatabaseConnection< T >](#).

5.10.2.7 ShowAllRecords()

```
void DB.IDatabaseConnection< T >.ShowAllRecords ()
```

Wyświetla wszystkie rekordy z bazy danych.

Implemented in [DB.DatabaseConnection< T >](#).

5.10.2.8 UpdateRecord()

```
void DB.IDatabaseConnection< T >.UpdateRecord (  
    int id,  
    T obj)
```

Aktualizuje istniejący rekord w bazie danych.

Parameters

<i>id</i>	ID rekordu do aktualizacji.
<i>obj</i>	Zaktualizowany obiekt.

Implemented in [DB.DatabaseConnection< T >](#).

The documentation for this interface was generated from the following file:

- [IDatabaseConnection.cs](#)

5.11 Adapters.FileSaveAdapter.IFileSaveAdapter Interface Reference

Interfejs definiujący metodę zapisywania danych do pliku.

Public Member Functions

- void [SaveToFile](#) (string filePath, object data)
Zapisuje dane do pliku w podanej ścieżce pliku.

5.11.1 Detailed Description

Interfejs definiujący metodę zapisywania danych do pliku.

5.11.2 Member Function Documentation

5.11.2.1 SaveToFile()

```
void Adapters.FileSaveAdapter.IFileSaveAdapter.SaveToFile (  
    string filePath,  
    object data)
```

Zapisuje dane do pliku w podanej ścieżce pliku.

Parameters

<i>filePath</i>	Ścieżka pliku, do którego mają zostać zapisane dane.
<i>data</i>	Dane do zapisania w pliku.

Implemented in [Adapters.FileSaveAdapter.JsonFileSaveAdapter](#), and [Adapters.FileSaveAdapter.TxtFileSaveAdapter](#).

The documentation for this interface was generated from the following file:

- [IFileSaveAdapter.cs](#)

5.12 Adapters.JsonAdapter.IJsonAdapter< T > Interface Template Reference

Interfejs definiujący metody serializacji i deserializacji danych JSON.

Public Member Functions

- T [Deserialize](#) (ref Utf8JsonReader reader, JsonSerializerOptions options)
Deserializuje dane JSON do obiektu typu T.
- void [Serialize](#) (Utf8JsonWriter writer, T value, JsonSerializerOptions options)
Serializuje obiekt typu T do formatu JSON.

5.12.1 Detailed Description

Interfejs definiujący metody serializacji i deserializacji danych JSON.

Template Parameters

<i>T</i>	Typ danych do serializacji i deserializacji.
----------	--

5.12.2 Member Function Documentation

5.12.2.1 Deserialize()

```
T Adapters.JsonAdapter.IJsonAdapter< T >.Deserialize (  
    ref Utf8JsonReader reader,  
    JsonSerializerOptions options)
```

Deserializuje dane JSON do obiektu typu *T*.

Parameters

<i>reader</i>	Reader JSON.
<i>options</i>	Opcje serializacji JSON.

Returns

Obiekt typu *T*.

Implemented in [Adapters.JsonAdapter.IngredientListAdapter](#), and [Adapters.JsonAdapter.ProductAdapter](#).

5.12.2.2 Serialize()

```
void Adapters.JsonAdapter.IJsonAdapter< T >.Serialize (  
    Utf8JsonWriter writer,  
    T value,  
    JsonSerializerOptions options)
```

Serializuje obiekt typu *T* do formatu JSON.

Parameters

<i>writer</i>	Writer JSON.
<i>value</i>	Wartość do serializacji.
<i>options</i>	Opcje serializacji JSON.

The documentation for this interface was generated from the following file:

- IJsonAdapter.cs

5.13 Recipe.Ingredient Class Reference

Reprezentuje składnik przepisu, zawierający produkt i jego ilość.

Public Member Functions

- [Ingredient](#) ([Product.Product](#) product, double amount)
Inicjalizuje nową instancję klasy [Ingredient](#) z określonym produktem i ilością.

Properties

- [Product.Product](#) **Product** [get, set]
Produkt będący składnikiem przepisu.
- double **Amount** [get, set]
Ilość produktu.

5.13.1 Detailed Description

Reprezentuje składnik przepisu, zawierający produkt i jego ilość.

5.13.2 Constructor & Destructor Documentation

5.13.2.1 Ingredient()

```
Recipe.Ingredient.Ingredient (
    Product.Product product,
    double amount)
```

Inicjalizuje nową instancję klasy [Ingredient](#) z określonym produktem i ilością.

Parameters

<i>product</i>	Produkt będący składnikiem przepisu.
<i>amount</i>	Ilość produktu.

The documentation for this class was generated from the following file:

- Ingredient.cs

5.14 Recipe.IngredientList Class Reference

Reprezentuje listę składników przepisu.

Public Member Functions

- **IngredientList** ()
Inicjalizuje nową instancję klasy [IngredientList](#).
- **IEnumerator< [Ingredient](#) > GetEnumerator** ()
Zwraca enumerator dla listy składników.
- **void Add** ([Product.Product](#) product, double amount)
Dodaje składnik do listy.
- **void Remove** ([Product.Product](#) product)
Usuwa składnik z listy.
- **double GetTotalProtein** ()
Oblicza całkowitą ilość białka w składnikach.
- **double GetTotalFat** ()
Oblicza całkowitą ilość tłuszczu w składnikach.
- **double GetTotalCarbohydrates** ()
Oblicza całkowitą ilość węglowodanów w składnikach.

Properties

- **List< [Ingredient](#) > Ingredients** [get, set]
Lista składników.

5.14.1 Detailed Description

Reprezentuje listę składników przepisu.

5.14.2 Member Function Documentation

5.14.2.1 Add()

```
void Recipe.IngredientList.Add (  
    Product.Product product,  
    double amount)
```

Dodaje składnik do listy.

Parameters

<i>product</i>	Produkt do dodania.
<i>amount</i>	Ilość produktu.

5.14.2.2 GetEnumerator()

```
IEnumerator< Ingredient > Recipe.IngredientList.GetEnumerator ()
```

Zwraca enumerator dla listy składników.

Returns

Enumerator dla listy składników.

5.14.2.3 GetTotalCarbohydrates()

```
double Recipe.IngredientList.GetTotalCarbohydrates ()
```

Oblicza całkowitą ilość węglowodanów w składnikach.

Returns

Całkowita ilość węglowodanów.

5.14.2.4 GetTotalFat()

```
double Recipe.IngredientList.GetTotalFat ()
```

Oblicza całkowitą ilość tłuszczu w składnikach.

Returns

Całkowita ilość tłuszczu.

5.14.2.5 GetTotalProtein()

```
double Recipe.IngredientList.GetTotalProtein ()
```

Oblicza całkowitą ilość białka w składnikach.

Returns

Całkowita ilość białka.

5.14.2.6 Remove()

```
void Recipe.IngredientList.Remove (  
    Product.Product product)
```

Usuwa składnik z listy.

Parameters

<i>product</i>	Produkt do usunięcia.
----------------	-----------------------

The documentation for this class was generated from the following file:

- IngredientList.cs

5.15 Adapters.JsonAdapter.IngredientListAdapter Class Reference

Adapter do serializacji i deserializacji listy składników przepisu.

Public Member Functions

- [Recipe.IngredientList Deserialize](#) (ref Utf8JsonReader reader, JsonSerializerOptions options)
Deserializuje dane JSON do obiektu typu [Recipe.IngredientList](#).
- void [Serialize](#) (Utf8JsonWriter writer, [Recipe.IngredientList](#) value, JsonSerializerOptions options)
Serializuje obiekt typu [Recipe.IngredientList](#) do formatu JSON.

Public Member Functions inherited from [Adapters.JsonAdapter.IJsonAdapter< T >](#)

- void [Serialize](#) (Utf8JsonWriter writer, T value, JsonSerializerOptions options)
Serializuje obiekt typu T do formatu JSON.

5.15.1 Detailed Description

Adapter do serializacji i deserializacji listy składników przepisu.

5.15.2 Member Function Documentation

5.15.2.1 Deserialize()

```
Recipe.IngredientList Adapters.JsonAdapter.IngredientListAdapter.Deserialize (  
    ref Utf8JsonReader reader,  
    JsonSerializerOptions options)
```

Deserializuje dane JSON do obiektu typu [Recipe.IngredientList](#).

Parameters

<i>reader</i>	Reader JSON.
<i>options</i>	Opcje serializacji JSON.

Returns

Obiekt typu [Recipe.IngredientList](#).

Implements [Adapters.JsonAdapter.IJsonAdapter< T >](#).

5.15.2.2 Serialize()

```
void Adapters.JsonAdapter.IngredientListAdapter.Serialize (
    Utf8JsonWriter writer,
    Recipe.IngredientList value,
    JsonSerializerOptions options)
```

Serializuje obiekt typu [Recipe.IngredientList](#) do formatu JSON.

Parameters

<i>writer</i>	Writer JSON.
<i>value</i>	Wartość do serializacji.
<i>options</i>	Opcje serializacji JSON.

The documentation for this class was generated from the following file:

- IngredientListAdapter.cs

5.16 Adapters.JsonAdapter.IngredientListConverter Class Reference

Konwerter JSON dla obiektu [Recipe.IngredientList](#).

Public Member Functions

- override [Recipe.IngredientList Read](#) (ref Utf8JsonReader reader, Type typeToConvert, JsonSerializerOptions options)
Odczytuje obiekt [Recipe.IngredientList](#) z danych JSON.
- override void [Write](#) (Utf8JsonWriter writer, [Recipe.IngredientList](#) value, JsonSerializerOptions options)
Zapisuje obiekt [Recipe.IngredientList](#) do formatu JSON.

5.16.1 Detailed Description

Konwerter JSON dla obiektu [Recipe.IngredientList](#).

5.16.2 Member Function Documentation

5.16.2.1 Read()

```
override Recipe.IngredientList Adapters.JsonAdapter.IngredientListConverter.Read (
    ref Utf8JsonReader reader,
    Type typeToConvert,
    JsonSerializerOptions options)
```

Odczytuje obiekt [Recipe.IngredientList](#) z danych JSON.

Parameters

<i>reader</i>	Reader JSON.
<i>typeToConvert</i>	Typ danych do konwersji.
<i>options</i>	Opcje serializacji JSON.

Returns

Obiekt typu [Recipe.IngredientList](#).

5.16.2.2 Write()

```
override void Adapters.JsonAdapter.IngredientListConverter.Write (
    Utf8JsonWriter writer,
    Recipe.IngredientList value,
    JsonSerializerOptions options)
```

Zapisuje obiekt [Recipe.IngredientList](#) do formatu JSON.

Parameters

<i>writer</i>	Writer JSON.
<i>value</i>	Wartość do serializacji.
<i>options</i>	Opcje serializacji JSON.

The documentation for this class was generated from the following file:

- IngredientListConverter.cs

5.17 ShoppingList.IShoppingList Interface Reference

Definiuje metody dla listy zakupów.

Public Member Functions

- void [AddProduct](#) ([Product.Product](#) product, int quantity)
Dodaje produkt do listy zakupów z określoną ilością.
- void [RemoveProduct](#) ([Product.Product](#) product)
Usuwa produkt z listy zakupów.
- void [MarkAsAdded](#) ([Product.Product](#) product)
Oznacza produkt jako dodany do koszyka.
- void [DisplayProducts](#) ()
Wyświetla produkty na liście zakupów.
- List< [ShoppingItem](#) > [GetItems](#) ()
Zwraca listę pozycji na liście zakupów.
- void [GenerateFromRecipe](#) ([Recipe.Recipe](#) recipe)
Generuje pozycje listy zakupów na podstawie podanego przepisu.

5.17.1 Detailed Description

Definiuje metody dla listy zakupów.

5.17.2 Member Function Documentation

5.17.2.1 AddProduct()

```
void ShoppingList.IShoppingList.AddProduct (
    Product.Product product,
    int quantity)
```

Dodaje produkt do listy zakupów z określoną ilością.

Parameters

<i>product</i>	Produkt do dodania.
<i>quantity</i>	Ilość produktu do dodania.

Implemented in [ShoppingList.Decorators.ShoppingListDecorator](#), and [ShoppingList.ShoppingList](#).

5.17.2.2 DisplayProducts()

```
void ShoppingList.IShoppingList.DisplayProducts ()
```

Wyświetla produkty na liście zakupów.

Implemented in [ShoppingList.Decorators.CategorizedDisplayDecorator](#), [ShoppingList.Decorators.MarkAsAddedDecorator](#), and [ShoppingList.Decorators.ShoppingListDecorator](#).

5.17.2.3 GenerateFromRecipe()

```
void ShoppingList.IShoppingList.GenerateFromRecipe (
    Recipe.Recipe recipe)
```

Generuje pozycje listy zakupów na podstawie podanego przepisu.

Parameters

<i>recipe</i>	Przepis, z którego generowane są pozycje.
---------------	---

Implemented in [ShoppingList.Decorators.ShoppingListDecorator](#), and [ShoppingList.ShoppingList](#).

5.17.2.4 GetItems()

```
List< ShoppingItem > ShoppingList.IShoppingList.GetItems ()
```

Zwraca listę pozycji na liście zakupów.

Returns

Lista pozycji na liście zakupów.

Implemented in [ShoppingList.Decorators.ShoppingListDecorator](#).

5.17.2.5 MarkAsAdded()

```
void ShoppingList.IShoppingList.MarkAsAdded (  
    Product.Product product)
```

Oznacza produkt jako dodany do koszyka.

Parameters

<i>product</i>	Produkt do oznaczenia.
----------------	------------------------

Implemented in [ShoppingList.Decorators.MarkAsAddedDecorator](#), and [ShoppingList.Decorators.ShoppingListDecorator](#).

5.17.2.6 RemoveProduct()

```
void ShoppingList.IShoppingList.RemoveProduct (  
    Product.Product product)
```

Usuwa produkt z listy zakupów.

Parameters

<i>product</i>	Produkt do usunięcia.
----------------	-----------------------

Implemented in [ShoppingList.Decorators.ShoppingListDecorator](#), and [ShoppingList.ShoppingList](#).

The documentation for this interface was generated from the following file:

- IShoppingList.cs

5.18 Adapters.FileSaveAdapter.JsonFileSaveAdapter Class Reference

Adapter do zapisywania danych w formacie JSON.

Public Member Functions

- void [SaveToFile](#) (string filePath, object data)
Zapisuje dane do pliku w formacie JSON w podanej ścieżce pliku.

5.18.1 Detailed Description

Adapter do zapisywania danych w formacie JSON.

5.18.2 Member Function Documentation

5.18.2.1 SaveToFile()

```
void Adapters.FileSaveAdapter.JsonFileSaveAdapter.SaveToFile (
    string filePath,
    object data)
```

Zapisuje dane do pliku w formacie JSON w podanej ścieżce pliku.

Parameters

<i>filePath</i>	Ścieżka pliku, do którego mają zostać zapisane dane.
<i>data</i>	Dane do zapisania w pliku.

Implements [Adapters.FileSaveAdapter.IFileSaveAdapter](#).

The documentation for this class was generated from the following file:

- [JsonFileSaveAdapter.cs](#)

5.19 ShoppingList.Decorators.MarkAsAddedDecorator Class Reference

Dekorator, który oznacza produkt jako dodany do koszyka i wyświetla listę zakupów według kategorii.

Public Member Functions

- [MarkAsAddedDecorator](#) (IShoppingList shoppingList)
Inicjalizuje nową instancję klasy [MarkAsAddedDecorator](#) z podaną listą zakupów.
- override void [MarkAsAdded](#) (Product.Product product)
Oznacza produkt jako dodany do koszyka.
- override void [DisplayProducts](#) ()
Wyświetla produkty na liście zakupów według kategorii.

Public Member Functions inherited from ShoppingList.Decorators.ShoppingListDecorator

- [ShoppingListDecorator](#) ([IShoppingList](#) shoppingList)
Inicjalizuje nową instancję klasy [ShoppingListDecorator](#) z podaną listą zakupów.
- virtual void [AddProduct](#) ([Product.Product](#) product, int quantity)
Dodaje produkt do listy zakupów z określoną ilością.
- virtual void [GenerateFromRecipe](#) ([Recipe.Recipe](#) recipe)
Generuje pozycje listy zakupów na podstawie podanego przepisu.
- virtual void [RemoveProduct](#) ([Product.Product](#) product)
Usuwa produkt z listy zakupów.
- virtual List< [ShoppingItem](#) > [GetItems](#) ()
Zwraca listę pozycji na liście zakupów.

Additional Inherited Members

Protected Attributes inherited from ShoppingList.Decorators.ShoppingListDecorator

- [IShoppingList](#) shoppingList

5.19.1 Detailed Description

Dekorator, który oznacza produkt jako dodany do koszyka i wyświetla listę zakupów według kategorii.

5.19.2 Constructor & Destructor Documentation

5.19.2.1 MarkAsAddedDecorator()

```
ShoppingList.Decorators.MarkAsAddedDecorator.MarkAsAddedDecorator (
    IShoppingList shoppingList)
```

Inicjalizuje nową instancję klasy [MarkAsAddedDecorator](#) z podaną listą zakupów.

Parameters

<i>shoppingList</i>	Lista zakupów do dekorowania.
---------------------	-------------------------------

5.19.3 Member Function Documentation

5.19.3.1 DisplayProducts()

```
override void ShoppingList.Decorators.MarkAsAddedDecorator.DisplayProducts () [virtual]
```

Wyświetla produkty na liście zakupów według kategorii.

Reimplemented from [ShoppingList.Decorators.ShoppingListDecorator](#).

5.19.3.2 MarkAsAdded()

```
override void ShoppingList.Decorators.MarkAsAddedDecorator.MarkAsAdded (
    Product.Product product) [virtual]
```

Oznacza produkt jako dodany do koszyka.

Parameters

<i>product</i>	Produkt do oznaczenia.
----------------	------------------------

Reimplemented from [ShoppingList.Decorators.ShoppingListDecorator](#).

The documentation for this class was generated from the following file:

- MarkAsAddedDecorator.cs

5.20 Meal.Meal Class Reference

Reprezentuje posiłek, zawierający nazwę, datę, typ posiłku oraz listę przepisów.

Classes

- class [MealMemento](#)
Reprezentuje memento posiłku, zawierające nazwę, datę i typ posiłku.

Public Types

- enum [MealType](#) {
 Breakfast , **SnackI** , **Lunch** , **SnackII** ,
 Dinner }
Typy posiłków.

Public Member Functions

- **Meal** ()
Inicjalizuje nową instancję klasy [Meal](#).

Properties

- string **Name** [get, set]
Nazwa posiłku.
- DateTime **Date** [get, set]
Data posiłku.
- [MealType](#) **Type** [get, set]
Typ posiłku.
- List< [Recipe.Recipe](#) > **Recipes** [get, set]
Lista przepisów wchodzących w skład posiłku.

5.20.1 Detailed Description

Reprezentuje posiłek, zawierający nazwę, datę, typ posiłku oraz listę przepisów.

The documentation for this class was generated from the following file:

- Meal.cs

5.21 Meal.MealHistory Class Reference

Reprezentuje historię posiłków, przechowującą mementa posiłków.

Public Member Functions

- **MealHistory** ()
Inicjalizuje nową instancję klasy [MealHistory](#).
- void **Add** (MealMemento memento)
Dodaje memento posiłku do historii.
- MealMemento **Get** (int index)
Pobiera memento posiłku z historii na podstawie indeksu.
- void **Remove** (int index)
Usuwa memento posiłku z historii na podstawie indeksu.
- void **Clear** ()
Czyści historię posiłków.

Properties

- List< MealMemento > **History** [get, set]
Lista mementów posiłków.

5.21.1 Detailed Description

Reprezentuje historię posiłków, przechowującą mementa posiłków.

5.21.2 Member Function Documentation

5.21.2.1 Add()

```
void Meal.MealHistory.Add (  
    MealMemento memento)
```

Dodaje memento posiłku do historii.

Parameters

<i>memento</i>	Memento posiłku do dodania.
----------------	-----------------------------

5.21.2.2 Get()

```
MealMemento Meal.MealHistory.Get (  
    int index)
```

Pobiera memento posiłku z historii na podstawie indeksu.

Parameters

<i>index</i>	Indeks mementa do pobrania.
--------------	-----------------------------

Returns

Memento posiłku.

5.21.2.3 Remove()

```
void Meal.MealHistory.Remove (  
    int index)
```

Usuwa memento posiłku z historii na podstawie indeksu.

Parameters

<i>index</i>	Indeks mementa do usunięcia.
--------------	------------------------------

The documentation for this class was generated from the following file:

- MealHistory.cs

5.22 Meal.Meal.MealMemento Class Reference

Reprezentuje memento posiłku, zawierające nazwę, datę i typ posiłku.

Public Member Functions

- [MealMemento](#) (string name, DateTime date, [MealType](#) type)
Inicjalizuje nową instancję klasy [MealMemento](#) z podanymi wartościami.

Properties

- string **Name** [get, set]
Nazwa posiłku.
- DateTime **Date** [get, set]
Data posiłku.
- MealType **Type** [get, set]
Typ posiłku.

5.22.1 Detailed Description

Reprezentuje memento posiłku, zawierające nazwę, datę i typ posiłku.

5.22.2 Constructor & Destructor Documentation

5.22.2.1 MealMemento()

```
Meal.Meal.MealMemento.MealMemento (  
    string name,  
    DateTime date,  
    MealType type)
```

Inicjalizuje nową instancję klasy [MealMemento](#) z podanymi wartościami.

Parameters

<i>name</i>	Nazwa posiłku.
<i>date</i>	Data posiłku.
<i>type</i>	Typ posiłku.

The documentation for this class was generated from the following file:

- Meal.cs

5.23 Product.Product Class Reference

Reprezentuje produkt z nazwą, kategorią, jednostką miary oraz wartościami odżywczymi.

Public Types

- enum [Categories](#) {
 Fruit , **Vegetable** , **Meat** , **Dairy** ,
 Grain , **Other** }
Kategorie produktów.
- enum [Units](#) { **Grams** , **Milliliters** , **Units** }
Jednostki miary produktów.

Public Member Functions

- **Product** (string name, **Units** unit, **Categories** category, double protein, double fat, double carbohydrates)
*Inicjalizuje nową instancję klasy **Product** z określoną nazwą, jednostką miary, kategorią oraz wartościami odżywczymi.*
- override string **ToString** ()
Zwraca reprezentację tekstową produktu.

Properties

- string **Name** [get, set]
Nazwa produktu.
- **Categories** **Category** [get, set]
Kategoria produktu.
- **Units** **Unit** [get, set]
Jednostka miary produktu.
- double **Protein** [get, set]
Ilość białka w produkcie.
- double **Fat** [get, set]
Ilość tłuszczu w produkcie.
- double **Carbohydrates** [get, set]
Ilość węglowodanów w produkcie.

5.23.1 Detailed Description

Reprezentuje produkt z nazwą, kategorią, jednostką miary oraz wartościami odżywczymi.

5.23.2 Constructor & Destructor Documentation

5.23.2.1 Product()

```
Product.Product.Product (
    string name,
    Units unit,
    Categories category,
    double protein,
    double fat,
    double carbohydrates)
```

Inicjalizuje nową instancję klasy **Product** z określoną nazwą, jednostką miary, kategorią oraz wartościami odżywczymi.

Parameters

<i>name</i>	Nazwa produktu.
<i>unit</i>	Jednostka miary produktu.
<i>category</i>	Kategoria produktu.
<i>protein</i>	Ilość białka w produkcie.
<i>fat</i>	Ilość tłuszczu w produkcie.
<i>carbohydrates</i>	Ilość węglowodanów w produkcie.

5.23.3 Member Function Documentation

5.23.3.1 ToString()

```
override string Product.Product.ToString ()
```

Zwraca reprezentację tekstową produktu.

Returns

Reprezentacja tekstowa produktu.

The documentation for this class was generated from the following file:

- Product.cs

5.24 Adapters.JsonAdapter.ProductAdapter Class Reference

Adapter do serializacji i deserializacji obiektu [Product.Product](#).

Public Member Functions

- [Product.Product Deserialize](#) (ref Utf8JsonReader reader, JsonSerializerOptions options)
Deserializuje dane JSON do obiektu typu [Product.Product](#).
- void [Serialize](#) (Utf8JsonWriter writer, [Product.Product](#) value, JsonSerializerOptions options)
Serializuje obiekt typu [Product.Product](#) do formatu JSON.

Public Member Functions inherited from [Adapters.JsonAdapter.IJsonAdapter< T >](#)

- void [Serialize](#) (Utf8JsonWriter writer, T value, JsonSerializerOptions options)
Serializuje obiekt typu T do formatu JSON.

5.24.1 Detailed Description

Adapter do serializacji i deserializacji obiektu [Product.Product](#).

5.24.2 Member Function Documentation

5.24.2.1 Deserialize()

```
Product.Product Adapters.JsonAdapter.ProductAdapter.Deserialize (  
    ref Utf8JsonReader reader,  
    JsonSerializerOptions options)
```

Deserializuje dane JSON do obiektu typu [Product.Product](#).

Parameters

<i>reader</i>	Reader JSON.
<i>options</i>	Opcje serializacji JSON.

Returns

Obiekt typu [Product.Product](#).

Implements [Adapters.JsonAdapter.IJsonAdapter< T >](#).

5.24.2.2 Serialize()

```
void Adapters.JsonAdapter.ProductAdapter.Serialize (  
    Utf8JsonWriter writer,  
    Product.Product value,  
    JsonSerializerOptions options)
```

Serializuje obiekt typu [Product.Product](#) do formatu JSON.

Parameters

<i>writer</i>	Writer JSON.
<i>value</i>	Wartość do serializacji.
<i>options</i>	Opcje serializacji JSON.

The documentation for this class was generated from the following file:

- ProductAdapter.cs

5.25 Adapters.JsonAdapter.ProductConverter Class Reference

Konwerter JSON dla obiektu [Product.Product](#).

Public Member Functions

- override [Product.Product Read](#) (ref Utf8JsonReader reader, Type typeToConvert, JsonSerializerOptions options)
Odczytuje obiekt [Product.Product](#) z danych JSON.
- override void [Write](#) (Utf8JsonWriter writer, [Product.Product](#) value, JsonSerializerOptions options)
Zapisuje obiekt [Product.Product](#) do formatu JSON.

5.25.1 Detailed Description

Konwerter JSON dla obiektu [Product.Product](#).

5.25.2 Member Function Documentation

5.25.2.1 Read()

```
override Product.Product Adapters.JsonAdapter.ProductConverter.Read (  
    ref Utf8JsonReader reader,  
    Type typeToConvert,  
    JsonSerializerOptions options)
```

Odczytuje obiekt [Product.Product](#) z danych JSON.

Parameters

<i>reader</i>	Reader JSON.
<i>typeToConvert</i>	Typ danych do konwersji.
<i>options</i>	Opcje serializacji JSON.

Returns

Obiekt typu [Product.Product](#).

5.25.2.2 Write()

```
override void Adapters.JsonAdapter.ProductConverter.Write (  
    Utf8JsonWriter writer,  
    Product.Product value,  
    JsonSerializerOptions options)
```

Zapisuje obiekt [Product.Product](#) do formatu JSON.

Parameters

<i>writer</i>	Writer JSON.
<i>value</i>	Wartość do serializacji.
<i>options</i>	Opcje serializacji JSON.

The documentation for this class was generated from the following file:

- ProductConverter.cs

5.26 Program.Program Class Reference

Główna klasa programu.

Static Public Member Functions

- static void **Main** (string[] args)
Punkt wejścia dla aplikacji.

5.26.1 Detailed Description

Główna klasa programu.

The documentation for this class was generated from the following file:

- Program.cs

5.27 Recipe.Recipe Class Reference

Reprezentuje przepis kulinarny, zawierający nazwę, listę składników, instrukcje, czas przygotowania oraz wartości odżywcze.

Public Member Functions

- **Recipe** ()
Inicjalizuje nową instancję klasy [Recipe](#).
- override string **ToString** ()
Zwraca reprezentację tekstową przepisu.

Properties

- string **Name** [get, set]
Nazwa przepisu.
- [IngredientList](#) **Ingredients** [get, set]
Lista składników przepisu.
- List< string > **Instructions** [get, set]
Lista instrukcji przygotowania przepisu.
- TimeSpan **TotalTime** [get, set]
Całkowity czas przygotowania przepisu.
- int **Calories** [get, set]
Liczba kalorii w przepisie.
- double **Protein** [get, set]
Ilość białka w przepisie.
- double **Fat** [get, set]
Ilość tłuszczu w przepisie.
- double **Carbohydrates** [get, set]
Ilość węglowodanów w przepisie.

5.27.1 Detailed Description

Reprezentuje przepis kulinarny, zawierający nazwę, listę składników, instrukcje, czas przygotowania oraz wartości odżywcze.

5.27.2 Member Function Documentation

5.27.2.1 ToString()

```
override string Recipe.Recipe.ToString ()
```

Zwraca reprezentację tekstową przepisu.

Returns

Reprezentacja tekstowa przepisu.

The documentation for this class was generated from the following file:

- [Recipe.cs](#)

5.28 Recipe.RecipeBuilder Class Reference

Klasa budowniczego dla obiektu [Recipe](#), umożliwiającą stopniowe dodawanie składników, instrukcji i innych właściwości.

Public Member Functions

- [RecipeBuilder AddName](#) (string name)
Dodaje nazwę przepisu.
- [RecipeBuilder AddIngredient](#) ([Product.Product](#) product, double amount)
Dodaje składnik do przepisu.
- [RecipeBuilder RemoveIngredient](#) ([Product.Product](#) product)
Usuwa składnik z przepisu.
- [RecipeBuilder AddInstructionStep](#) (string instruction)
Dodaje krok instrukcji do przepisu.
- [RecipeBuilder AddTotalTime](#) (TimeSpan time)
Dodaje całkowity czas przygotowania przepisu.
- [Recipe Build](#) ()
Zwraca zbudowany obiekt [Recipe](#).

5.28.1 Detailed Description

Klasa budowniczego dla obiektu [Recipe](#), umożliwiającą stopniowe dodawanie składników, instrukcji i innych właściwości.

5.28.2 Member Function Documentation

5.28.2.1 AddIngredient()

```
RecipeBuilder Recipe.RecipeBuilder.AddIngredient (  
    Product.Product product,  
    double amount)
```

Dodaje składnik do przepisu.

Parameters

<i>product</i>	Produkt do dodania.
<i>amount</i>	Ilość produktu.

Returns

Obiekt [RecipeBuilder](#).

5.28.2.2 AddInstructionStep()

```
RecipeBuilder Recipe.RecipeBuilder.AddInstructionStep (  
    string instruction)
```

Dodaje krok instrukcji do przepisu.

Parameters

<i>instruction</i>	Krok instrukcji.
--------------------	------------------

Returns

Obiekt [RecipeBuilder](#).

5.28.2.3 AddName()

```
RecipeBuilder Recipe.RecipeBuilder.AddName (  
    string name)
```

Dodaje nazwę przepisu.

Parameters

<i>name</i>	Nazwa przepisu.
-------------	-----------------

Returns

Obiekt [RecipeBuilder](#).

5.28.2.4 AddTotalTime()

```
RecipeBuilder Recipe.RecipeBuilder.AddTotalTime (  
    TimeSpan time)
```

Dodaje całkowity czas przygotowania przepisu.

Parameters

<i>time</i>	Całkowity czas przygotowania.
-------------	-------------------------------

Returns

Obiekt [RecipeBuilder](#).

5.28.2.5 Build()

```
Recipe Recipe.RecipeBuilder.Build ()
```

Zwraca zbudowany obiekt [Recipe](#).

Returns

Zbudowany obiekt [Recipe](#).

5.28.2.6 RemoveIngredient()

```
RecipeBuilder Recipe.RecipeBuilder.RemoveIngredient (  
    Product.Product product)
```

Usuwa składnik z przepisu.

Parameters

<i>product</i>	Produkt do usunięcia.
----------------	-----------------------

Returns

Obiekt [RecipeBuilder](#).

The documentation for this class was generated from the following file:

- [RecipeBuilder.cs](#)

5.29 DB.Record< T > Class Template Reference

Klasa reprezentująca rekord w bazie danych.

Public Member Functions

- [Record](#) (int *Id*, T *obj*)
Inicjalizuje nową instancję klasy [Record<T>](#).
- override string [ToString](#) ()
Zwraca reprezentację tekstową obiektu przechowywanego w rekordzie.

Public Attributes

- T *obj*
Obiekt przechowywany w rekordzie.

Properties

- int *Id* [get, set]
ID rekordu.

5.29.1 Detailed Description

Klasa reprezentująca rekord w bazie danych.

Template Parameters

<i>T</i>	Typ danych przechowywanych w rekordzie.
----------	---

5.29.2 Member Function Documentation

5.29.2.1 Record()

```
DB.Record< T >.Record (  
    int Id,  
    T obj)
```

Inicjalizuje nową instancję klasy [Record<T>](#).

Parameters

<i>Id</i>	ID rekordu.
<i>obj</i>	Obiekt przechowywany w rekordzie.

5.29.2.2 ToString()

```
override string DB.Record< T >.ToString ()
```

Zwraca reprezentację tekstową obiektu przechowywanego w rekordzie.

Returns

Reprezentacja tekstowa obiektu.

The documentation for this class was generated from the following file:

- Record.cs

5.30 ShoppingList.ShoppingItem Class Reference

Reprezentuje pozycję na liście zakupów, która zawiera produkt i jego ilość.

Public Member Functions

- [ShoppingItem](#) ([Product.Product](#) product, int quantity)
Inicjalizuje nową instancję klasy [ShoppingItem](#) z określonym produktem i ilością.

Properties

- [Product.Product](#) **Product** [get, set]
Produkt na liście zakupów.
- int **Quantity** [get, set]
Ilość produktu.
- bool **IsAdded** [get, set]
Określa, czy produkt został dodany do koszyka.

5.30.1 Detailed Description

Reprezentuje pozycję na liście zakupów, która zawiera produkt i jego ilość.

5.30.2 Constructor & Destructor Documentation

5.30.2.1 ShoppingItem()

```
ShoppingList.ShoppingItem.ShoppingItem (  
    Product.Product product,  
    int quantity)
```

Inicjalizuje nową instancję klasy [ShoppingItem](#) z określonym produktem i ilością.

Parameters

<i>product</i>	Produkt na liście zakupów.
<i>quantity</i>	Ilość produktu.

The documentation for this class was generated from the following file:

- ShoppingItem.cs

5.31 ShoppingList.ShoppingList Class Reference

Reprezentuje listę zakupów, która może przechowywać produkty i ich ilości.

Public Member Functions

- void [AddProduct](#) ([Product.Product](#) product, int quantity)
Dodaje produkt do listy zakupów z określoną ilością. Jeśli produkt już istnieje na liście, jego ilość zostaje zwiększona.
- void [GenerateFromRecipe](#) ([Recipe.Recipe](#) recipe)
Generuje pozycje listy zakupów na podstawie podanego przepisu.
- void [RemoveProduct](#) ([Product.Product](#) product)
Usuwa produkt z listy zakupów.

Public Member Functions inherited from [ShoppingList.IShoppingList](#)

- void [MarkAsAdded](#) ([Product.Product](#) product)
Oznacza produkt jako dodany do koszyka.
- void [DisplayProducts](#) ()
Wyświetla produkty na liście zakupów.
- List< [ShoppingItem](#) > [GetItems](#) ()
Zwraca listę pozycji na liście zakupów.

5.31.1 Detailed Description

Reprezentuje listę zakupów, która może przechowywać produkty i ich ilości.

5.31.2 Member Function Documentation

5.31.2.1 AddProduct()

```
void ShoppingList.ShoppingList.AddProduct (
    Product.Product product,
    int quantity)
```

Dodaje produkt do listy zakupów z określoną ilością. Jeśli produkt już istnieje na liście, jego ilość zostaje zwiększona.

Parameters

<i>product</i>	Produkt do dodania.
<i>quantity</i>	Ilość produktu do dodania.

Implements [ShoppingList.IShoppingList](#).

5.31.2.2 GenerateFromRecipe()

```
void ShoppingList.ShoppingList.GenerateFromRecipe (  
    Recipe.Recipe recipe)
```

Generuje pozycje listy zakupów na podstawie podanego przepisu.

Parameters

<i>recipe</i>	Przepis, z którego generowane są pozycje.
---------------	---

Implements [ShoppingList.IShoppingList](#).

5.31.2.3 RemoveProduct()

```
void ShoppingList.ShoppingList.RemoveProduct (  
    Product.Product product)
```

Usuwa produkt z listy zakupów.

Parameters

<i>product</i>	Produkt do usunięcia.
----------------	-----------------------

Implements [ShoppingList.IShoppingList](#).

The documentation for this class was generated from the following file:

- ShoppingList.cs

5.32 ShoppingList.Decorators.ShoppingListDecorator Class Reference

Abstrakcyjna klasa dekoratora dla listy zakupów, implementująca interfejs [IShoppingList](#).

Public Member Functions

- [ShoppingListDecorator](#) ([IShoppingList](#) shoppingList)
Inicjalizuje nową instancję klasy [ShoppingListDecorator](#) z podaną listą zakupów.
- virtual void [AddProduct](#) ([Product.Product](#) product, int quantity)
Dodaje produkt do listy zakupów z określoną ilością.
- virtual void [GenerateFromRecipe](#) ([Recipe.Recipe](#) recipe)
Generuje pozycje listy zakupów na podstawie podanego przepisu.
- virtual void [RemoveProduct](#) ([Product.Product](#) product)
Usuwa produkt z listy zakupów.
- virtual void [MarkAsAdded](#) ([Product.Product](#) product)
Oznacza produkt jako dodany do koszyka.
- virtual void [DisplayProducts](#) ()
Wyświetla produkty na liście zakupów.
- virtual List< [ShoppingItem](#) > [GetItems](#) ()
Zwraca listę pozycji na liście zakupów.

Protected Attributes

- [IShoppingList](#) shoppingList

5.32.1 Detailed Description

Abstrakcyjna klasa dekoratora dla listy zakupów, implementująca interfejs [IShoppingList](#).

5.32.2 Constructor & Destructor Documentation

5.32.2.1 ShoppingListDecorator()

```
ShoppingList.Decorators.ShoppingListDecorator.ShoppingListDecorator (
    IShoppingList shoppingList)
```

Inicjalizuje nową instancję klasy [ShoppingListDecorator](#) z podaną listą zakupów.

Parameters

<i>shoppingList</i>	Lista zakupów do dekorowania.
---------------------	-------------------------------

5.32.3 Member Function Documentation

5.32.3.1 AddProduct()

```
virtual void ShoppingList.Decorators.ShoppingListDecorator.AddProduct (
    Product.Product product,
    int quantity) [virtual]
```

Dodaje produkt do listy zakupów z określoną ilością.

Parameters

<i>product</i>	Produkt do dodania.
<i>quantity</i>	Ilość produktu do dodania.

Implements [ShoppingList.IShoppingList](#).

5.32.3.2 DisplayProducts()

```
virtual void ShoppingList.Decorators.ShoppingListDecorator.DisplayProducts () [virtual]
```

Wyświetla produkty na liście zakupów.

Implements [ShoppingList.IShoppingList](#).

Reimplemented in [ShoppingList.Decorators.CategorizedDisplayDecorator](#), and [ShoppingList.Decorators.MarkAsAddedDecorator](#).

5.32.3.3 GenerateFromRecipe()

```
virtual void ShoppingList.Decorators.ShoppingListDecorator.GenerateFromRecipe (  
    Recipe.Recipe recipe) [virtual]
```

Generuje pozycje listy zakupów na podstawie podanego przepisu.

Parameters

<i>recipe</i>	Przepis, z którego generowane są pozycje.
---------------	---

Implements [ShoppingList.IShoppingList](#).

5.32.3.4 GetItems()

```
virtual List< ShoppingItem > ShoppingList.Decorators.ShoppingListDecorator.GetItems () [virtual]
```

Zwraca listę pozycji na liście zakupów.

Returns

Lista pozycji na liście zakupów.

Implements [ShoppingList.IShoppingList](#).

5.32.3.5 MarkAsAdded()

```
virtual void ShoppingList.Decorators.ShoppingListDecorator.MarkAsAdded (  
    Product.Product product) [virtual]
```

Oznacza produkt jako dodany do koszyka.

Parameters

<i>product</i>	Produkt do oznaczenia.
----------------	------------------------

Implements [ShoppingList.IShoppingList](#).

Reimplemented in [ShoppingList.Decorators.MarkAsAddedDecorator](#).

5.32.3.6 RemoveProduct()

```
virtual void ShoppingList.Decorators.ShoppingListDecorator.RemoveProduct (  
    Product.Product product) [virtual]
```

Usuwa produkt z listy zakupów.

Parameters

<i>product</i>	Produkt do usunięcia.
----------------	-----------------------

Implements [ShoppingList.IShoppingList](#).

The documentation for this class was generated from the following file:

- [ShoppingListDecorator.cs](#)

5.33 Adapters.FileSaveAdapter.TxtFileSaveAdapter Class Reference

Adapter do zapisywania danych w formacie tekstowym.

Public Member Functions

- void [SaveToFile](#) (string filePath, object data)
Zapisuje dane do pliku w formacie tekstowym w podanej ścieżce pliku.

5.33.1 Detailed Description

Adapter do zapisywania danych w formacie tekstowym.

5.33.2 Member Function Documentation

5.33.2.1 SaveToFile()

```
void Adapters.FileSaveAdapter.TxtFileSaveAdapter.SaveToFile (  
    string filePath,  
    object data)
```

Zapisuje dane do pliku w formacie tekstowym w podanej ścieżce pliku.

Parameters

<i>filePath</i>	Ścieżka pliku, do którego mają zostać zapisane dane.
<i>data</i>	Dane do zapisania w pliku.

Implements [Adapters.FileSaveAdapter.IFileSaveAdapter](#).

The documentation for this class was generated from the following file:

- TxtFileSaveAdapter.cs

Index

- Adapters, [7](#)
- Adapters.FileSaveAdapter, [7](#)
- Adapters.FileSaveAdapter.FileSaveManager, [21](#)
 - SaveToFile, [22](#)
- Adapters.FileSaveAdapter.IFileSaveAdapter, [26](#)
 - SaveToFile, [26](#)
- Adapters.FileSaveAdapter.JsonFileSaveAdapter, [35](#)
 - SaveToFile, [36](#)
- Adapters.FileSaveAdapter.TxtFileSaveAdapter, [56](#)
 - SaveToFile, [56](#)
- Adapters.JsonAdapter, [7](#)
- Adapters.JsonAdapter.IJsonAdapter< T >, [27](#)
 - Deserialize, [27](#)
 - Serialize, [27](#)
- Adapters.JsonAdapter.IngredientListAdapter, [31](#)
 - Deserialize, [31](#)
 - Serialize, [31](#)
- Adapters.JsonAdapter.IngredientListConverter, [32](#)
 - Read, [32](#)
 - Write, [33](#)
- Adapters.JsonAdapter.ProductAdapter, [43](#)
 - Deserialize, [43](#)
 - Serialize, [44](#)
- Adapters.JsonAdapter.ProductConverter, [44](#)
 - Read, [45](#)
 - Write, [46](#)
- Add
 - Meal.MealHistory, [39](#)
 - Recipe.IngredientList, [29](#)
- AddIngredient
 - Recipe.RecipeBuilder, [48](#)
- AddIngredientCommand
 - Recipe.Commands.AddIngredientCommand, [11](#)
- AddInstructionStep
 - Recipe.RecipeBuilder, [48](#)
- AddName
 - Recipe.RecipeBuilder, [49](#)
- AddProduct
 - ShoppingList.Decorators.ShoppingListDecorator, [54](#)
 - ShoppingList.IShoppingList, [34](#)
 - ShoppingList.ShoppingList, [52](#)
- AddRecord
 - DB.DatabaseConnection< T >, [18](#)
 - DB.IDatabaseConnection< T >, [24](#)
- AddTotalTime
 - Recipe.RecipeBuilder, [49](#)
- Build
 - Recipe.RecipeBuilder, [49](#)
- CategorizedDisplayDecorator
 - ShoppingList.Decorators.CategorizedDisplayDecorator, [13](#)
- DatabaseConnection
 - DB.DatabaseConnection< T >, [18](#)
- DB, [8](#)
- DB.ConnectionManager, [14](#)
 - getConnection< T >, [15](#)
 - getInstance, [16](#)
- DB.Database< T >, [16](#)
 - getInstance, [17](#)
- DB.DatabaseConnection< T >, [17](#)
 - AddRecord, [18](#)
 - DatabaseConnection, [18](#)
 - DeleteRecord, [18](#)
 - GetAllRecords, [18](#)
 - GetLastRecord, [19](#)
 - GetRecordById, [19](#)
 - GetRecordByName, [19](#)
 - ShowAllRecords, [20](#)
 - UpdateRecord, [20](#)
- DB.IDatabaseConnection< T >, [23](#)
 - AddRecord, [24](#)
 - DeleteRecord, [24](#)
 - GetAllRecords, [24](#)
 - GetLastRecord, [24](#)
 - GetRecordById, [25](#)
 - GetRecordByName, [25](#)
 - ShowAllRecords, [25](#)
 - UpdateRecord, [25](#)
- DB.Record< T >, [50](#)
 - Record, [50](#)
 - ToString, [51](#)
- DeleteIngredientCommand
 - Recipe.Commands.DeleteIngredientCommand, [21](#)
- DeleteRecord
 - DB.DatabaseConnection< T >, [18](#)
 - DB.IDatabaseConnection< T >, [24](#)
- Deserialize
 - Adapters.JsonAdapter.IJsonAdapter< T >, [27](#)
 - Adapters.JsonAdapter.IngredientListAdapter, [31](#)
 - Adapters.JsonAdapter.ProductAdapter, [43](#)
- DisplayProducts
 - ShoppingList.Decorators.CategorizedDisplayDecorator, [13](#)
 - ShoppingList.Decorators.MarkAsAddedDecorator, [37](#)
 - ShoppingList.Decorators.ShoppingListDecorator, [55](#)

- ShoppingList.IShoppingList, 34
- Execute
 - Recipe.Commands.AddIngredientCommand, 12
 - Recipe.Commands.DeleteIngredientCommand, 21
 - Recipe.Commands.ICommand, 23
- ExecuteCommand
 - Recipe.Commands.CommandManager, 14
- GenerateFromRecipe
 - ShoppingList.Decorators.ShoppingListDecorator, 55
 - ShoppingList.IShoppingList, 34
 - ShoppingList.ShoppingList, 53
- Get
 - Meal.MealHistory, 40
- GetAllRecords
 - DB.DatabaseConnection< T >, 18
 - DB.IDatabaseConnection< T >, 24
- getConnection< T >
 - DB.ConnectionManager, 15
- GetEnumerator
 - Recipe.IngredientList, 29
- getInstance
 - DB.ConnectionManager, 16
 - DB.Database< T >, 17
- GetItems
 - ShoppingList.Decorators.ShoppingListDecorator, 55
 - ShoppingList.IShoppingList, 34
- GetLastRecord
 - DB.DatabaseConnection< T >, 19
 - DB.IDatabaseConnection< T >, 24
- GetRecordById
 - DB.DatabaseConnection< T >, 19
 - DB.IDatabaseConnection< T >, 25
- GetRecordByName
 - DB.DatabaseConnection< T >, 19
 - DB.IDatabaseConnection< T >, 25
- GetTotalCarbohydrates
 - Recipe.IngredientList, 30
- GetTotalFat
 - Recipe.IngredientList, 30
- GetTotalProtein
 - Recipe.IngredientList, 30
- Ingredient
 - Recipe.Ingredient, 28
- MarkAsAdded
 - ShoppingList.Decorators.MarkAsAddedDecorator, 37
 - ShoppingList.Decorators.ShoppingListDecorator, 55
 - ShoppingList.IShoppingList, 35
- MarkAsAddedDecorator
 - ShoppingList.Decorators.MarkAsAddedDecorator, 37
- Meal, 8
 - Meal.Meal, 38
 - Meal.Meal.MealMemento, 40
 - MealMemento, 41
 - Meal.MealHistory, 39
 - Add, 39
 - Get, 40
 - Remove, 40
 - MealMemento
 - Meal.Meal.MealMemento, 41
 - Product, 8
 - Product.Product, 42
 - Product.Product, 41
 - Product, 42
 - ToString, 43
 - Program, 8
 - Program.Program, 46
 - Read
 - Adapters.JsonAdapter.IngredientListConverter, 32
 - Adapters.JsonAdapter.ProductConverter, 45
 - Recipe, 9
 - Recipe.Commands, 9
 - Recipe.Commands.AddIngredientCommand, 11
 - AddIngredientCommand, 11
 - Execute, 12
 - Undo, 12
 - Recipe.Commands.CommandManager, 13
 - ExecuteCommand, 14
 - Recipe.Commands.DeleteIngredientCommand, 20
 - DeleteIngredientCommand, 21
 - Execute, 21
 - Undo, 21
 - Recipe.Commands.ICommand, 22
 - Execute, 23
 - Undo, 23
 - Recipe.Ingredient, 28
 - Ingredient, 28
 - Recipe.IngredientList, 29
 - Add, 29
 - GetEnumerator, 29
 - GetTotalCarbohydrates, 30
 - GetTotalFat, 30
 - GetTotalProtein, 30
 - Remove, 30
 - Recipe.Recipe, 47
 - ToString, 47
 - Recipe.RecipeBuilder, 48
 - AddIngredient, 48
 - AddInstructionStep, 48
 - AddName, 49
 - AddTotalTime, 49
 - Build, 49
 - RemoveIngredient, 49
 - Record
 - DB.Record< T >, 50
 - Remove
 - Meal.MealHistory, 40
 - Recipe.IngredientList, 30

- RemoveIngredient
 - Recipe.RecipeBuilder, [49](#)
- RemoveProduct
 - ShoppingList.Decorators.ShoppingListDecorator, [56](#)
 - ShoppingList.IShoppingList, [35](#)
 - ShoppingList.ShoppingList, [53](#)
- SaveToFile
 - Adapters.FileSaveAdapter.FileSaveManager, [22](#)
 - Adapters.FileSaveAdapter.IFileSaveAdapter, [26](#)
 - Adapters.FileSaveAdapter.JsonFileSaveAdapter, [36](#)
 - Adapters.FileSaveAdapter.TxtFileSaveAdapter, [56](#)
- Serialize
 - Adapters.JsonAdapter.IJsonAdapter< T >, [27](#)
 - Adapters.JsonAdapter.IngredientListAdapter, [31](#)
 - Adapters.JsonAdapter.ProductAdapter, [44](#)
- ShoppingItem
 - ShoppingList.ShoppingItem, [51](#)
- ShoppingList, [9](#)
- ShoppingList.Decorators, [9](#)
- ShoppingList.Decorators.CategorizedDisplayDecorator, [12](#)
 - CategorizedDisplayDecorator, [13](#)
 - DisplayProducts, [13](#)
- ShoppingList.Decorators.MarkAsAddedDecorator, [36](#)
 - DisplayProducts, [37](#)
 - MarkAsAdded, [37](#)
 - MarkAsAddedDecorator, [37](#)
- ShoppingList.Decorators.ShoppingListDecorator, [53](#)
 - AddProduct, [54](#)
 - DisplayProducts, [55](#)
 - GenerateFromRecipe, [55](#)
 - GetItems, [55](#)
 - MarkAsAdded, [55](#)
 - RemoveProduct, [56](#)
 - ShoppingListDecorator, [54](#)
- ShoppingList.IShoppingList, [33](#)
 - AddProduct, [34](#)
 - DisplayProducts, [34](#)
 - GenerateFromRecipe, [34](#)
 - GetItems, [34](#)
 - MarkAsAdded, [35](#)
 - RemoveProduct, [35](#)
- ShoppingList.ShoppingItem, [51](#)
 - ShoppingItem, [51](#)
- ShoppingList.ShoppingList, [52](#)
 - AddProduct, [52](#)
 - GenerateFromRecipe, [53](#)
 - RemoveProduct, [53](#)
- ShoppingListDecorator
 - ShoppingList.Decorators.ShoppingListDecorator, [54](#)
- ShowAllRecords
 - DB.DatabaseConnection< T >, [20](#)
 - DB.IDatabaseConnection< T >, [25](#)
- ToString
- DB.Record< T >, [51](#)
- Product.Product, [43](#)
- Recipe.Recipe, [47](#)
- Undo
 - Recipe.Commands.AddIngredientCommand, [12](#)
 - Recipe.Commands.DeleteIngredientCommand, [21](#)
 - Recipe.Commands.ICommand, [23](#)
- UpdateRecord
 - DB.DatabaseConnection< T >, [20](#)
 - DB.IDatabaseConnection< T >, [25](#)
- Write
 - Adapters.JsonAdapter.IngredientListConverter, [33](#)
 - Adapters.JsonAdapter.ProductConverter, [46](#)