

Różnica między List a ObservableCollection

► 1. List<T>

To zwykła lista.

Zalety:

- prosta,
- szybka,
- dobra do przechowywania danych.

Wada:

! Nie informuje interfejsu użytkownika (UI), że się zmieniła.
Jeśli coś dodasz do Listy, ListView o tym nie wie.

Dlatego trzeba "wymusić" odświeżenie.

```
Lista.ItemsSource = null;  
Lista.ItemsSource = _zakupy;
```

► 2. ObservableCollection<T>

To specjalna lista, która **wysyła powiadomienia do UI**.

Gdy coś dodasz:

- UI zauważy zmianę,
- ListView sam się odświeży,
- nie musisz niczego resetować ręcznie.

Najprostsza definicja:

ObservableCollection to lista, która potrafi sama powiadomić interfejs, że się zmieniła.

Dlatego ListView/CollectionView idealnie z nią współpracuje.

● Zamiana List na ObservableCollection

Nasza aplikacja korzysta z List<string>.
Zamienimy ją na ObservableCollection<string>.

► Kod przed:

```
List<string> _zakupy;
```

► Kod po:

```
using System.Collections.ObjectModel;  
  
ObservableCollection<string> _zakupy;
```

Inicjalizacja:

```
_zakupy = new ObservableCollection<string>  
{  
    "Mleko", "Masło", "Jajka"  
};
```

ListView nadal działa identycznie:

```
Lista.ItemsSource = _zakupy;
```

Nic więcej nie musisz zmieniać.

● Usunięcie metody odświeżania

ObservableCollection sama powiadamia UI o zmianie, więc:

- ✗ metoda `odswiez()` jest niepotrzebna
- ✗ nie trzeba ustawiać `ItemsSource = null`

Wyrzucamy:

```
void odswiez()  
{  
    Lista.ItemsSource = null;  
    Lista.ItemsSource = _zakupy;  
}
```

W metodzie `Dodaj_Clicked` usuwamy:

```
odswiez();
```

Zostaje samo:

```
void Dodaj_Clicked(object sender, EventArgs e)  
{  
    var nowy = Nowy.Text?.Trim();  
    if (string.IsNullOrEmpty(nowy)) return;  
  
    _zakupy.Add(nowy);    // Teraz to automatycznie odświeży listę
```

```
        Nowy.Text = "";  
    }
```

✓ Efekt końcowy

- kod jest krótszy,
- nie trzeba ręcznie odświeżać listy,
- aplikacja jest bardziej profesjonalna,
- rozumiesz *dlaczego* coś działa (a nie tylko „bo tak trzeba”).

■ Gotowy kod po zmianach

```
using System.Collections.ObjectModel;  
  
namespace mobilne4p8  
{  
    public partial class MainPage : ContentPage  
    {  
        ObservableCollection<string> _zakupy;  
  
        public MainPage()  
        {  
            InitializeComponent();  
            _zakupy = new ObservableCollection<string>  
            {  
                "Mleko", "Masło", "Jajka"  
            };  
  
            Lista.ItemsSource = _zakupy;  
        }  
  
        void Dodaj_Clicked(object sender, EventArgs e)  
        {  
            var nowy = Nowy.Text?.Trim();  
            if (string.IsNullOrEmpty(nowy)) return;  
  
            _zakupy.Add(nowy);  
  
            Nowy.Text = "";  
        }  
  
        void Lista_ItemSelected(object sender, SelectedItemChangedEventArgs e)  
        {  
            var wybor = e.SelectedItem as string;  
            Informacja.Text = wybor == null ?  
                "nic nie wybrano" :  
                $"Wybrano: {wybor}";  
        }  
    }  
}
```