



ListView и работа с данными

ListView

Последнее обновление: 27.03.2017

Помогите им найти вашу
компанию в Google!

НАЧНИТЕ СЕЙЧАС

Компенсируем
до \$60

Google AdWords

Для отображения списка объектов в Xamarin.Forms применяется элемент **ListView**. Он позволяет отображать как наборы довольно простых данных, например, строк, так и более сложные объекты, позволяя кастомизировать их отображение.

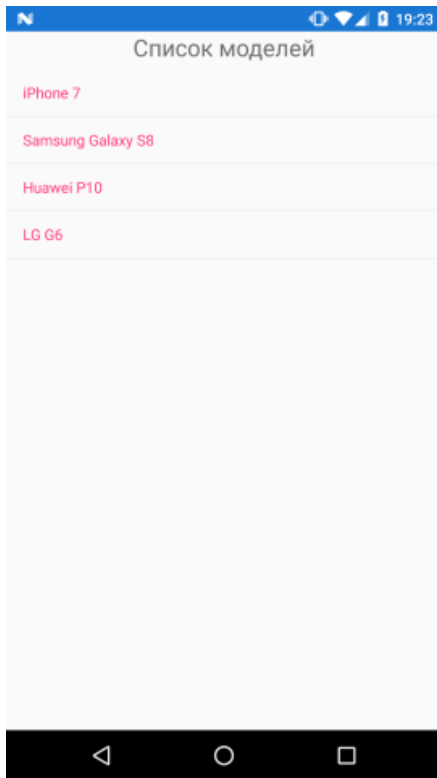
Для привязки набора объектов в ListView определено свойство **ItemsSource**. Например, выведем массив строк:

```
using Xamarin.Forms;

namespace HelloApp
{
    public partial class MainPage : ContentPage
    {
        public MainPage()
        {
            Label header = new Label
            {
                Text = "Список моделей",
                FontSize = Device.GetNamedSize(NamedSize.Large, typeof(Label)),
                HorizontalOptions = LayoutOptions.Center
            };

            string[] phones = new string[] { "iPhone 7", "Samsung Galaxy S8", "Huawei P10", "LG G6" };

            ListView listView = new ListView();
            // определяем источник данных
            listView.ItemsSource = phones;
            this.Content = new StackLayout { Children = { header, listView } };
        }
    }
}
```



Важно, что привязке здесь идет именно к массиву строк, а не к списку `List<string>`.

Аналогичный массив в XAML мы могли бы вывести следующим образом:

```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
  xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
  x:Class="HelloApp.MainPage">
  <ContentPage.Resources>
    <ResourceDictionary>
      <x:Array x:Key="phones" Type="{x:Type x:String}">
        <x:String>iPhone 7</x:String>
        <x:String>HP Elite z3</x:String>
        <x:String>Huawei P10</x:String>
        <x:String>Xiaomi Mi6</x:String>
      </x:Array>
    </ResourceDictionary>
  </ContentPage.Resources>
  <StackLayout>
    <Label Text="Список моделей" HorizontalOptions="Center" FontSize="Large" />
    <ListView x:Name="phonesList" ItemsSource="{StaticResource Key=phones}" />
  </StackLayout>
</ContentPage>
```

Выбор элемента

ListView выбор одного элемента одномоментно. По умолчанию выбранный элемент выделяется. Когда пользователь нажимает на элемент списка, у ListView срабатывают два события: `ItemTapped` и `ItemSelected`. Между ними есть различия

Повторное нажатие на один и тот же элемент не вызовет повторного события `ItemSelected`, так как элемент остается выбранным. А вот событие `ItemTapped` будет срабатывать именно столько раз сколько пользователь нажал на него, даже повторно. Также событие `ItemSelected` будет вызвано, если с элемента будет снято выделение.

Обработаем выделение элемента. Для этого определим следующий код в XAML:

```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
  xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
  x:Class="HelloApp.MainPage">
  <StackLayout>
    <Label x:Name="selected" HorizontalOptions="Center" FontSize="Large" />
    <ListView x:Name="phonesList" ItemSelected="phonesList_ItemSelected" />
  </StackLayout>
</ContentPage>
```

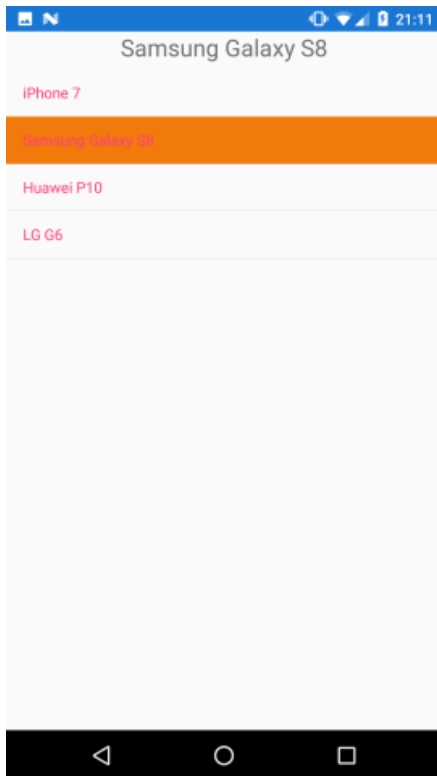
А в файле кода `MainPage.xaml.cs` определим источник данных и обработчик события `ItemSelected`:

```
using Xamarin.Forms;

namespace HelloApp
{
    public partial class MainPage : ContentPage
    {
        public MainPage()
        {
            InitializeComponent();
            string[] phones = new string[] { "iPhone 7", "Samsung Galaxy S8", "Huawei P10", "LG G6" };
            phonesList.ItemsSource = phones;
        }
        private void phonesList_ItemSelected(object sender, SelectedItemChangedEventArgs e)
        {
            if(e.SelectedItem!=null)
                selected.Text = e.SelectedItem.ToString();
        }
    }
}
```

При выборе элемента мы можем получить выбранный элемент через свойство `SelectedItem` объекта `SelectedItemChangedEventArgs`, который передается в качестве параметра. Так как событие `ItemSelected` может срабатывать и при снятии выделения элемента, то в обработчике необходимо вначале проверить значение `e.SelectedItem` на `null`.

Поскольку список содержит массив строк, то каждый его элемент представляет строку, которую мы можем получить с помощью метода `e.SelectedItem.ToString()`.



Как можно увидеть на скриншоте, выбранный элемент выделяется цветом. Однако выделение на Android оранжевым цветом элементов списка, которые имеют по умолчанию розовые буквы, наверное, не самое удачное дизайнерское решение. Кроме того, возможно выделение в принципе неприемлемо, и в этом случае мы его можем сразу же снять после выбора элемента:

```
private void phonesList_ItemSelected(object sender, SelectedItemChangedEventArgs e)
{
    if(e.SelectedItem!=null)
        selected.Text = e.SelectedItem.ToString();
    ((ListView)sender).SelectedItem = null;
}
```

Привязка к выбранному элементу

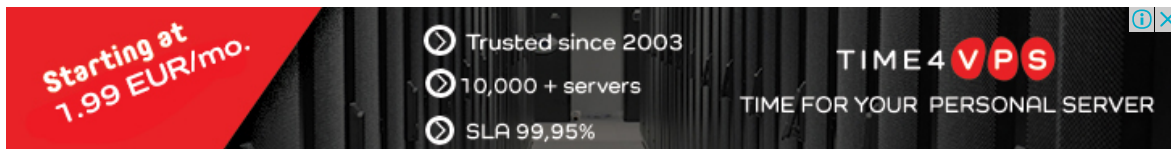
В примере выше мы вполне могли бы обойтись без обработки события и просто установить привязку элемента `Label` к выбранному объекту списка:

```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    x:Class="HelloApp.MainPage">
```

```

<ContentPage.Resources>
  <ResourceDictionary>
    <x:Array x:Key="phones" Type="{x:Type x:String}">
      <x:String>iPhone 7</x:String>
      <x:String>HP Elite z3</x:String>
      <x:String>Huawei P10</x:String>
      <x:String>Xiaomi Mi6</x:String>
    </x:Array>
  </ResourceDictionary>
</ContentPage.Resources>
<StackLayout>
  <Label Text="{Binding Source={x:Reference Name=phonesList}, Path=SelectedItem}"
    HorizontalOptions="Center" FontSize="Large" />
  <ListView x:Name="phonesList" ItemsSource="{StaticResource phones}" />
</StackLayout>
</ContentPage>

```



[Назад](#) [Содержание](#) [Вперед](#)



12 Комментариев metanit.com

1 Войти ▾

♥ Рекомендовать Поделиться

Лучшее в начале ▾



Присоединиться к обсуждению...

ВОЙТИ С ПОМОЩЬЮ

ИЛИ ЧЕРЕЗ DISQUS ?

Имя



Alexi Sailor • 3 месяца назад

Подскажите пожалуйста, почему ругается на конструкцию selected.Text ? Выдаёт "Имя selected не существует в текущем контексте"

^ | ▾ • Ответить • Поделиться ›



Metanit Модератор ➔ Alexi Sailor • 3 месяца назад

на странице должно быть Label x:Name="selected"

^ | ▾ • Ответить • Поделиться ›



Snappy Lis • 6 месяцев назад

Как изменить цвет выделения элемента ListView? Это реализуется на каждой платформе или существует

единое решение?

^ | v • Ответить • Поделиться ›



semen semen → Snappy Lis • месяц назад

`e.selecteditem=null;`

^ | v • Ответить • Поделиться ›



Игорь • год назад

Совет по ListView. Используйте Класс ObservableCollection<T>. Т.к. при удалении из ListView элемента(например). Ваша коллекция должна иметь возможность без Exceptions обновится.

^ | v • Ответить • Поделиться ›



Makarkin & Partners / Макаркин • год назад

что то на `public List<Language> Langs Language` - ругается

^ | v • Ответить • Поделиться ›



Metanit Модератор → Makarkin & Partners / Макаркин • год назад

Проверьте, что класс Language находится в одном пространстве имен со StartPage

^ | v • Ответить • Поделиться ›



Evgeniy Ryabchenko • 2 года назад

Доброй ночи. Подскажите, как можно реализовать обработку короткого и длинного клика на любом элементе? ListView хоть и имеет подобные возможности, но, как мне кажется, нелогично использовать подобный тип, если элемент в единственном экземпляре(кнопка, например). Gesture Listener пробовал реализовать, но не получилось. В каком из классов его необходимо реализовывать? При его реализации в MainActivity.cs Андройда вылетает ошибка при запуске приложения.

^ | v • Ответить • Поделиться ›



Мария • 2 года назад

Добрый день!

Подскажите пожалуйста есть ли какая то хитрость для "устранения" следующего свойства списка: когда кликаешь на объект списка и например по этому клику переходим на другое окно приложения, то при нажатие кнопки "Назад", выбранный элемент списка не активен для клика(т.е. он уже выбран), а хочется чтобы можно было снова на него нажать.

Есть ли решение такой проблемы?

А также хотелось бы уточнить: возможно ли с помощью кода изменить стиль выбранного элемента списка?

Спасибо заранее.

^ | v • Ответить • Поделиться ›



Alexandr Grebenyuk • 3 года назад

Как применить кастомные стили для выбранного элемента списка с помощью XAML?

^ | v • Ответить • Поделиться ›



Евгений Зарожный • 3 года назад

А как сделать обработчик если делать представление через код, а не через xaml?

^ | v • Ответить • Поделиться ›



Metanit Модератор → Евгений Зарожный • 3 года назад

`langsList.ItemSelected+=langsList_ItemSelected;`

^ | v • Ответить • Поделиться ›

ТАКЖЕ НА METANIT.COM

C# и .NET | Соккрытие методов

3 комментариев • 2 месяца назад



Везнич — пример, чтоб видно разницу между new и

`Avatar override class Water { public virtual void Drink() { Console.WriteLine("water life"); } } class Tea : Water { public`

Go | Соответствие интерфейсу

1 комментарий • 24 дня назад

Go | Каналы

2 комментариев • 9 дней назад



Metanit — поправил

Аватар

Vue.js | Навигация и ссылки

1 комментарий • 2 месяца назад

1 комментарий • 24 дня назад

1 комментарий • 3 месяца назад



Пу — Еще давай. Не останавливайся, пожалуйста. Это гениальный язык. Он, конечно, не божественен, как шарп, но гениальный.



Delirium4Dude — Огромное спасибо тебе за твои труды! Один из самых толковых мануалов в Рунете

[Подписаться](#)[Добавь Disqus на свой сайт](#)[Добавить Disqus](#)[Добавить](#)[Конфиденциальность](#)

-34%

-70%

-30%

-29%

-49%

-39%

-14%

-50%

-39%

Farfetch UK Limited...

[Вконтакте](#) | [Twitter](#) | [Google+](#) | [Канал сайта на youtube](#) | [Помощь сайту](#)

Copyright © metanit.com, 2012-2017. Все права защищены.

