



## TableView

Последнее обновление: 13.06.2016



### 1 Попробуй SE Ranking

Проверка позиций сайта от 1 коп, анализ конкурентов и SEO аудит сайта. 14 дней бесплатно! [seranking.ru](http://seranking.ru)

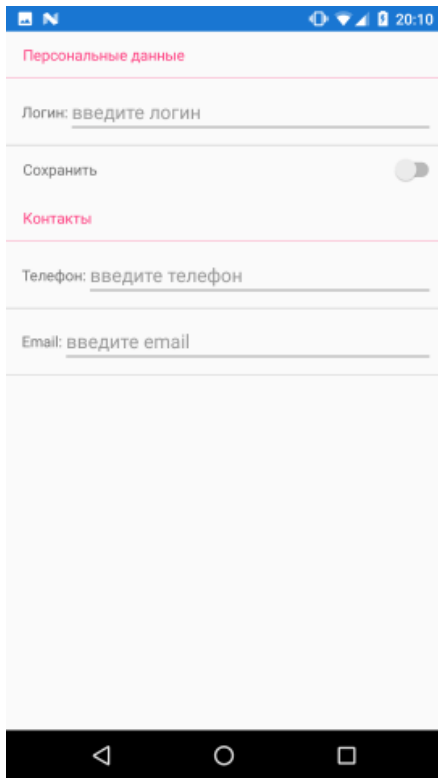
### 2 Минск Москва от 35 руб.

Бронь билетов на автобус. Ежедневные рейсы. Звоните. [minsk-moskva.by](http://minsk-moskva.by)

TableView позволяет создавать таблицы с различным содержимым, при этом в таблицах могут размещаться и другие элементы управления. TableView может быть полезен для отображения списка различных настроек, вывод данных в виде формы, либо для построчного отображения данных.

Элементы в TableView организованы в секции (элементы TableSection). Корневым элементов в TableView является элемент **TableRoot**, который инкапсулирует в себе все секции:

```
public partial class MainPage : ContentPage
{
    public MainPage()
    {
        this.Content = new TableView
        {
            Intent = TableIntent.Form,
            Root = new TableRoot("Ввод данных")
            {
                new TableSection("Персональные данные")
                {
                    new EntryCell
                    {
                        Label = "Логин:",
                        Placeholder = "введите логин",
                        Keyboard = Keyboard.Default
                    },
                    new SwitchCell { Text = "Сохранить" }
                },
                new TableSection("Контакты")
                {
                    new EntryCell
                    {
                        Label = "Телефон:",
                        Placeholder = "введите телефон",
                        Keyboard = Keyboard.Telephone
                    },
                    new EntryCell
                    {
                        Label = "Email:",
                        Placeholder = "введите email",
                        Keyboard = Keyboard.Email
                    }
                }
            }
        };
    }
}
```



Чтобы определить содержимое TableView, надо его свойству Root присвоить некоторый объект TableRoot. TableRoot хранит секции таблицы в виде объектов TableSection. Каждая же секция в свою очередь содержит набор отдельных ячеек или элементов Cell

Аналог 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">
  <TableView>
    <TableView.Root>
      <TableRoot>
        <TableSection Title="Персональные данные">
          <EntryCell Label="Логин" Keyboard="Default" Placeholder="Введите логин" />
          <SwitchCell Text="Сохранить" />
        </TableSection>
        <TableSection Title="Контакты">
          <EntryCell Label="Телефон" Keyboard="Telephone" Placeholder="номер телефона" />
          <EntryCell Label="Email" Keyboard="Email" Placeholder="Введите email" />
        </TableSection>
      </TableRoot>
    </TableView.Root>
  </TableView>
</ContentPage>
```

## Типы ячеек

При создании таблицы мы можем использовать разные виды ячеек:

- **EntryCell**: представляет метку с текстовым полем для ввода данных
- **SwitchCell**: представляет метку с переключателем
- **TextCell**: две метки для вывода текста
- **ImageCell**: аналогична TextCell со включением изображения
- **ViewCell**: содержимое и формат отображения данных ячейки определяется разработчиком

Наиболее часто используемыми из них являются SwitchCell и EntryCell.

Каждый тип ячеек имеет свой набор свойств. Например, для ячейки типа SwitchCell можно выделить два следующих свойства:

- **Text**: представляет текст ячейки
- **On**: указывает, находится в отмеченном или нет состоянии

А у EntryCell можно выделить следующие свойства:

- **Keyboard**: тип клавиатуры, которая отображается для ввода текста
- **Label**: текстовая метка, которая отображается слева от поля ввода
- **LabelColor**: цвет текста метки
- **Placeholder**: текст, отображаемый до ввода текста
- **Text**: сам введенный текст
- **HorizontalTextAlignment**: горизонтальное выравнивание текста

Виды таблиц

Свойство **Intent** определяет виды табличных представлений и может принимать следующие значения:

- **Data**: предназначен для простого отображения данных
- **Form**: представляет форму ввода данных, как в примере выше
- **Menu**: используется для вывода меню
- **Settings**: используется для отображения набора настроек

Хотя на первый взгляд разница между этими пунктами не столь очевидна, однако же свойство Intent влияет на то, как TableView будет отображаться на каждой платформе. И хорошей практикой является все же определение свойства Intent и придание ему того значения, которое наиболее оптимально подходит таблицы по ее назначению.

1

**лучший хостинг-провайдер**  
Делаем так, чтобы Ваши веб-проекты работали. Более 80 000 клиентов работает с нами [hoster.by](#)

>

×

ⓘ

2

**Викиум тренажер для мозга**  
[Развивай память, внимание, логику и мышление с помощью игр. Попробуй бесплатно! 6+](#) [wikium.ru](#)

>

×

ⓘ

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



**Гибкий кабель-канал** ▾

[promtechnolog.ru](#)

Яндекс.Директ

**Сильный мороз в Минской области?** ▾

[yandex.by](#)

Яндекс.Директ

10 Комментариев

metanit.com

1

Войти ▾

♥ Рекомендовать

🔗 Поделиться

Лучшее в начале ▾

Присоединиться к обсуждению...

ВОЙТИ С ПОМОЩЬЮ

ИЛИ ЧЕРЕЗ DISQUS ?

Имя

**Maxim Dragaev** • 5 месяцев назад

Добрый день. Код с# работает без ошибок, копирую код XAML выдает ошибку "не удалось привести тип объекта xamarin.forms.xaml.listnode к типу xamarin.forms.xaml.elementnode". В чем может быть проблема?

^ | v • Ответить • Поделиться ›

**Рустам** • 9 месяцев назад

Полностью скопировал код и получил вот такое исключение: "An unhandled exception occurred." Если не создавать TableView, а создать просто кнопку или несколько кнопок, то всё работает. Подскажите в чём дело, пожалуйста.

^ | v • Ответить • Поделиться ›

**Metanit** Модератор → Рустам • 9 месяцев назад

может быть, неправильно скопировали код

^ | v • Ответить • Поделиться ›

**Рустам** → Metanit • 9 месяцев назад

да нет, всё проверено по несколько раз

^ | v • Ответить • Поделиться ›

**Metanit** Модератор → Рустам • 9 месяцев назад

Ни в Xaml, ни в C# точно ошибки нет

^ | v • Ответить • Поделиться ›

**Рустам** → Metanit • 9 месяцев назад

Xaml Я не пробовал.

скопировал только код C#, хотел только его использовать. Разве ещё что-то в XAML нужно дописывать?

^ | v • Ответить • Поделиться ›

**Metanit** Модератор → Рустам • 9 месяцев назад

нет, либо то, либо это, два варианта как альтернатива. Но код на C# точно должен работать.

^ | v • Ответить • Поделиться ›

**Makarkin & Partners / Макаркин** • год назад

что-то клавиатура не выезжает, это у меня что то?

^ | v • Ответить • Поделиться ›

**Metanit** Модератор → Makarkin & Partners / Макаркин • год назад

тестируйте на реальном устройстве, а не на эмуляторе. Хотя в работе xamarin вполне возможны и баги

^ | v • Ответить • Поделиться ›

**Makarkin & Partners / Макаркин** → Metanit • год назад

да действительно. Решилось просто созданием нового эмулятор Android.

^ | v • Ответить • Поделиться ›

ТАКЖЕ НА METANIT.COM

**ASP.NET Core | SignalR Core. Первое приложение**

25 комментариев • 4 месяца назад

**Metanit** — помещаете код из index.html в представление и все

**Vue.js | Валидация props**

1 комментарий • 4 месяца назад

**eugene81** — Подключена минимизированная версия библиотеки - сообщения об ошибках не показываются. При подключении не сжатой - видны

**Руководство по Angular в ASP.NET Core 2.0**

2 комментариев • 2 месяца назад

**Metanit** — да, еще будут статьи

**Angular в ASP.NET Core | CRUD и маршрутизация. Часть 2**

5 комментариев • месяц назад

**Igor Teterkin** — Поправка - это в IE11. В google chrome всё ок



GAMISS



Shop Now >>

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

Контакты для связи: metanit22@mail.ru

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