

Forms and Setting Pages

By: Mosh Hamedani

Switch

For capturing a boolean value.

```
<Switch IsToggled="true" Toggled="..." />
```

Slider and Stepper

Slider: a double between 0 and 1 (by default).

```
<Slider Maximum="255" Minimum="10" Value="50" ValueChanged="..." />
```

Stepper: an integer between 0 and 100 (by default).

```
<Stepper Increment="5" />
```

Entry and Editor

Entry for capturing one line of text, **Editor** for capturing multiple lines.

```
<Entry Keyboard="..." Placeholder="Phone" IsPassword="true"  
    Completed="..." TextChanged="..." />
```

```
<Editor />
```

Forms and Setting Pages

By: Mosh Hamedani

Picker

For allowing the user to select from a short list of items. Prefer to use picker with navigation technique.

In XAML:

```
<Picker>
    <Picker.ItemsSource>
        <x:String>...</x:String>
        <x:String>...</x:String>
    </Picker.ItemsSource>
</Picker>
```

In code-behind:

```
foreach (var str in list)
    picker.Items.Add(str);
```

Date and Time Picker

```
<DatePicker Date="..." MinimumDate="..." MaximumDate="..."
    Format="d MMM yyyy" DateSelected="..." />
```

```
<TimePicker Time="13:00" Format="..." />
```

Forms and Setting Pages

By: Mosh Hamedani

Referencing static members in XAML

```
<ContentPage xmlns:sys="clr-namespace:System;assembly=mscorlib">
    <DatePicker Date="{x:Static sys:DateTime.Today}" />
</ContentPage>
```

TableView

```
<TableView Intent="Form">
    <TableRoot>
        <TableSection Title="Section1">
            ... (one or more cells)
        </TableSection>
    </TableRoot>
</TableView>
```

Cell types:

- TextCell
- ImageCell
- EntryCell
- SwitchCell
- ViewCell (for implementing custom cells)

```
<TextCell Text="..." Detail="..." />
<EntryCell Label="Title" Placeholder="(eg shopping)" />
<SwitchCell Text="Notifications" On="true" OnChanged="..." />
```

Forms and Setting Pages

By: Mosh Hamedani

Bindable Properties

```
public class DateCell : ViewCell
{
    public static readonly BindableProperty LabelProperty =
        BindableProperty.Create(
            "Label",
            typeof(string),
            typeof(DateCell));

    public string Label
    {
        get { return (string) GetValue(LabelProperty); }
        set { SetValue(LabelProperty, value); }
    }
}
```

Picker with Navigation

In XAML:

```
<ViewCell Tapped="Handle_Tapped">
    <StackLayout>
        <Label ... />
        <Label x:Name="contactMethod" ... />
    </StackLayout>
</ViewCell>
```

Forms and Setting Pages

By: Mosh Hamedani

In code-behind:

```
void Handle_Tapped(...)
{
    var page = new ContactMethodsPage();
    page.ContactMethods.ItemSelected += (source, args) =>
    {
        contactMethod.Text = args.SelectedItem.ToString();
        Navigation.PopAsync();
    };

    Navigation.PushAsync(page);
}
```

This requires exposing the ListView in ContactDetails page via a public property.