## Работа с датами. Calendar и DatePicker

Элементы для работы с данными представлены следующими классами: **Calendar** и **DatePicker**.

Calendar представляет собой элемент в виде календаря, тогда как DatePicker - текстовое поле для ввода даты с выпадающим календарем после ввода.

Они имеют некоторые общие свойства:

|  |  |
| --- | --- |
| **BlackoutDates** | Принимает в качестве значения объект **CalendarDateRange**, задающий с помощью свойств **Start** и **End** диапазон дат, которые будут зачеркнуты в календаре. |
| **DisplayDateStart** и **DisplayDateEnd** | Задают соответственно начальную и конечную дату диапазона, который будет отображаться в календаре. |
| **IsTodayHighlighted** | Отмечает, будет ли выделена текущая дата |
| **SelectedDate**(**SelectedDates**) | Задает выделенную дату (диапазон выделенных дат) |
| **FirstDayOfWeek** | Задает первый день недели |

Также Calendar имеет еще два важных свойства: **DisplayMode**(формат отображения дат) и **SelectionMode** (способ выделения).

DisplayMode может принимать одно из следующих значений:

* **Month** (по умолчанию) отображает все дни текущего месяца
* **Decade** отображает все года текущего десятилетия
* **Year** отображает все месяцы текущего года

SelectionMode может принимать одно из следующих значений:

* **SingleDate** (по умолчанию) выделяет только одну дату
* **None** запрещает выделение
* **SingleRange** по нажатию на Ctrl выделяет несколько последовательно идущих дат
* **MultipleRange** по нажатию на Ctrl выделяет несколько не последовательно идущих диапазонов дат

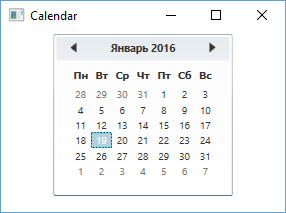
Например:

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | <Calendar x:Name="calendar1" FirstDayOfWeek="Monday"          SelectedDatesChanged="calendar\_SelectedDatesChanged">      <Calendar.BlackoutDates>          <CalendarDateRange Start="10/5/2013" End="10/8/2013"></CalendarDateRange>      </Calendar.BlackoutDates>  </Calendar> |

Обратите внимание, что при задании даты мы сначала пишем месяц, а потом число.

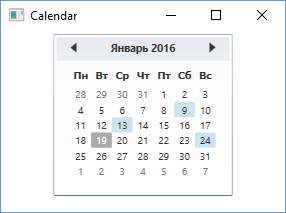
Чтобы использовать в программе выбор даты пользователем, мы можем обработать событие SelectedDatesChanged в коде c#:

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | private void calendar\_SelectedDatesChanged(object sender, SelectionChangedEventArgs e)  {      DateTime? selectedDate = calendar1.SelectedDate;        MessageBox.Show(selectedDate.Value.Date.ToShortDateString());  } |



Для создания набора выделенных дат нам надо подключить в xaml пространство имен System:

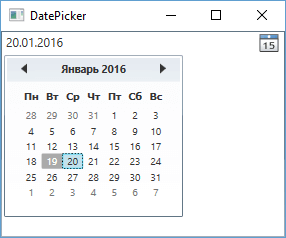
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | <Calendar x:Name="calendar1" FirstDayOfWeek="Monday" SelectionMode="MultipleRange"          xmlns:sys="clr-namespace:System;assembly=mscorlib">      <Calendar.SelectedDates>          <sys:DateTime>01/9/2016</sys:DateTime>          <sys:DateTime>01/13/2016</sys:DateTime>          <sys:DateTime>01/24/2016</sys:DateTime>      </Calendar.SelectedDates>  </Calendar> |



### DatePicker

Кроме выше названных свойств элемент DatePicker определяет еще ряд свойств, среди которых **IsDropDownOpen** (показывает, будет ли связанный с элементом всплывающий календарь оставаться открытым после выбора даты) и **SelectedDateFormat** (формат даты, принимает значение Short - формат в виде дд.мм.гггг или мм.дд.гггг и значение Long - с полными названиями месяцев):

|  |  |
| --- | --- |
| 1 | <DatePicker Name="datePicker1" IsDropDownOpen="True" SelectedDateFormat="Short" /> |



И при нажатии на элемент отобразится тот же календарь для дальнейшего выбора даты.