

Окна сообщений в C# и платформе .NET.

Как правило, для вывода сообщений применяется элемент `MessageBox`. Однако кроме собственно вывода строки сообщения данный элемент может устанавливать ряд настроек, которые определяют его поведение.

Для вывода сообщения в классе `MessageBox` предусмотрен метод **Show**, который имеет различные версии и может принимать ряд параметров. Рассмотрим одну из наиболее используемых версий:

```
DialogResult = MessageBox.Show (
    string text,
    string caption,
    MessageBoxButtons buttons,
    MessageBoxIcon icon,
    MessageBoxDefaultButton defaultButton,
    MessageBoxOptions options );
```

Или заголовок: **DialogResult result = MessageBox.Show(...**

Различия будут в проверке, какая кнопка нажата.

Здесь применяются следующие параметры:

text: текст сообщения

caption: текст заголовка окна сообщения

buttons: кнопки, используемые в окне сообщения. Принимает одно из значений перечисления **MessageBoxButtons**:

- `AbortRetryIgnore`: три кнопки `Abort` (Отмена), `Retry` (Повтор), `Ignore` (Пропустить)
- `OK`: одна кнопка `OK`
- `OKCancel`: две кнопки `OK` и `Cancel` (Отмена)
- `RetryCancel`: две кнопки `Retry` (Повтор) и `Cancel` (Отмена)
- `YesNo`: две кнопки `Yes` и `No`
- `YesNoCancel`: три кнопки `Yes`, `No` и `Cancel` (Отмена)

Таким образом, в зависимости от выбора окно сообщения может иметь от одной до трех кнопок.

icon: значок окна сообщения. Может принимать одно из следующих значений перечисления **MessageBoxIcon**:

- `Asterisk, Information`: значок, состоящий из буквы `i` в нижнем регистре, помещенной в кружок
- `Error, Hand, Stop`: значок, состоящий из белого знака "X" на круге красного цвета.
- `Exclamation, Warning`: значок, состоящий из восклицательного знака в желтом треугольнике
- `Question`: значок, состоящий из вопросительного знака на периметре круга
- `None`: значок у сообщения отсутствует

defaultButton: кнопка, на которую по умолчанию устанавливается фокус. Принимает одно из значений перечисления **MessageBoxDefaultButton**:

- `Button1`: первая кнопка из тех, которые задаются перечислением `MessageBoxButtons`
- `Button2`: вторая кнопка
- `Button3`: третья кнопка

options: параметры окна сообщения. Принимает одно из значений перечисления **MessageBoxOptions**:

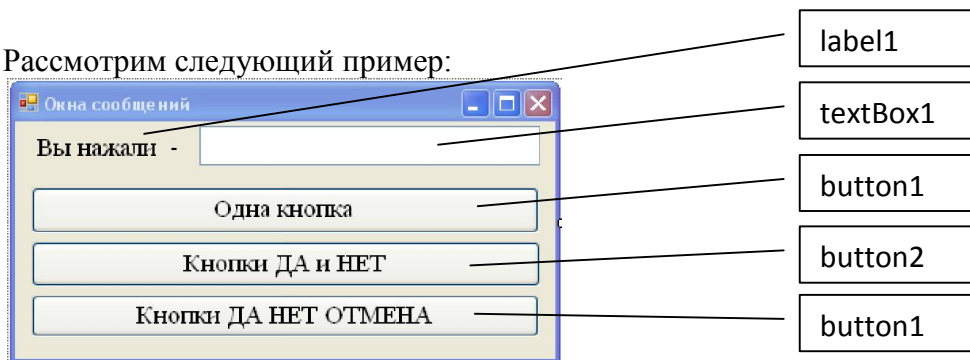
- `DefaultDesktopOnly`: окно сообщения отображается на активном рабочем столе.
- `RightAlign`: текст окна сообщения выравнивается по правому краю
- `RtlReading`: все элементы окна располагаются в обратном порядке справа налево

- **ServiceNotification**: окно сообщения отображается на активном рабочем столе, даже если в системе не зарегистрирован ни один пользователь

Метод **MessageBox.Show** возвращает объект **DialogResult**, с помощью которого мы можем узнать, какую кнопку в окне сообщения нажал пользователь. **DialogResult** представляет перечисление, в котором определены следующие значения:

- **Abort**: нажата кнопка Abort
- **Retry**: нажата кнопка Retry
- **Ignore**: нажата кнопка Ignore
- **OK**: нажата кнопка OK
- **Cancel**: нажата кнопка Cancel
- **None**: отсутствие результата
- **Yes**: нажата кнопка Yes и No
- **No**: нажата кнопка No

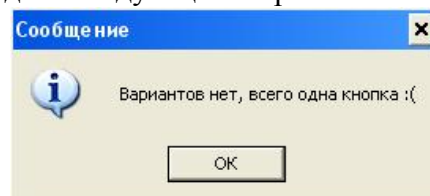
Рассмотрим следующий пример:



Создадим событие для первой кнопки

```
private void button1_Click(object sender, EventArgs e)
{
    MessageBox.Show(
        " Вариантов нет, всего одна кнопка :( ",
        "Сообщение",
        MessageBoxButtons.OK,
        MessageBoxIcon.Information,
        MessageBoxDefaultButton.Button1,
        MessageBoxOptions.DefaultDesktopOnly);
    textBox1.Text="Вы нажали ОК";
    // Ставим нашу форму поверх всех окон
    this.TopMost = true;
}
```

Окно сообщения будет выглядеть следующим образом:



Создадим событие для второй кнопки. Здесь мы создаем переменную **result** типа **DialogResult**.

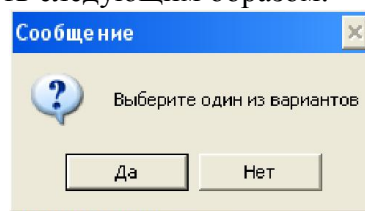
```
private void button2_Click(object sender, EventArgs e)
{
    DialogResult result = MessageBox.Show(
        "Выберите один из вариантов",
```

```

        "Сообщение",
        MessageBoxButtons.YesNo,
        MessageBoxIcon.Question,
        MessageBoxDefaultButton.Button1,
        MessageBoxOptions.DefaultDesktopOnly);
        // Проверяем какая кнопка нажата ...
    if (result == DialogResult.Yes)
        textBox1.Text="Вы нажали ДА";
    else
        textBox1.Text="Вы нажали НЕТ";
    // Ставим нашу форму по верх всех окон
    this.TopMost = true;
}

```

Окно сообщения будет выглядеть следующим образом:



Создадим событие для третьей кнопки. Можно без создания переменной.

```

private void button3_Click(object sender, EventArgs e)
{
    DialogResult = MessageBox.Show(
        "Выберите один из вариантов",
        "Сообщение",
        MessageBoxButtons.YesNoCancel,
        MessageBoxIcon.Error,
        MessageBoxDefaultButton.Button1,
        MessageBoxOptions.DefaultDesktopOnly);
    if (DialogResult == DialogResult.Yes)
        textBox1.Text = "Вы нажали ДА";
    if (DialogResult == DialogResult.No)
        textBox1.Text = "Вы нажали НЕТ";
    if (DialogResult == DialogResult.Cancel)
        textBox1.Text = "Вы нажали ОТМЕНА";
    this.TopMost = true;
}

```

Окно сообщения будет выглядеть следующим образом:

