

Лабораторная работа №16

Разработка событий

1 Цель работы

1.1 Научиться создавать, вызывать и обрабатывать события на C#.

2 Литература

2.1 <https://metanit.com/sharp/tutorial/3.14.php>

3 Задание

Для заданий 1-3 нужно создать отдельные консольные проекты, в которые добавить класс Пользователь с закрытыми полями логин и пароль и открытыми свойствами на чтение и запись для доступа к этим полям.

3.1 Реализация интерфейса INotifyPropertyChanged

3.1.1 Реализовать в классе Пользователь интерфейс INotifyPropertyChanged и создать метод для вызова события.

3.1.2 Реализовать вызов метода из п.3.1.1 при изменении логина и пароля (если исходное значение изменялось).

3.1.3 В консольном приложении создать объект класса пользователя. Подписаться на событие для вывода сообщения о том, какое свойство изменено.

3.2 Получение данных об отправителе события

3.2.1 Добавить в класс Пользователь событие типа EventHandler.

3.2.2 Реализовать вызов события при изменении логина и пароля. В параметрах метода события должен передаваться текущий объект и EventArgs.Empty.

3.2.3 В консольном приложении создать объект класса пользователя. Подписаться на событие для вывода сообщения об изменении свойства класса в следующем формате:

Изменены данные пользователя со следующим логином: <i>логин</i> .

Логин должен браться из параметров объекта-отправителя. Для получения данных о пользователе sender привести к типу Пользователь.

3.3 Создание класса данных события

3.3.1 Создать класс данных события InfoEventArgs со свойствами:

- имяСвойства (строка),
- текстОшибки (строка),
- датаИзменения (дата и время).

3.3.2 Добавить в класс Пользователь событие типа EventHandler<InfoEventArgs>.

3.3.3 Реализовать вызов события при изменении логина и пароля. В параметрах метода события должен передаваться текущий объект и объект типа InfoEventArgs, у которого настроены свойства:

- имяСвойства: «логин» или «пароль» (в зависимости от того, в каком свойстве вызывается событие),
- текстОшибки: для логина ошибка, если логин — пустая строка, для пароля — если строка меньше 6 или больше 20 символов. При корректных данных — пустая строка,
- датаИзменения: текущая дата и время.

3.3.4 В консольном приложении создать объект класса пользователя. Подписаться на событие для вывода сообщения об изменении свойства класса в следующем формате:

<i>дата</i> : у пользователя <i>логин</i> изменено <i>имяСвойства</i> . Ошибка: <i>текстОшибки</i>

Дата, имя свойства и текст ошибки должны браться из параметров объекта класса данных события. Логин должен браться из параметров объекта-отправителя.

3.4 Подписка нескольких объектов на один обработчик события

3.4.1 Создать оконное приложение и разместить на форме:

- 10 кнопок (на каждой — отдельная цифра от 0 до 9),
- поле ввода для отображения нажатых цифр,
- кнопку «Очистить» для очистки поля ввода.

3.4.2 Подписать кнопки с цифрами на один обработчик события `DigitButton_Click`.

3.4.3 В обработчике события получить текст нажатой кнопки и добавить его в конец текста поля ввода.

3.5 Создание делегата для события

3.5.1 Создать оконное приложение из двух форм:

- главная форма отображает текст в многострочном поле ввода (`TextBox` с установленным свойством `Multiline`) и при нажатии на кнопку «Замена» открывает форму замены,
- форма замены используется для указания в одном поле ввода текста для поиска, во втором поле ввода — текста для замены.

3.5.2 Реализовать замену текста в поле ввода главной формы при нажатии на кнопку «Заменить» на форме замены.

Для реализации:

- создать для события делегат типа `void` со строковым параметром,
- в форме замены:
 - создать открытое событие на основе созданного делегата,
 - создать обработчик события нажатия на кнопку «Заменить» (при нажатии вызывать созданное событие),
- в главной форме:
 - создать подписку на событие, созданное в форме замены,
 - в обработчике события на главной форме выполнить замену текста.

4 Порядок выполнения работы

4.1 Выполнить все задания из п.3 в одном решении `LabWork16`, каждое — в отдельном проекте `TaskN`. Возможные ошибки требуется обрабатывать. Выполнить форматирование и рефакторинг кода.

4.2 Ответить на контрольные вопросы.

5 Содержание отчета

5.1 Титульный лист

5.2 Цель работы

5.3 Ответы на контрольные вопросы

5.4 Вывод

6 Контрольные вопросы

6.1 Что такое «событие» в С#?

6.2 Как объявить событие на С#?

6.3 Как создать обработчик события?

6.4 Какой класс является родительским для всех классов данных события?

6.5 Какие классы делегатов являются стандартными для создания событий в С#?