

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

3.1 Реализация интерфейса INotifyPropertyChanged

3.1.1 Создать класс Пользователь с закрытыми полями логин и пароль и открытыми свойствами для доступа к ним.

3.1.2 В классе Пользователь:

- реализовать интерфейс INotifyPropertyChanged,
- создать метод для вызова события PropertyChanged,
- реализовать вызов метода при изменении логина и пароля (если исходное значение изменилось).

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

3.2 Создание класса данных события

3.2.1 В приложение из п.3.1 добавить класс данных события InfoEventArgs со свойствами на чтение и инициализацию:

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

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

3.2.3 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),

- текстОшибка: для логина ошибка, если логин — пустая строка, для пароля — если строка меньше 8 символов. При корректных данных — пустая строка,

- датаИзменения: текущая дата и время.

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

Без ошибки:

дата: у пользователя логин изменено имяСвойства.

С ошибкой:

дата: у пользователя логин не изменено имяСвойства. Ошибка: текстОшибка

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

3.3 Подписка нескольких объектов на один обработчик события

3.3.1 Создать оконное приложение и разместить на форме:

- FlowLayoutPanel,

- многострочное поле ввода (RichTextBox или TextBox с установленным свойством Multiline) для отображения нажатых букв.

3.3.2 Программно добавить 33 кнопки (на каждой – отдельный символ русского алфавита от а до я) и подписать их на один обработчик события LetterButton_Click.

3.3.3 В обработчике события получать текст нажатой кнопки и добавлять его в конец текста поля ввода.

3.4 Создание подписки и отписки на событие в оконном приложении

3.4.1 Добавить в приложение из п.3.3 кнопку «Замена». При нажатии кнопки открывать форму замены, на которой разместить:

- поле ввода текста для поиска,
- поле ввода текста для замены,
- кнопку «Заменить».

3.4.2 Реализовать замену текста в поле ввода главной формы при нажатии на кнопку «Заменить» на форме замены, используя механизм событий и стандартные события:

- событие EventHandler<ReplaceEventArgs> ReplaceRequested,
- класс ReplaceEventArgs со свойствами SearchText и ReplaceText.

3.4.3 Реализовать отписку от события формы замены на главной форме при событии закрытия формы замены (FormClosed).

4 Порядок выполнения работы

4.1 Выполнить все задания из п.3 в одном решении LabWork16. 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 C#?

6.2 Как объявить событие на C#?

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 C#?