

# Лабораторная работа 4.

## Делегаты. События

### Информация для всех вариантов

В лабораторной работе требуется определить класс, содержащий типизированную коллекцию, который с помощью событий извещает об изменениях в коллекции и в данных ее элементов.

Коллекция состоит из объектов ссылочных типов. Набор ссылок, образующих коллекцию, изменяется при удалении/добавлении элементов или при изменении одной из входящих в коллекцию ссылок, например, когда одной из ссылок присваивается новое значение.

Класс, содержащий коллекцию, сообщает об этих изменениях в коллекции с помощью событий, для которых определяется пользовательский делегат.

Данные элементов, входящих в коллекцию, изменяются при вызове метода `get` в свойствах, определенных в элементах коллекции. При этом значение самой ссылки, которая входит в коллекцию, не изменяется.

Элементы сообщают об изменениях в своих данных с помощью события **Propertychange** интерфейса **System.ComponentModel.INotifyPropertyChanged**. Делегат, отвечающий этому событию,

```
delegate void PropertyChangedEventHandler  
(Object sender, PropertyChangedEventArgs e);
```

определен в базовой библиотеке. В классе **PropertyChangedEventArgs** есть конструктор с единственным параметром типа `string`, через который обработчикам события обычно передается имя свойства, которое является источником изменений данных объекта.

Класс, содержащий коллекцию, должен подписаться на событие **PropertyChanged** для каждого элемента коллекции. Подписку на событие **PropertyChanged** надо выполнить во всех методах, которые добавляют в коллекцию новые элементы. При удалении элемента из коллекции от подписки на его событие **PropertyChanged** необходимо отказаться.

При подписке на событие **PropertyChanged** как обработчик события надо использовать метод класса с коллекцией. В этом методе события **PropertyChanged**, которые происходят в элементах коллекции, преобразуются в события класса, содержащего коллекцию. Для этого из информации, которую получает обработчик события **PropertyChanged**, формируются данные для нового события, которое бросается из класса-коллекции.

События регистрируются в специальных классах-слушателях. Класс-слушатель должен отслеживать изменения только в тех объектах, которые

ВХОДЯТ В КОЛЛЕКЦИЮ.

## Вариант 1. Требования к программе

Определить новые версии классов **Student** и **StudentCollection<TKey>** из лабораторной работы 3.

Новая версия класса **Student** реализует интерфейс **System.ComponentModel.INotifyPropertyChanged**. Событие **PropertyChanged** из интерфейса **System.ComponentModel.INotifyPropertyChanged** происходит при изменении значений свойств класса **Student**, связанных с номером группы и формой обучения. Название свойства, значение которого изменилось, событие **PropertyChanged** передает своим обработчикам через свойство **PropertyName** класса **PropertyChangedEventArgs**.

Для информации о типе изменений, которые произошли в коллекциях, определить перечисление (enum) **Action** со значениями **Add**, **Remove** и **Property**.

Для события, которое бросают методы класса **StudentCollection<TKey>**, определить делегат **StudentsChangedHandler<TKey>** с сигнатурой:

```
void StudentsChangedHandler<TKey>
(object source, StudentsChangedEventArgs<TKey> args);
```

Класс **StudentsChangedEventArgs<TKey>**, производный от класса **System.EventArgs**, содержит

- открытое автореализуемое свойство типа **string** с названием коллекции;
- открытое автореализуемое свойство типа **Action** с информацией о том, чем вызвано событие, - удалением элемента, добавлением элемента или изменением данных элемента;
- открытое автореализуемое свойство типа **string** с названием свойства класса **Student**, которое является источником изменения данных элемента; для событий, брошенных при удалении или добавлении элемента, значение свойства - пустая строка;
- открытое автореализуемое свойство типа **TKey** с ключом добавленного, удаленного или измененного элемента;
- конструктор с параметрами типа **string**, **Action**, **string** и **TKey** для инициализации значений всех свойств класса;
- перегруженную версию метода **string ToString()**.

В новую версию класса **StudentCollection<TKey>** добавить

- открытое автореализуемое свойство типа **string** с названием

коллекции;

- метод `bool Remove(Student st)` для удаления элемента со значением `st` из словаря `Dictionary<TKey, Student>`; если в словаре нет элемента `st`, метод возвращает значение `false`;
- событие **StudentsChanged** типа **StudentsChangedHandler<TKey>**, которое происходит, когда в коллекцию добавляются элементы, из нее удаляется элемент или изменяются данные одного из ее элементов.

Определить класс **Journal**, который можно использовать для накопления информации об изменениях в коллекциях типа `StudentCollection<TKey>`. Класс **Journal** хранит информацию об изменениях в коллекциях в списке объектов типа **JournalEntry**. Класс **JournalEntry** содержит информацию об отдельном изменении, которое произошло в коллекциях.

Класс **JournalEntry** содержит автореализуемые свойства

- типа `string` с названием коллекции;
- типа `Action` с информацией о типе события;
- типа `string` с названием свойства класса `Student`, которое явилось причиной изменения данных элемента;
- типа `string` с текстовым представлением ключа добавленного, удаленного или измененного элемента;
- конструктор для инициализации всех свойств класса;
- перегруженную версию метода `string ToString()`.

Класс **Journal** содержит

- закрытое поле типа `System.Collections.Generic.List<JournalEntry>`;
- обработчик события `StudentsChanged`, который на основе информации из объекта `StudentsChangedEventArgs`, создает элемент `JournalEntry` и добавляет его в список `List<JournalEntry>`;
- перегруженную версию метода `string ToString()` для формирования строки с информацией обо всех элементах списка `List<JournalEntry>`.

В методе **Main()**

1. Создать две коллекции `StudentCollection<string>` с разными названиями.
2. Создать объект `Journal` и подписать его на события `StudentsChanged` из обеих коллекций `StudentCollection<string>`.
3. Внести изменения в коллекции `StudentCollection<string>`:
  - добавить элементы `Student` в коллекции;
  - изменить значения разных свойств элементов, входящих в коллекцию;
  - удалить элемент из коллекции;
  - изменить данные в удаленном элементе.

4. Вывести данные объекта Journal.

## Вариант 2. Требования к программе

Определить новые версии классов **Edition**, **Magazine** и **MagazineCollection<TKey>** из лабораторной работы 3.

Новая версия класса **Edition** реализует интерфейс **System.ComponentModel.INotifyPropertyChanged**. Событие **PropertyChanged** из интерфейса **System.ComponentModel.INotifyPropertyChanged** происходит при изменении значений свойств класса **Edition**, связанных с тиражом и датой выхода издания. Название свойства, значение которого изменилось, событие **PropertyChanged** передает своим обработчикам через свойство **PropertyName** класса **PropertyChangedEventArgs**.

Для информации о типе изменений, которые произошли в коллекциях, определить перечисление (enum) **Update** со значениями **Add**, **Replace** и **Property**.

Для события, которое бросают методы новой версии класса **MagazineCollection<TKey>**, определить универсальный делегат **MagazinesChangedHandler<TKey>** с сигнатурой:

```
void MagazinesChangedHandler<TKey>
(object source, MagazinesChangedEventArgs<TKey> args);
```

Класс **MagazinesChangedEventArgs<TKey>**, производный от класса **System.EventArgs**, содержит

- открытое автореализуемое свойство типа **string** с названием коллекции;
- открытое автореализуемое свойство типа **Update** с информацией о том, чем вызвано событие, - добавлением нового элемента в коллекцию, заменой элемента в коллекции или изменением данных элемента;
- открытое автореализуемое свойство типа **string** с названием свойства класса **Magazine**, которое является источником изменения данных элемента; для событий, порожденных добавлением или заменой элемента, значение свойства - пустая строка;
- открытое автореализуемое свойство типа **TKey** с ключом элемента, который был добавлен в коллекцию, заменил один из элементов коллекции или элемента, у которого были изменены данные;
- конструктор с параметрами типа **string**, **Update**, **string** и **TKey** для инициализации значений всех свойств класса;
- перегруженную версию метода **string ToString()**.

В новую версию класса **MagazineCollection<TKey>** добавить

- открытое автореализуемое свойство типа `string` с названием коллекции;
- метод `bool Replace(Magazine mold, Magazine mnew)` для замены в словаре `Dictionary<TKey, Magazine>` элемента со значением `mold` на элемент со значением `mnew`; если в словаре нет элемента со значением `mold`, метод возвращает значение `false`;
- событие **MagazinesChanged** типа `MagazinesChangedHandler<TKey>`, которое происходит при добавлении нового элемента в коллекцию, замене элемента в коллекции или при изменении данных одного из ее элементов.

Определить класс **Listener**, собирающий информацию об изменениях в классе `MagazineCollection<TKey>`. Класс `Listener` содержит список из элементов типа **ListEntry**. Каждый элемент `ListEntry` содержит информацию об отдельном изменении объекта `MagazineCollection<TKey>`, в результате которого произошло событие `MagazinesChanged`.

Класс **ListEntry** содержит автореализуемые свойства

- типа `string` с названием коллекции;
- типа `Update` с информацией о типе события;
- типа `string` с названием свойства класса `Magazine`, которое явилось причиной изменения данных элемента;
- типа `string` с текстовым представлением ключа добавленного, удаленного или измененного элемента;
- конструктор для инициализации всех свойств класса;
- перегруженную версию метода `string ToString()`.

Класс **Listener** содержит

- закрытое поле типа `System.Collections.Generic.List<ListEntry>`;
- обработчик события `MagazinesChanged`, который на основе информации из объекта `MagazinesChangedEventArgs`, создает элемент `ListEntry` и добавляет его к списку;
- перегруженную версию метода `string ToString()` для формирования строки с информацией обо всех элементах списка `List<ListEntry>`.

В методе **Main()**

1. Создать два объекта `MagazineCollection<string>` с разными названиями.
2. Создать объект типа `Listener` и подписать его на события `MagazinesChanged` из обоих объектов `MagazineCollection<string>`.
3. Внести изменения в `MagazineCollection<string>`:
  - добавить элементы в коллекции;
  - изменить значения разных свойств элементов, входящих в

- коллекцию;
- заменить один из элементов коллекции;
- изменить данные в элементе, который был удален из коллекции при замене элемента.

4. Вывести данные объекта Listener.

## Вариант 3. Требования к программе

Определить новые версии классов **ResearchTeam** и **ResearchTeamCollection<TKey>** из лабораторной работы 3.

Новая версия класса **ResearchTeam** реализует интерфейс **System.ComponentModel.INotifyPropertyChanged**. Событие **PropertyChanged** из интерфейса **System.ComponentModel.INotifyPropertyChanged** происходит при изменении значений свойств класса **ResearchTeam**, связанных с названием темы и продолжительностью исследований. Название свойства, значение которого изменилось, событие **PropertyChanged** передает своим обработчикам через свойство **PropertyName** класса **PropertyChangedEventArgs**. Для информации о типе изменений, которые произошли в коллекциях, определить перечисление (enum) **Revision** со значениями **Remove**, **Replace** и **Property**.

Для события **ResearchTeamsChanged**, которое бросают методы новой версии класса **ResearchTeamCollection<TKey>**, определить универсальный делегат **ResearchTeamsChangedHandler<TKey>** с сигнатурой:

```
void ResearchTeamsChangedHandler<TKey>
    (object source, ResearchTeamsChangedEventArgs<TKey>
    args) ;
```

Класс **ResearchTeamsChangedEventArgs<TKey>**, производный от класса **System.EventArgs**, содержит

- открытое автореализуемое свойство типа **string** с названием коллекции;
- открытое автореализуемое свойство типа **Revision** с информацией о том, чем вызвано событие, - удалением, заменой элемента или изменением данных элемента;
- открытое автореализуемое свойство типа **string** с названием свойства класса **ResearchTeam**, которое является источником изменения данных элемента; для событий, брошенных при удалении или замене элемента, значение свойства - пустая строка;
- открытое автореализуемое свойство типа **int** с номером регистрации объекта **ResearchTeam** для элемента, который был удален, в том числе и при замене элемента, или элемента, данные которого были

- изменены;
- конструктор с параметрами типа string, Revision, string и int для инициализации значений всех свойств класса;
- перегруженную версию метода string ToString().

В новую версию класса **ResearchTeamCollection<TKey>** добавить

- открытое автореализуемое свойство типа string с названием коллекции;
- метод bool Remove(ResearchTeam rt) для удаления элемента со значением rt из словаря Dictionary<TKey, ResearchTeam>; если в словаре нет элемента rt, метод возвращает значение false;
- метод bool Replace(ResearchTeam rtold, ResearchTeam rtnew) для замены в словаре Dictionary<TKey, ResearchTeam > элемента со значением rtold на элемент со значением rtnew; если в словаре нет элемента со значением rtold, метод возвращает значение false;
- событие **ResearchTeamsChanged** типа ResearchTeamsChangedHandler<TKey>, которое происходит, когда изменяется набор элементов в коллекции-словаре Dictionary<TKey,ResearchTeam> или изменяются данные одного из ее элементов.

Определить класс **TeamsJournal**, который можно использовать для накопления информации об изменениях в коллекциях типа ResearchTeamCollection<TKey>. Класс TeamsJournal содержит список из элементов типа **TeamsJournalEntry**. Каждый элемент TeamsJournalEntry содержит информацию об отдельном изменении объекта ResearchTeamCollection<TKey>, в результате которого произошло событие ResearchTeamsChanged.

Класс **TeamsJournalEntry** содержит автореализуемые свойства

- типа string с названием коллекции;
- типа Revision с информацией о типе события;
- типа string с названием свойства класса ResearchTeam, которое явилось причиной изменения данных элемента;
- типа int с номером регистрации объекта ResearchTeam для удаленного элемента или элемента, данные которого были изменены;
- конструктор для инициализации всех свойств класса;
- перегруженную версию метода string ToString().

Класс **TeamsJournal** содержит

- закрытое поле List<TeamsJournalEntry> для списка изменений;
- обработчик события ResearchTeamsChanged; обработчик использует информацию, которая передается ему через объект ResearchTeamsChangedEventArgs, создает элемент TeamJournalEntry и

- добавляет его к списку `List<TeamsJournalEntry>`;
- перегруженную версию метода `string ToString()` для формирования строки с информацией обо всех элементах списка `List<TeamsJournalEntry>`.

В методе **Main()**

1. Создать две коллекции `ResearchTeamCollection<string>`.
2. Создать объект `TeamsJournal`, подписать его на события `ResearchTeamsChanged` из обоих объектов `ResearchTeamCollection<string>`.
3. Внести изменения в коллекции `ResearchTeamCollection<string>`
  - добавить элементы в коллекции;
  - изменить значения разных свойств элементов, входящих в коллекцию;
  - удалить элемент из коллекции;
  - изменить данные в удаленном элементе;
  - заменить один из элементов коллекции;
  - изменить данные в элементе, который был удален из коллекции при замене элемента.
4. Вывести данные объекта `TeamsJournal`.