Лабораторная работа 3

Универсальные типы. Классы-коллекции. Методы расширения класса System.Linq.Enumerable

Требования к программе, общие для всех вариантов

Во всех вариантах лабораторной работы 3 требуется определить класс TestCollections, который содержит поля следующих типов

• System.Collections.Generic.List ;

• System.Collections.Generic.List ;

•System.Collections.Generic.Dictionary ;

• System.Collections.Generic.Dictionary .

Конкретные значения типовых параметров TKey и TValue зависят от варианта. Во всех вариантах тип ключа TKey и тип значения TValue связаны отношением базовый-производный. Во всех вариантах в классе TValue определено свойство, которое возвращает ссылку на объект типа TKey с данными, совпадающими с данными подобъекта базового класса (это свойство должно возвращать ссылку на объект типа TKey, а не ссылку на вызывающий объект TValue).

В конструкторе класса TestCollections создаются коллекции с заданным числом элементов. Надо сравнить время поиска элемента в коллекцияхсписках List и время поиска элемента по ключу и элемента по значению в коллекциях-словарях Dictionary.

Для автоматической генерации элементов коллекций в классе TestCollections надо определить статический метод, который принимает один целочисленный параметр типа int и возвращает ссылку на объект типа TValue.

Каждый объект TValue содержит подобъект базового класса TKey. Соответствие между значениями целочисленного параметра метода и подобъектами TKey класса TValue должно быть взаимно-однозначным – равным значениям параметра должны отвечать равные объекты TKey и наоборот. Равенство объектов типа TKey трактуется так же, как это было сделано в лабораторной работе 2 при определении операций равенства объектов.

Все четыре коллекции содержат одинаковое число элементов. Каждому элементу из коллекции List должен отвечать элемент в коллекции Dictionary с равным значением ключа. Список List состоит из строк, которые получены в результате вызова метода ToString() для объектов TKey из списка List. Каждому элементу списка List отвечает элемент в коллекции-словаре Dictionary с равным значением ключа типа string.

Число элементов в коллекциях вводится пользователем в процессе работы приложения. Если при вводе была допущена ошибка, приложение должно обработать исключение, сообщить об ошибке ввода и повторить прием ввода до тех пор, пока не будет правильно введено целочисленное значение.

Для четырех разных элементов – первого, центрального, последнего и элемента, не входящего в коллекцию – надо измерить время поиска

• элемента в коллекциях List и List с помощью метода Contains;

• элемента по ключу в коллекциях Dictionary< TKey, TValue> и Dictionary с помощью метода ContainsKey;

• значения элемента в коллекции Dictionary< TKey, TValue > с помощью метода ContainsValue.

Так как статический метод для автоматической генерации элементов должен обеспечивать взаимно-однозначное соответствие между значением целочисленного параметра метода и объектами TKey, этот метод можно использовать как при создании коллекций с большим числом элементов, так и для генерации элемента для поиска.

Вариант 1.

Требования к программе

Определить новые версии классов Person и Student из лабораторной работы 2.

В класс Person добавить реализацию интерфейсов

• System.IComparable для сравнения объектов типа Person по полю с фамилией;

• System.Collections.Generic.IComparer для сравнения объектов типа Person по дате рождения.

В новой версии класса Student для списков зачетов и экзаменов использовать типы

• System.Collections.Generic.List для списка зачетов;

• System.Collections.Generic.List для списка экзаменов.

В новой версии класса Student сохранить все остальные поля, свойства и методы из предыдущей версии класса, внести необходимые исправления в код свойств и методов из-за изменения типов полей для списков зачетов и экзаменов.

Определить вспомогательный класс, реализующий интерфейс System.Collections.Generic.IComparer, который можно использовать для сравнения объектов типа Student по среднему баллу.

Определить класс StudentCollection, который содержит

• закрытое поле типа System.Collections.Generic.List;

• метод void AddDefaults(), c помощью которого можно добавить некоторое число элементов типа Student для инициализации коллекции по умолчанию;

• метод void AddStudents (params Student[] ) для добавления элементов в список List;

• перегруженную версию виртуального метода string ToString() для формирования строки c информацией обо всех элементах списка List, включающую значения всех полей, список зачетов и экзаменов для каждого элемента Student;

• метод string ToShortString(), который формирует строку c информацией обо всех элементах списка List, содержащую значения всех полей, средний балл, число зачетов и число экзаменов для каждого элемента Student, но без списков зачетов и экзаменов.

В классе StudentCollection определить методы, выполняющие сортировку списка List

• по фамилии студента с использованием интерфейса IComparable, реализованного в классе Person;

• по дате рождения студента с использованием интерфейса IComparer, реализованного в классе Person;

• по среднему баллу с использованием интерфейса IComparer, реализованного во вспомогательном классе.

В классе StudentCollection определить свойства и методы, выполняющие операции со списком List с использованием методов расширения класса System.Linq.Enumerable, и статические методы-селекторы, которые необходимы для выполнения соответствующих операций со списком:

• свойство типа double (только с методом get), возвращающее максимальное значение среднего балла для элементов списка List; если в коллекции нет элементов, свойство возвращает некоторое значение по умолчанию; для поиска максимального значения среднего балла использовать метод Max класса System.Linq.Enumerable;

• свойство типа IEnumerable (только с методом get), возвращающее подмножество элементов списка List с формой обучения Education.Specialist; для формирования подмножества использовать метод Where класса System.Linq.Enumerable;

• метод List AverageMarkGroup(double value), который возвращает список, в который входят элементы Student из списка List с заданным значением среднего балла; для формирования списка использовать методы Group и ToList класса System.Linq.Enumerable.

Определить класс TestCollections, в котором в качестве типа TKey используется класс Person, а в качестве типа TValue - класс Student. Класс содержит закрытые поля с коллекциями типов

• System.Collections.Generic.List<Person>;

• System.Collections.Generic.List<string>;

• System.Collections.Generic.Dictionary <Person, Student>;

• System.Collections.Generic.Dictionary <string, Student>.

В классе TestCollections определить

• статический метод с одним целочисленным параметром типа int, который возвращает ссылку на объект типа Student и используется для автоматической генерации элементов коллекций;

• конструктор c параметром типа int (число элементов в коллекциях) для автоматического создания коллекций с заданным числом элементов;

• метод, который вычисляет время поиска элемента в списках List<Person> и List<string>, время поиска элемента по ключу и время поиска элемента по значению в коллекциях-словарях Dictionary<Person, Student> и Dictionary<string, Student>.

В методе Main()

1. Создать объект типа StudentCollection. Добавить в коллекцию несколько различных элементов типа Student и вывести объект StudentCollection.

2. Для созданного объекта StudentCollection вызвать методы, выполняющие сортировку списка List по разным критериям, и после каждой сортировки вывести данные объекта. Выполнить сортировку

• по фамилии студента;

• по дате рождения;

• по среднему баллу.

3. Вызвать методы класса StudentCollection, выполняющие операции со списком List, и после каждой операции вывести результат операции. Выполнить

• вычисление максимального значения среднего балла для элементов списка;

• фильтрацию списка для отбора студентов с формой обучения Education.Specialist;

• группировку элементов списка по значению среднего балла; вывести все группы элементов.

4. Создать объект типа TestCollections. Вызвать метод для поиска в коллекциях первого, центрального, последнего и элемента, не 30 входящего в коллекции. Вывести значения времени поиска для всех четырех случаев. Вывод должен содержать информацию о том, к какой коллекции и к какому элементу относится данное значение.