

Инструменты и средства программирования

Лабораторная работа №7

Многопоточное программирование

Цель работы: знакомство с пространством имен `system.Threading.Tasks`

Задача работы: научиться запускать методы в отдельных потоках, использовать механизмы синхронизации для управления работой потоков.

Время выполнения работы: 4 часа

Результат выполнения работы: программа, обеспечивающая функционал согласно заданию.

Классы создавать в отдельном проекте – библиотека классов .NET. ||

Задание

Описать **один** класс предметной области согласно варианту. Класс должен содержать свойства:

- `Id` (уникальный номер),
- название объекта,
- войство, необходимое для подсчета статистической информации

Создать класс **`StreamService<T>`**, в котором описать три метода:

1. `public async Task WriteToStreamAsync(Stream stream, IEnumerable<T> data, IProgress<string> progress)` – записывает коллекцию *data* в поток *stream*;
2. метод `public async Task CopyFromStreamAsync(Stream stream, string filename, IProgress<string> progress)` - копирует информацию из потока *stream* в файл с именем *fileName*;
3. метод `public async Task<int> GetStatisticsAsync(string fileName, Func<T, bool> filter)` считывает объекты типа *T* из файла с именем *filename* и возвращает количество объектов, удовлетворяющих условию *filter*.

Использовать **асинхронные** методы чтения/записи в поток/файл

Предусмотреть **синхронизацию** методов 1 и 2.

В методах 1 и 2 предусмотреть оповещение о начале и конце записи/чтения в/из потока *stream* с указанием номера потока выполнения (`Thread.CurrentThread.ManagedThreadId`). Соответствующее сообщение должно быть выведено в консоль в классе Program (использовать `IProgress`)

В классе Program:

1. Сделать метод Main асинхронным (`static async Task Main(string[] args)`)
2. Создать коллекцию из 1000 объектов согласно индивидуальному заданию.
3. Вывести в консоль номер потока выполнения и сообщение о начале работы;
4. Синхронно (без использования *await*) запустить методы 1 и 2 класса StreamService (для гарантирования последовательности запуска потоков установите задержку 100-200 мс между запусками методов). В качестве данных передавайте коллекцию из п.2. В качестве параметра stream использовать MemoryStream. Методы 1 и 2 должны использовать **один и тот же** экземпляр *stream*.
5. Вывести в консоль номер потока выполнения и сообщение о том, что потоки 1 и 2 запущены;
6. **Ожидать** завершения выполнения методов 1 и 2;
7. **Асинхронно** получить статистические данные (метод `GetStatisticsAsync`);
8. Полученные статистические данные вывести в консоль
9. Модифицировать методы 1 и 2 для генерирования события о процессе выполнения задания (использовать `IProgress`). В событии передавать Id потока и выполняемое действие. В классе Program подписаться на событие и вывести в консоль информацию в виде:

Поток xxxxxxxxxxxx: уууу

Где: xxx – id потока, ууу – процент выполнения задачи

Рекомендации к заданию 2:

1000 объектов можно создать в цикле, присваивая формальные значения свойств класса в виде `$«[Имя свойства] {i}»`. Можно также использовать генератор случайных чисел.

Для генерирования случайных строковых данных можно подгрузить Nuget пакет `Lorem.Universal.NET`.

Примечание:

Асинхронные вызовы в методе `Main` класса `Program` доступны в C# версии 7.1 и выше проектах `.Net Framework` или `.Net Core 2.x` и выше.

Варианты заданий

1. Предметная область – недвижимость. Статистическая информация – количество домов, в которых количество жильцов больше 100;
2. Предметная область – компьютеры. Статистическая информация – количество компьютеров определенной марки
3. Предметная область – пассажиры. Статистическая информация – количество пассажиров, у которых есть багаж
4. Предметная область – сотрудники предприятия. Статистическая информация – количество сотрудников старше 35 лет
5. Предметная область – автопарк. Статистическая информация – количество автомобилей, у которых техосмотр должен проводиться в текущем году
6. Предметная область – агентство по трудоустройству. Статистическая информация – количество кандидатов на работу определенного профиля
7. Предметная область – автосалон. Статистическая информация – количество автомобилей с объемом двигателя более 2 литров
8. Предметная область – художественная галерея. Статистическая информация – количество работ определенного мастера
9. Предметная область – пассажиры. Статистическая информация – количество пассажиров, у которых есть багаж
10. Предметная область – продукты питания. Статистическая информация – количество продуктов, у которых закончился срок годности
11. Предметная область – музыка. Статистическая информация – количество песен определенного исполнителя

- 12.Предметная область – студенты. Статистическая информация – количество студентов, у которых средний балл больше 9
- 13.Предметная область – биология. Статистическая информация – количество существ, которые умеют летать
- 14.Предметная область – обучающие курсы. Статистическая информация – количество курсов, на которых количество слушателей больше 10
- 15.Предметная область – игрушки. Статистическая информация – количество игрушек для детей от 6 лет
- 16.Предметная область – клиенты банка. Статистическая информация – количество клиентов, открывших счет в текущем году
- 17.Предметная область – багаж. Статистическая информация – количество багажа, у которого вес превышает 20 кг.
- 18.Предметная область – пациенты больницы. Статистическая информация – количество пациентов с определенным диагнозом
- 19.Предметная область –соревнования по стрельбе. Статистическая информация – количество спортсменов, у которых сумма баллов больше 80
- 20.Предметная область – супергерои. Статистическая информация – количество героев, которые владеют определенным оружием.