



Программа курса «Платформа Microsoft .NET и язык программирования C#»

Платформа Microsoft .NET та мова програмування C#
Microsoft .NET Framework and C# Programming Language

Для групп полустационара. Версия 3.1.2

Объем курса: 13 уроков

Цель курса

Обучить слушателя основам разработки приложений с использованием платформы Microsoft .NET и языка программирования C#. Научить выбирать правильные механизмы и конструкции для решения той или иной задачи.

По окончании курса слушатель будет:

- понимать причины возникновения платформы Microsoft .NET;
- оперировать базовыми терминами платформы Microsoft .NET: CLR, CLS, CTS, BCL;
- уметь использовать рефлекторы и дотфускаторы;
- разбираться в тонкостях реализации ООП в C#;
- уметь создавать классы пользовательских исключений;
- уметь создавать пользовательские делегаты и события;
- взаимодействовать со сборщиком мусора;
- взаимодействовать с унаследованным кодом;
- уметь использовать механизмы сериализации;
- применять инструменты логирования.

По окончании данного курса студент сдает практическое задание и теоретический экзамен по материалам курса. Для допуска к экзамену должны быть сданы все домашние и практические задания.

Практическое задание должно охватывать максимум материала из различных разделов курса.

Тематический план

- Урок 1.** Введение в платформу Microsoft .NET. Типы данных. Операторы
- Урок 2.** Массивы и строки. Перечисления
- Урок 3.** Структуры. Классы. Свойства
- Урок 4.** Наследование и полиморфизм
- Урок 5.** Перегрузка операторов
- Урок 6.** Интерфейсы
- Урок 7.** Обработка исключений. Сборщик мусора
- Урок 8.** Generics, использование generic-коллекций
- Урок 9.** Делегаты, события. LINQ
- Урок 10.** Взаимодействие с файловой системой. Сериализация
- Урок 11.** Работа с XML-файлами. Логирование
- Урок 12.** Использование унаследованного программного кода
- Урок 13.** Экзамен

Урок 1

Введение в платформу Microsoft .NET. Типы данных. Операторы

1. Введение в платформу Microsoft .NET.
 - История и этапы развития технологий программирования.
 - Причины возникновения платформы Microsoft .NET.
 - Сравнительный анализ преимуществ и недостатков платформы Microsoft .NET.
2. Базовые понятия платформы Microsoft .NET.
 - Архитектура платформы Microsoft .NET.
 - Общеязыковая среда исполнения CLR (common language runtime).
 - Стандартная система типов CTS (common type system).
 - Стандартная языковая спецификация CLS (common language specification).
 - Библиотека классов FCL (BCL).
 - Языки платформы Microsoft .NET.
 - Язык CIL (Common Intermediate Language).
 - Схема компиляции и исполнения приложения платформы Microsoft .NET.
 - Понятия метаданных, манифеста, сборки.
3. .NET Core, .NET Standard.
 - Что такое .NET Core?
 - Цели и задачи .NET Core.
 - Что такое .NET Standard?
 - Цели и задачи .NET Standard.
 - Сравнение .NET Core, .NET Standard, .NET Framework.
4. Введение в язык программирования C#.
 - Плюсы и минусы языка программирования C#.
 - Простейшая программа на языке программирования C#.
5. Рефлекторы и дотфускаторы.
 - Что такое рефлектор?
 - Необходимость использования рефлектора.
 - Обзор существующих рефлекторов.
 - Что такое дотфускатор?
 - Необходимость использования дотфускаторов.
 - Обзор существующих дотфускаторов.

6. Типы данных.
 - Целочисленные типы данных.
 - Типы данных для чисел с плавающей точкой.
 - Символьный тип данных.
 - Логический тип данных.
7. Nullable типы.
 - Что такое nullable тип.
 - Цели и задачи nullable типов.
 - Операции доступные для nullable типов.
 - Примеры использования.
8. Литералы.
9. Переменные.
 - Понятие переменной.
 - Правила именования переменных.
 - Область видимости переменных.
10. Ввод, вывод в консольном приложении.
11. Структурные и ссылочные типы.
12. Преобразование типов.
 - Явное преобразование.
 - Неявное преобразование.
13. Операторы.
 - Арифметические операторы.
 - Операторы отношений.
 - Логические операторы.
 - Битовые операторы.
 - Оператор присваивания.
 - Приоритет операторов.
14. Условия.
 - Условный оператор if.
 - Условный оператор if else.
 - Условный оператор switch.
 - Тернарный оператор ?..
15. Циклы.
 - Цикл for.
 - Цикл while.
 - Цикл do while.
 - Цикл foreach.

- Инструкция break.
- Инструкция continue.
- Инструкция goto.

Урок 2

Массивы и строки. Перечисления

1. Массивы.
 - Одномерные массивы.
 - Многомерные массивы.
 - Зубчатые массивы.
 - Использование цикла foreach.
2. Строки.
 - Создание строки.
 - Операции со строками.
 - Особенности использования строк.
3. Использование аргументов командной строки.
4. Перечисления (enum).
 - Понятие перечисления.
 - Синтаксис объявления перечисления.
 - Необходимость и особенности применения перечисления.
 - Установка базового типа перечисления.
 - Использование методов для перечислений.

Урок 3

Структуры. Классы. Свойства

1. Структуры.
 - Понятие структуры.
 - Синтаксис объявления структуры.
 - Необходимость и особенности применения структур.
 - Конструктор без параметров и структуры
2. Синтаксис объявления класса.
3. Модификаторы доступа языка программирования C#.
4. Поля класса.
5. Конструкторы.
 - Понятие конструктора.

- Параметризованный конструктор.
 - Перегруженные конструкторы.
 - Статический конструктор.
6. Ключевое слово `this`.
7. Методы класса.
- Передача параметров.
 - Ключевое слово `return`.
 - Перегрузка методов.
 - Краткий синтаксис однострочных методов.
8. Использование `ref` и `out` параметров.
- Использование модификатора `ref`.
 - Использование модификатора `out`.
 - Кортежи.
9. Создание методов с переменным количеством аргументов.
10. Частичные типы (`partial types`).
11. Свойства.
- Что такое свойства?
 - Синтаксис объявления свойств.
 - Краткий синтаксис однострочных свойств.
 - Примеры использования свойств.
 - Автоматические свойства (`auto-property`):
 - что такое автоматические свойства?
 - примеры использования автоматических свойств;
 - инициализация автоматических свойств.
 - `Null-conditional` оператор.
12. Пространство имен.
- Что такое пространство имен?
 - Цели и задачи пространства имен.
 - Ключевое слово `using`.
 - Объявление пространства имен.
 - Вложенные пространства имен.
 - Разбиение пространства имен на части.
 - Пространство имен по умолчанию.
 - Директива псевдонима `using`.
 - Использование `using` для подключения статических членов.

Урок 4

Наследование и полиморфизм

1. Наследование в C#.
 - Анализ механизма наследования в C#.
 - Спецификаторы доступа при наследовании.
 - Особенности использования конструкторов при наследовании.
 - Ключевое слово base.
 - Соккрытие имен при наследовании.
2. Использование ключевого слова sealed.
3. Использование ссылок на базовый класс.
4. Полиморфизм в C#. Виртуальные методы
 - Что такое виртуальный метод?
 - Переопределение виртуальных методов.
 - Необходимость использования виртуальных методов.
5. Абстрактный класс.
6. Анализ базового класса Object.

Урок 5

Перегрузка операторов

1. Введение в перегрузку операторов
2. Перегрузка унарных операторов.
3. Перегрузка бинарных операторов.
4. Перегрузка операторов отношений.
5. Перегрузка операторов true и false.
6. Перегрузка логических операторов.
7. Перегрузка операторов преобразования.
8. Индексаторы.
 - Понятие индексатора.
 - Создание одномерных индексаторов.
 - Создание многомерных индексаторов.
 - Перегрузка индексаторов.

Урок 6

Интерфейсы

1. Понятие интерфейса.
2. Синтаксис объявления интерфейсов.
3. Примеры создания интерфейсов.
4. Интерфейсные ссылки.
5. Интерфейсные свойства и индексы.
6. Наследование интерфейсов.
7. Проблемы сокрытия имен при наследовании интерфейсов.
8. Анализ стандартных интерфейсов.

Урок 7

Обработка исключений. Сборщик мусора

1. Иерархия исключений.
 - Базовый класс System.Exception.
 - Анализ иерархии стандартных исключений.
 - Наследование и исключения.
 - Наследование от стандартных классов исключений.
2. Основы обработки исключений.
 - Ключевое слово try.
 - Ключевое слово catch.
 - Ключевое слово finally.
 - Ключевое слово throw.
3. Тонкости обработки исключений.
 - Перехват всех исключений.
 - Вложенные блоки try.
 - Повторное генерирование исключений.
4. Применение конструкций checked и unchecked.
5. Фильтры исключений.
6. Использование nameof.
7. Жизненный цикл объектов.
8. Понятие сборщика мусора.
9. Понятие поколений при сборке мусора.
10. Класс System.GC.

11. Финализатор и метод Finalize.
12. Метод Dispose и интерфейс IDisposable.
13. Использование using при работе с классами, реализующими интерфейс IDisposable.

Урок 8

Generics, использование generic-коллекций

1. Generics.
 - Что такое generics?
 - Необходимость использования generics. Упаковка, распаковка (boxing, unboxing).
 - Сравнительный анализ generic-коллекций и необобщенных коллекций.
 - Создание generic-классов.
 - Вложенные типы внутри generic-класса.
 - Использование ограничений.
 - Создание generic-методов.
 - Generic-интерфейсы.
2. Итераторы.
 - Что такое итератор?
 - Синтаксис и примеры использования итераторов.
3. Коллекции.
 - Понятие коллекции.
 - Generic-коллекции.
 - Классы коллекций List<T>, Dictionary<TKey,TValue>, Stack<T>, Queue<T>, SortedList<TKey,TValue> и другие.
 - Интерфейсы коллекций IList<T>, IEnumerator<T>, IEnumerable<T>, ICollection<T>, IDictionary<TKey,TValue>, IComparer<T>, ...
 - Примеры использования классов generic-коллекций.

Урок 9

Делегаты, события. LINQ

1. Делегаты.
 - Понятие делегата.
 - Базовые классы для делегатов:
 - System.Delegate;
 - System.MulticastDelegate.

- Синтаксис объявления делегата.
 - Цели и задачи делегатов.
 - Вызов нескольких методов через делегат (multicasting).
 - Создание generic делегатов.
2. События.
 - Понятие события.
 - Синтаксис объявления события.
 - Необходимость и особенности применения событий.
 - Применение события для многоадресного делегата.
 - Использование событийных средств доступа.
 3. Анонимные методы.
 4. Лямбда выражения.
 - Реализация тела метода в виде выражения.
 5. Extension методы.
 6. LINQ to Object.
 - Роль LINQ.
 - Исследование операций запросов LINQ.
 - Возврат результата запроса LINQ. Анонимные типы.
 - Применение запросов LINQ к объектам коллекций.

Урок 10

Взаимодействие с файловой системой. Сериализация

1. Модель потоков в C#. Пространство System.IO.
2. Класс Stream.
3. Анализ байтовых классов потоков.
4. Анализ символьных классов потоков.
5. Анализ двоичных классов потоков.
6. Использование класса FileStream для файловых операций.
7. Использование класса StreamWriter для файловых операций.
8. Использование класса StreamReader для файловых операций.
9. Использование класса BinaryWriter для файловых операций.
10. Использование класса BinaryReader для файловых операций.
11. Использование классов Directory, DirectoryInfo, File и FileInfo для файловых операций.
12. Регулярные выражения.

13. Понятие атрибутов.
14. Что такое сериализация?
15. Отношения между объектами.
16. Графы отношений объектов.
17. Атрибуты для сериализации [Serializable] и [NonSerialized].
18. Форматы сериализации.
 - Пространство System.Runtime.Serialization.Formatters.
 - Двоичное форматирование. Класс BinaryFormatter.
 - Soap форматирование. Класс SoapFormatter.
 - Сериализация Xml. Класс XmlSerializer.
 - Примеры использования сериализации.
 - Создание пользовательского формата сериализации. Интерфейс ISerializable.

Урок 11

Работа с XML-файлами. Логирование

1. Парсеры XML.
 - Что такое парсер?
 - Цели и задачи парсера.
 - DOM- и SAX-парсеры.
2. Примеры создания XML-документов.
3. XML-документация.
 - Что такое xml-документация кода?
 - Зачем использовать xml-документацию.
 - Примеры использования.
4. Что такое логирование?
5. Цели и задачи логирования.
6. Ситуации, требующие логирования.
7. Инструменты логирования:
 - log4net;
 - Serilog;
 - NLog.
8. Практические примеры использования.

Урок 12

Использование унаследованного программного кода

1. Почему необходимо использовать унаследованный программный код.
2. Пространство System.Runtime.InteropServices.
3. Взаимодействие с модулями Dll.
4. Класс DllImportAttribute.
 - Поле ExactSpelling.
 - Поле EntryPoint.
 - Поле CharSet.
 - Поле CallingConvention.
 - Поле SetLastError.
5. Примеры использования.
6. Создание DLL на языке C#.
7. Global Assembly Cache.
 - Что такое Global Assembly Cache?
 - Цели и задачи GAC.
 - Строгие имена и версии DLL.
 - Установка сборок в GAC.
8. Класс Assembly.
 - Загрузка сборок.
9. Механизмы безопасности в .Net Framework (Code Access Security).

Урок 13

Экзамен