

Содержание

Введение	14
Соглашения, принятые в этой книге	15
От издательства	15
Глава 1. На старт, внимание, марш!	17
Загрузка и установка инструментальных средств	17
Загрузка Visual C# Express	18
Выбор типа приложения	19
Создание проектов и решений	20
Создание приложения Windows	20
Просмотр исходного кода	21
Переименование решения	22
Сохранение решения	22
Запуск приложения Windows	23
Создание приложения Windows, говорящего "привет"	24
Добавление комментариев в приложение	27
Навигация по пользовательским элементам управления решения	28
Создание консольного приложения	30
Добавление проекта консольного приложения в решение	31
Создание консольного приложения, говорящего "привет"	31
Установка стартового проекта	31
Запуск консольного проекта	32
Создание библиотеки классов	32
Добавление проекта библиотеки классов к решению	32
Перенос функциональных возможностей	32
Определение ссылок	33
Вызов функций библиотеки классов	34
Использование переменных и констант	35
Понятие о том, как работает .NET Framework	37
Что следует запомнить	39
Самостоятельные упражнения	40
Глава 2. Типы и значения чисел .NET	41
Организация разработки	41
Организация калькулятора	42
Сосредоточимся на калькуляторе	43
Реализация библиотеки классов	46
Создание метода Add ()	48
Создание проверочного кода для метода Add ()	50
Понятие проблем с числовыми значениями и типами	55
Понятие типов и значений чисел	57
Понятие значений и ссылок	57
Понятие числовых типов CLR	59
Завершение калькулятора	62
Что следует запомнить	63
Самостоятельные упражнения	63
Глава 3. Манипулирование строками	65
Организация приложения перевода	65

Построение приложения переводчика	66
Создание класса Translator	66
Перевод слова Hello	67
Создание проверочного приложения	68
Ответ на вопрос об ответственности	69
Исследуем тип string	69
Решение проблемы отступа	74
Строка в кавычках	78
Соотнесение символов	79
Работа с языками и национальными форматами	80
Установка языка и национального формата в Windows	81
Анализ и обработка чисел	81
Работа с национальным форматом	84
Что следует запомнить	86
Самостоятельные упражнения	86
Глава 4. Структуры данных, решения и циклы	89
Понятие алгоритма поиска в глубину	89
Реализация пользовательских типов	92
Объявление структур и классов	92
Ограничения типа значения	93
Организация алгоритма поиска	98
Написание кода поиска в глубину	100
Определение и реализация структуры данных	100
Определение проверки алгоритма	109
Реализация алгоритма поиска в глубину	113
Запуск алгоритма поиска в глубину	120
Что следует запомнить	121
Самостоятельные упражнения	122
Глава 5. Обработка исключений C#	123
Понятие ошибки, исключения и обработки исключения	123
Запуск отладчика	125
Обработка исключений	125
Перехват исключения	126
Реализация обработчиков исключений	128
Защита от прокрутки стека	131
Фильтрация исключений	132
Создание кода, защищенного от исключений	135
Запись защитного кода	135
Использование стандартного состояния	138
Обработка ошибок и предупреждений	139
Что следует запомнить	140
Самостоятельные упражнения	140
Глава 6. Основы объектно-ориентированного программирования	141
Понятие разных валют	142
Организация приложения по обмену валют	142
Написание проверок для приложения обмена валют	143
Приступая к работе со структурным кодом	143
Понятие базовых классов	143
Понятие наследования	144

8 Содержание

Использование свойств C#	146
Понятие наследования и модификаторов области видимости	149
Осуществление верификации	153
Завершение базового класса	155
Код конвертации валют биржевого брокера и гостиничного обменного пункта	156
Реализация класса ActiveCurrencyTrader	156
Реализация класса HotelCurrencyTrader	159
Подробнее о директивах препроцессора, свойствах и абстрактных методах	161
Подробнее о директивах препроцессора	161
Подробнее об областях видимости свойств	163
Подробнее о ключевом слове abstract	163
Что следует запомнить	165
Самостоятельные упражнения	166
Глава 7. Компоненты и иерархия объектов	167
Основные концепции налогообложения	167
Организация налогового приложения	168
Программирование с использованием идей	169
Представление идей с помощью интерфейсов C#	170
Как работают компоненты и наследование	172
Реализация процессора расчета налогов	178
Определение интерфейсов	178
Реализация базового класса процессора расчета налогов	180
Использование стандартных реализаций	184
Реализация класса BaseTaxAccount	186
Использование базовых функций процессора расчета налогов	187
Реализация процессора расчета налогов	188
Использование процессора расчета налогов	192
Подробнее о наследовании и приведении типов	193
Подробнее о наследовании	193
Подробнее о преобразовании типов	199
Что следует запомнить	200
Самостоятельные упражнения	200
Глава 8. Архитектура на базе компонентов	201
Понятие ядра	201
Организация приложения освещения	202
Построение ядра	203
Определение интерфейсов	204
Реализация ядра	207
Определение ядра как интерфейса, а не класса	223
Создание полного приложения	224
Определение некоторых комнат	225
Создайте экземпляры PublicRoom и PrivateRoom	226
Подробнее о закрытых классах и инициализации объекта	227
Закрытые классы	227
Инициализация объекта с вложенными типами данных	228
Что следует запомнить	229
Самостоятельные упражнения	229
Глава 9. Списки, делегаты и лямбда-выражения	231
Управление коллекцией	231

Управление коллекциями до C# 2.0	231
Управление коллекцией после появления C# 2.0	236
Случай с кодом, который выглядит неправильно	237
Использование делегатов	240
Анонимные методы	245
Многоадресная пересылка и делегаты	245
Лямбда-выражения	246
Понятие лямбда-выражений	248
Создание алгоритма	249
Реализация алгоритма, использующего лямбда-выражения	249
Подробнее о типах коллекций	251
Традиционный список	251
Использование списка пар ключ–значение	252
Использование стека	252
Использование очереди	253
Что следует запомнить	253
Самостоятельные упражнения	254
Глава 10. Все о персистентности	255
Организация системы предсказания лотереи	255
Пересылка данных с помощью консоли	256
Чтение данных с консоли	256
Построение оболочки	258
Реализация приложения TextProcessor	267
Конвейерная пересылка бинарных данных	275
Определение интерфейсов и реализация оболочки	276
Определение типа	279
Преобразование текстового потока в бинарный	280
Преобразование бинарного потока в текстовый	281
Настройка сериализации	282
Специальная сериализация	283
Объявление переменной-члена как не подлежащей сериализации	284
Отделение объектов данных от объектов действия	284
Завершение специальных типов	284
Реализация метода GetHashCode ()	285
Реализация метода Equals ()	288
Что следует запомнить	289
Самостоятельные упражнения	289
Глава 11. Исследование общих типов .NET	291
Зачем использовать общие типы .NET?	291
Теория серверной электронной таблицы	294
Архитектура серверной электронной таблицы	296
Проект архитектуры	297
Определение интерфейсов серверных электронных таблиц	298
Реализация сервера электронной таблицы	307
Использование в электронной таблице лямбда-выражений	312
Назначение состояния без знания типа	315
Переопределение функции Tostring ()	317
Перебор данных с использованием перечислителей	318
Использование электронной таблицы	319

10 Содержание

Вычисление среднего значения	319
Почему вычисления срабатывают	321
Что следует запомнить	322
Самостоятельные упражнения	323
Глава 12. Конфигурация приложения и динамическая загрузка	325
Соглашение по конфигурации	325
Разъединение с помощью архитектуры конфигурации	327
Разъединение с помощью архитектуры соглашений	327
Установка проектов динамической загрузки	328
Подпись сборки	329
Установка пути вывода	330
Определение и обработка файла конфигурации	331
Создание файла конфигурации на базе XML	332
Добавление элементов конфигурации динамической загрузки	333
Чтение файла конфигурации	334
Динамическая загрузка сборки	335
Динамическое создание экземпляра типа	335
Расширение файла конфигурации	339
Загрузка строго именованной сборки	343
Настройка строго именованной сборки для GAC	344
Использование номеров версий	346
Реализация архитектуры на базе соглашений	348
Динамическая загрузка базового класса или типа интерфейса	350
Что следует запомнить	351
Самостоятельные упражнения	352
Глава 13. Все о многопоточности	353
Понятие многозадачности	353
Приоритетная многозадачность	354
Понятие квантования времени	354
Использование потоков	356
Создание нового потока	357
Ожидание завершения потока	358
Создание потока с состоянием	359
Синхронизация потоков	360
Предотвращение взаимоблокирования кода	365
Реализация потоковой архитектуры чтения и записи	369
Реализация архитектуры производителя и потребителя	372
Скрытая реализация производителя и потребителя	373
Реализация общей архитектуры производителя и потребителя	374
Асинхронный подход	376
Что следует запомнить	378
Самостоятельные упражнения	379
Глава 14. Использование реляционных баз данных	381
Понятие реляционных баз данных	381
Таблицы реляционных баз данных	382
Отношения базы данных	383
Доступ к реляционным базам данных	386
Разработка базы данных с помощью Visual C# Express	387
Настройка источника данных	387
Добавление таблиц	390

Доступ к базе данных с помощью ADO.NET	394
Подключение к базе данных	395
Добавление данных в таблицы	395
Выборка данных из таблицы	397
Удаление данных из базы данных	397
Закрытие подключения к базе данных	398
Резюме по использованию ADO.NET	398
Использование дизайнера набора данных	398
Создание отношений между таблицами	398
Использование созданного кода	403
Что следует запомнить	404
Самостоятельные упражнения	404
Глава 15. Изучение языка LINQ	405
Поиск частоты выигрышных чисел	405
Расширение системы предсказания лотереи	406
Реализация решения поиска частоты	409
Узнаем больше о приемах LINQ	415
Выбор и изменение данных	418
Выбор при анонимных типах	419
Обработка нескольких потоков	420
Сортировка результатов	421
Разбиение результатов на группы	421
Выполнение набора операций	423
Использование LINQ в других контекстах	425
Что следует запомнить	426
Самостоятельные упражнения	427
Глава 16. Написание функционального кода на C#	429
Что такое функциональное программирование	429
Сущность функционального программирования	431
Функции высшего порядка	432
Чистые функции	435
Вычисление функций	440
Рекурсия	442
Что следует запомнить	443
Самостоятельные упражнения	444
Глава 17. Остальные вопросы C#	445
Операторы	445
Использование арифметических операторов	445
Перегрузка операторов	449
Оператор <code>goto</code>	452
Понятие ограничений общих типов .NET	452
Использование ограничения <code>type</code>	453
Использование ограничения <code>new</code>	454
Использование ограничения <code>class</code>	455
Нулевые типы	455
Частичные классы и методы	457
Что следует запомнить	459
Самостоятельные упражнения	459
Предметный указатель	461