

ОГЛАВЛЕНИЕ

Благодарности	14
Об авторе	15
Авторские права на иллюстрации	15
От издательства	15
Введение	16
Что нового в пятом издании	18
Целевая аудитория	19
Подход, примененный в этой книге	20
Глава 1. Введение в объектно-ориентированные концепции	21
Фундаментальные концепции	22
Объекты и унаследованные системы	23
Процедурное программирование в сравнении с объектно-ориентированным	24
Переход с процедурной разработки на объектно-ориентированную	29
Процедурное программирование	29
Объектно-ориентированное программирование	29
Что такое объект?	30
Данные объектов	30
Поведение объектов	31
Что такое класс?	35
Создание объектов	35

Атрибуты	37
Методы	37
Сообщения	38
Использование диаграмм классов в качестве визуального средства	38
Инкапсуляция и сокрытие данных	39
Интерфейсы	39
Реализации	40
Реальный пример парадигмы «интерфейс/реализация»	41
Модель парадигмы «интерфейс/реализация»	41
Наследование	43
Суперклассы и подклассы	44
Абстрагирование	45
Отношения «является экземпляром»	46
Полиморфизм	47
Композиция	51
Абстрагирование	51
Отношения «содержит как часть»	52
Резюме	52
Глава 2. Как мыслить объектно	53
Разница между интерфейсом и реализацией	54
Интерфейс	56
Реализация	57
Пример интерфейса/реализации	57
Использование абстрактного мышления при проектировании классов	63
Обеспечение минимального интерфейса пользователя	65
Определение пользователей	66
Поведения объектов	67
Ограничения, налагаемые средой	67
Определение открытых интерфейсов	68
Определение реализации	69
Резюме	70
Ссылки	70

Глава 3. Прочие объектно-ориентированные концепции	71
Конструкторы	71
Когда осуществляется вызов конструктора?	72
Что находится внутри конструктора?	72
Конструктор по умолчанию	73
Использование множественных конструкторов	74
Перегрузка методов	75
Использование UML для моделирования классов	76
Как сконструирован суперкласс?	78
Проектирование конструкторов	79
Обработка ошибок	79
Игнорирование проблем	80
Проверка на предмет проблем и прерывание выполнения приложения	80
Проверка на предмет проблем и попытка устранить неполадки	80
Выбрасывание исключений	81
Важность области видимости	84
Локальные атрибуты	84
Атрибуты объектов	86
Атрибуты классов	88
Перегрузка операторов	89
Множественное наследование	90
Операции с объектами	91
Резюме	93
Ссылки	93
Глава 4. Анатомия класса	94
Имя класса	94
Комментарии	95
Атрибуты	97
Конструкторы	98
Методы доступа	101
Методы открытых интерфейсов	103
Методы закрытых реализаций	104
Резюме	105
Ссылки	105

Глава 5. Руководство по проектированию классов.....	106
Моделирование реальных систем	106
Определение открытых интерфейсов	108
Минимальный открытый интерфейс	108
Скрытие реализации	109
Проектирование надежных конструкторов (и, возможно, деструкторов)	110
Внедрение обработки ошибок в класс.....	111
Документирование класса и использование комментариев	111
Создание объектов с прицелом на взаимодействие.....	112
Проектирование с учетом повторного использования	113
Проектирование с учетом расширяемости	113
Делаем имена описательными	114
Абстрагирование непереносимого кода.....	115
Обеспечение возможности осуществлять копирование и сравнение	116
Сведение области видимости к минимуму	116
Проектирование с учетом сопровождаемости	117
Использование итерации в процессе разработки	118
Тестирование интерфейса	118
Использование постоянства объектов.....	120
Сериализация и маршалинг объектов	121
Резюме	122
Ссылки	122
Глава 6. Проектирование с использованием объектов	123
Руководство по проектированию.....	123
Проведение соответствующего анализа	128
Составление технического задания.....	128
Сбор требований	129
Разработка прототипа интерфейса пользователя	129
Определение классов	129
Определение ответственности каждого класса	130
Определение взаимодействия классов друг с другом	130
Создание модели классов для описания системы	130
Прототипирование интерфейса пользователя	130

Объектные обертки	131
Структурированный код	132
Обертывание структурированного кода	133
Обертывание непереносимого кода.....	135
Обертывание существующих классов	136
Резюме	137
Ссылки	138
Глава 7. Наследование и композиция	139
Повторное использование объектов	139
Наследование	141
Обобщение и конкретизация.....	145
Проектные решения	146
Композиция	148
Почему инкапсуляция является фундаментальной объектно-ориентированной концепцией	151
Как наследование ослабляет инкапсуляцию.....	151
Подробный пример полиморфизма	154
Ответственность объектов	154
Абстрактные классы, виртуальные методы и протоколы.....	158
Резюме	160
Ссылки	160
Глава 8. Фреймворки и повторное использование: проектирование с применением интерфейсов и абстрактных классов	162
Код: использовать повторно или нет?.....	162
Что такое фреймворк?.....	163
Что такое контракт?.....	166
Абстрактные классы.....	166
Интерфейсы.....	170
Связываем все воедино	172
Код, выдерживающий проверку компилятором	175
Заключение контракта	176
Системные «точки расширения».....	179

Пример из сферы электронного бизнеса	179
Проблема, касающаяся электронного бизнеса	179
Подход без повторного использования кода	180
Решение для электронного бизнеса	183
Объектная модель UML	183
Резюме	188
Ссылки	188
Глава 9. Создание объектов и объектно-ориентированное проектирование ...	189
Отношения композиции	190
Поэтапное создание	191
Типы композиции	194
Агрегации	194
Ассоциации	195
Использование ассоциаций в сочетании с агрегациями	196
Избегание зависимостей	197
Кардинальность	198
Ассоциации, включающие множественные объекты	200
Необязательные ассоциации	202
Связываем все воедино: пример	203
Резюме	204
Ссылки	204
Глава 10. Паттерны проектирования	205
Чем хороши паттерны проектирования?	206
Схема «Модель — Представление — Контроллер» в языке Smalltalk	207
Типы паттернов проектирования	209
Порождающие паттерны	210
Структурные паттерны	215
Паттерны поведения	218
Антипаттерны	219
Заключение	221
Ссылки	221

Глава 11. Избегание зависимостей и тесно связанных классов.....	222
Композиция против наследования и внедрения зависимостей	225
1. Наследование	225
2. Композиция.....	227
Внедрение зависимостей	230
Внедрение с помощью конструктора.....	232
Заключение.....	233
Ссылки	233
Глава 12. Принципы объектно-ориентированного проектирования SOLID.....	234
Принципы объектно-ориентированной разработки SOLID.....	236
1. SRP: принцип единственной ответственности	236
2. OCP: принцип открытости/закрытости	239
3. LSP: принцип подстановки Лисков.....	242
4. ISP: принцип разделения интерфейса	245
5. DIP: принцип инверсии зависимостей	246
Заключение.....	253
Ссылки	253
Об обложке	254