
Содержание

Предисловие	15
Глава 1. Введение в язык Java	21
1.1. Программная платформа Java	21
1.2. Характерные особенности Java	22
1.2.1. Простота	23
1.2.2. Объектно-ориентированный характер	23
1.2.3. Поддержка распределенных вычислений в сети	24
1.2.4. Надежность	24
1.2.5. Безопасность	24
1.2.6. Независимость от архитектуры компьютера	25
1.2.7. Переносимость	25
1.2.8. Интерпретируемость	26
1.2.9. Производительность	26
1.2.10. Многопоточность	27
1.2.11. Динамичность	27
1.3. Апплеты и Интернет	28
1.4. Краткая история развития Java	29
1.5. Распространенные заблуждения относительно Java	32
Глава 2. Среда программирования на Java	35
2.1. Установка комплекта Java Development Kit	35
2.1.1. Загрузка комплекта JDK	36
2.1.2. Установка комплекта JDK	37
2.1.3. Установка библиотек и документации	39
2.2. Применение инструментов командной строки	40
2.3. Применение IDE	45
2.4. Утилита JShell	48
Глава 3. Основные языковые конструкции Java	51
3.1. Простая программа на Java	51
3.2. Комментарии	55
3.3. Типы данных	55
3.3.1. Целочисленные типы данных	56
3.3.2. Числовые типы данных с плавающей точкой	57
3.3.3. Тип данных char	58
3.3.4. Юникод и тип char	59
3.3.5. Тип данных boolean	60
3.4. Переменные и константы	61
3.4.1. Объявление переменных	61
3.4.2. Инициализация переменных	62
3.4.3. Константы	63
3.4.4. Перечислимые типы	64
3.5. Операции	64
3.5.1. Арифметические операции	64
3.5.2. Математические функции и константы	65

3.5.3. Преобразование числовых типов	67
3.5.4. Приведение типов	68
3.5.5. Сочетание арифметических операций с присваиванием	69
3.5.6. Операции инкремента и декремента	69
3.5.7. Операции отношения и логические операции	70
3.5.8. Поразрядные логические операции	70
3.5.9. Круглые скобки и иерархия операций	71
3.6. Символьные строки	72
3.6.1. Подстроки	72
3.6.2. Сцепление строк	73
3.6.3. Принцип постоянства символьных строк	73
3.6.4. Проверка символьных строк на равенство	75
3.6.5. Пустые и нулевые строки	76
3.6.6. Кодовые точки и единицы	76
3.6.7. Прикладной программный интерфейс API класса String	77
3.6.8. Оперативно доступная документация на API	80
3.6.9. Построение символьных строк	82
3.7. Ввод и вывод	84
3.7.1. Чтение вводимых данных	84
3.7.2. Форматирование выводимых данных	86
3.7.3. Файловый ввод и вывод	91
3.8. Управляющая логика	93
3.8.1. Область видимости блоков	93
3.8.2. Условные операторы	93
3.8.3. Неопределенные циклы	97
3.8.4. Определенные циклы	100
3.8.5. Оператор switch для многовариантного выбора	104
3.8.6. Операторы прерывания логики управления программой	106
3.9. Большие числа	108
3.10. Массивы	111
3.10.1. Объявление массивов	112
3.10.2. Доступ к элементам массива	113
3.10.3. Цикл в стиле for each	114
3.10.4. Копирование массивов	115
3.10.5. Параметры командной строки	116
3.10.6. Сортировка массивов	116
3.10.7. Многомерные массивы	119
3.10.8. Неровные массивы	122
Глава 4. Объекты и классы	125
4.1. Введение в ООП	126
4.1.1. Классы	126
4.1.2. Объекты	128
4.1.3. Идентификация классов	128
4.1.4. Отношения между классами	129
4.2. Применение предопределенных классов	130
4.2.1. Объекты и объектные переменные	131
4.2.2. Класс LocalDate из библиотеки Java	134
4.2.3. Модифицирующие методы и методы доступа	135
4.3. Определение собственных классов	139
4.3.1. Класс Employee	139
4.3.2. Использование нескольких исходных файлов	142
4.3.3. Анализ класса Employee	142

4.3.4. Первые действия с конструкторами	143
4.3.5. Объявление локальных переменных с помощью ключевого слова var	144
4.3.6. Обработка пустых ссылок на объекты	145
4.3.7. Явные и неявные параметры	146
4.3.8. Преимущества инкапсуляции	147
4.3.9. Привилегии доступа к данным в классе	149
4.3.10. Закрытые методы	150
4.3.11. Конечные поля экземпляра	150
4.4. Статические поля и методы	151
4.4.1. Статические поля	151
4.4.2. Статические константы	152
4.4.3. Статические методы	153
4.4.4. Фабричные методы	154
4.4.5. Метод main ()	154
4.5. Параметры методов	157
4.6. Конструирование объектов	163
4.6.1. Перегрузка	163
4.6.2. Инициализация полей по умолчанию	164
4.6.3. Конструктор без аргументов	164
4.6.4. Явная инициализация полей	165
4.6.5. Имена параметров	166
4.6.6. Вызов одного конструктора из другого	167
4.6.7. Блоки инициализации	167
4.6.8. Уничтожение объектов и метод finalize ()	171
4.7. Пакеты	172
4.7.1. Именованые пакеты	172
4.7.2. Импорт классов	172
4.7.3. Статический импорт	174
4.7.4. Ввод классов в пакеты	174
4.7.5. Область видимости пакетов	177
4.7.6. Путь к классам	179
4.7.7. Указание пути к классам	181
4.8. Архивные JAR-файлы	182
4.8.1. Создание JAR-файлов	182
4.8.2. Файл манифеста	183
4.8.3. Исполняемые JAR-файлы	184
4.8.4. Многоверсионные архивные JAR-файлы	184
4.8.5. Примечание к параметрам командной строки	186
4.9. Документирующие комментарии	187
4.9.1. Вставка комментариев	188
4.9.2. Комментарии к классам	188
4.9.3. Комментарии к методам	189
4.9.4. Комментарии к полям	189
4.9.5. Комментарии общего характера	190
4.9.6. Комментарии к пакетам	191
4.9.7. Извлечение комментариев	191
4.10. Рекомендации по разработке классов	192
Глава 5. Наследование	195
5.1. Классы, суперклассы и подклассы	196
5.1.1. Определение подклассов	196
5.1.2. Переопределение методов	197
5.1.3. Конструкторы подклассов	199

5.1.4. Иерархии наследования	203
5.1.5. Полиморфизм	203
5.1.6. Представление о вызовах методов	205
5.1.7. Предотвращение наследования: конечные классы и методы	207
5.1.8. Приведение типов	209
5.1.9. Абстрактные классы	211
5.1.10. Защищенный доступ	216
5.2. Глобальный суперкласс Object	217
5.2.1. Переменные типа Object	218
5.2.2. Метод equals ()	218
5.2.3. Проверка объектов на равенство и наследование	219
5.2.4. Метод hashCode ()	223
5.2.5. Метод toString ()	225
5.3. Обобщенные списочные массивы	231
5.3.1. Объявление списочных массивов	232
5.3.2. Доступ к элементам списочных массивов	234
5.3.3. Совместимость типизированных и базовых списочных массивов	237
5.4. Объектные оболочки и автоупаковка	238
5.5. Методы с переменным числом параметров	242
5.6. Классы перечислений	243
5.7. Рефлексия	245
5.7.1. Класс Class	246
5.7.2. Основы обработки исключений	248
5.7.3. Ресурсы	249
5.7.4. Анализ функциональных возможностей классов с помощью рефлексии	251
5.7.5. Анализ объектов во время выполнения с помощью рефлексии	257
5.7.6. Написание кода универсального массива с помощью рефлексии	263
5.7.7. Вызов произвольных методов и конструкторов	266
5.8. Рекомендации по применению наследования	270
Глава 6. Интерфейсы, лямбда-выражения и внутренние классы	273
6.1. Интерфейсы	274
6.1.1. Понятие интерфейса	274
6.1.2. Свойства интерфейсов	280
6.1.3. Интерфейсы и абстрактные классы	281
6.1.4. Статические и закрытые методы	282
6.1.5. Методы с реализацией по умолчанию	283
6.1.6. Разрешение конфликтов с методами по умолчанию	284
6.1.7. Интерфейсы и обратные вызовы	286
6.1.8. Интерфейс Comparator	289
6.1.9. Клонирование объектов	290
6.2. Лямбда-выражения	296
6.2.1. Причины для употребления лямбда-выражений	296
6.2.2. Синтаксис лямбда-выражений	298
6.2.3. Функциональные интерфейсы	300
6.2.4. Ссылки на методы	302
6.2.5. Ссылки на конструкторы	305
6.2.6. Область видимости переменных	306
6.2.7. Обработка лямбда-выражений	308
6.2.8. Еще о компараторах	311
6.3. Внутренние классы	312
6.3.1. Доступ к состоянию объекта с помощью внутреннего класса	313

6.3.2. Специальные синтаксические правила для внутренних классов	316
6.3.3. О пользе, необходимости и безопасности внутренних классов	317
6.3.4. Локальные внутренние классы	320
6.3.5. Доступ к конечным переменным из внешних методов	321
6.3.6. Анонимные внутренние классы	322
6.3.7. Статические внутренние классы	325
6.4. Загрузчики служб	329
6.5. Прокси-классы	331
6.5.1. О применении прокси-классов	332
6.5.2. Создание прокси-объектов	332
6.5.3. Свойства прокси-классов	336
Глава 7. Исключения, утверждения и протоколирование	339
7.1. Обработка ошибок	340
7.1.1. Классификация исключений	341
7.1.2. Объявление проверяемых исключений	343
7.1.3. Порядок генерирования исключений	345
7.1.4. Создание классов исключений	347
7.2. Перехват исключений	348
7.2.1. Перехват одного исключения	348
7.2.2. Перехват нескольких исключений	350
7.2.3. Повторное генерирование и связывание исключений в цепочку	351
7.2.4. Блок оператора finally	352
7.2.5. Оператор try с ресурсами	355
7.2.6. Анализ элементов трассировки стека	356
7.3. Рекомендации по обработке исключений	361
7.4. Применение утверждений	364
7.4.1. Понятие утверждения	364
7.4.2. Разрешение и запрет утверждений	365
7.4.3. Проверка параметров с помощью утверждений	366
7.4.4. Документирование предположений с помощью утверждений	367
7.5. Протоколирование	368
7.5.1. Элементарное протоколирование	369
7.5.2. Усовершенствованное протоколирование	369
7.5.3. Смена диспетчера протоколирования	371
7.5.4. Локализация	373
7.5.5. Обработчики протоколов	374
7.5.6. Фильтры	378
7.5.7. Средства форматирования	378
7.5.8. “Рецепт” протоколирования	378
7.6. Рекомендации по отладке программ	387
Глава 8. Обобщенное программирование	393
8.1. Назначение обобщенного программирования	394
8.1.1. Преимущества параметров типа	394
8.1.2. На кого рассчитано обобщенное программирование	395
8.2. Определение простого обобщенного класса	396
8.3. Обобщенные методы	398
8.4. Ограничения на переменные типа	399
8.5. Обобщенный код и виртуальная машина	402
8.5.1. Стирание типов	402
8.5.2. Преобразование обобщенных выражений	403

8.5.3. Преобразование обобщенных методов	404
8.5.4. Вызов унаследованного кода	406
8.6. Ограничения и пределы обобщений	407
8.6.1. Параметрам типа нельзя приписывать простые типы	407
8.6.2. Во время выполнения можно запрашивать только базовые типы	407
8.6.3. Массивы параметризованных типов недопустимы	408
8.6.4. Предупреждения о переменном числе аргументов	409
8.6.5. Нельзя создавать экземпляры переменных типа	410
8.6.6. Нельзя строить обобщенные массивы	410
8.6.7. Переменные типа в статическом контексте обобщенных классов недействительны	412
8.6.8. Нельзя генерировать или перехватывать экземпляры обобщенного класса в виде исключений	412
8.6.9. Преодоление ограничения на обработку проверяемых исключений	413
8.6.10. Остерегайтесь конфликтов после стирания типов	415
8.7. Правила наследования обобщенных типов	416
8.8. Подстановочные типы	417
8.8.1. Понятие подстановочного типа	418
8.8.2. Ограничения супертипа на подстановки	419
8.8.3. Неограниченные подстановки	422
8.8.4. Захват подстановок	423
8.9. Рефлексия и обобщения	425
8.9.1. Обобщенный класс Class	425
8.9.2. Сопоставление типов с помощью параметров Class<T>	427
8.9.3. Сведения об обобщенных типах в виртуальной машине	427
8.9.4. Литералы типов	431

Глава 9. Коллекции 437

9.1. Каркас коллекций в Java	437
9.1.1. Разделение интерфейсов и реализаций коллекций	438
9.1.2. Интерфейс Collection	440
9.1.3. Итераторы	441
9.1.4. Обобщенные служебные методы	443
9.2. Интерфейсы в каркасе коллекций Java	446
9.3. Конкретные коллекции	448
9.3.1. Связные списки	450
9.3.2. Списочные массивы	458
9.3.3. Хеш-множества	459
9.3.4. Древовидные множества	463
9.3.5. Одно- и двухсторонние очереди	467
9.3.6. Очереди по приоритету	468
9.4. Отображения	470
9.4.1. Основные операции над отображениями	470
9.4.2. Обновление записей в отображении	473
9.4.3. Представления отображений	474
9.4.4. Слабые хеш-отображения	476
9.4.5. Связные хеш-множества и отображения	477
9.4.6. Перечислимые множества и отображения	478
9.4.7. Хеш-отображения идентичности	479
9.5. Представления и оболочки	481
9.5.1. Мелкие коллекции	481
9.5.2. Поддиапазоны	482
9.5.3. Немодифицируемые представления	483

9.5.4. Синхронизированные представления	484
9.5.5. Проверяемые представления	485
9.5.6. О необязательных операциях	485
9.6. Алгоритмы	489
9.6.1. Назначение обобщенных алгоритмов	490
9.6.2. Сортировка и перетасовка	491
9.6.3. Двоичный поиск	493
9.6.4. Простые алгоритмы	495
9.6.5. Групповые операции	496
9.6.6. Взаимное преобразование коллекций и массивов	497
9.6.7. Написание собственных алгоритмов	498
9.7. Унаследованные коллекции	499
9.7.1. Класс Hashtable	500
9.7.2. Перечисления	500
9.7.3. Таблицы свойств	501
9.7.4. Стеки	504
9.7.5. Битовые множества	505
Глава 10. Программирование графики	509
10.1. История развития инструментальных средств для разработки GUI на Java	509
10.2. Отображение фреймов	511
10.2.1. Создание фрейма	511
10.2.2. Свойства фрейма	513
10.3. Отображение данных в компоненте	517
10.3.1. Двухмерные формы	521
10.3.2. Окрашивание цветом	528
10.3.3. Применение шрифтов	530
10.3.4. Воспроизведение изображений	536
10.4. Обработка событий	537
10.4.1. Общее представление об обработке событий	537
10.4.2. Пример обработки событий от щелчков на экранных кнопках	539
10.4.3. Краткое обозначение приемников событий	543
10.4.4. Классы адаптеров	544
10.4.5. Действия	546
10.4.6. События от мыши	551
10.4.7. Иерархия событий в библиотеке AWT	557
10.5. Прикладной интерфейс Preferences API	560
Глава 11. Компоненты пользовательского интерфейса в Swing	567
11.1. Библиотека Swing и проектный шаблон “модель–представление–контроллер”	568
11.2. Введение в компоновку пользовательского интерфейса	572
11.2.1. Диспетчеры компоновки	572
11.2.2. Граничная компоновка	574
11.2.3. Сеточная компоновка	576
11.3. Ввод текста	577
11.3.1. Текстовые поля	578
11.3.2. Метки и пометка компонентов	580
11.3.3. Поля для ввода пароля	581
11.3.4. Текстовые области	582
11.3.5. Панели прокрутки	583

11.4. Компоненты для выбора разных вариантов	585
11.4.1. Флажки	585
11.4.2. Кнопки-переключатели	588
11.4.3. Границы	592
11.4.4. Комбинированные списки	594
11.4.5. Регулируемые ползунки	598
11.5. Меню	604
11.5.1. Создание меню	605
11.5.2. Пиктограммы в пунктах меню	607
11.5.3. Пункты меню с флажками и кнопками-переключателями	608
11.5.4. Всплывающие меню	609
11.5.5. Клавиши быстрого доступа и оперативные клавиши	611
11.5.6. Разрешение и запрет доступа к пунктам меню	613
11.5.7. Панели инструментов	618
11.5.8. Всплывающие подсказки	620
11.6. Расширенные средства компоновки	621
11.6.1. Диспетчер сеточно-контейнерной компоновки	622
11.6.2. Специальные диспетчеры компоновки	632
11.7. Диалоговые окна	636
11.7.1. Диалоговые окна для выбора разных вариантов	636
11.7.2. Создание диалоговых окон	641
11.7.3. Обмен данными	645
11.7.4. Диалоговые окна для выбора файлов	651
Глава 12. Параллелизм	661
12.1. Назначение потоков исполнения	662
12.2. Состояния потоков исполнения	667
12.2.1. Новые потоки исполнения	667
12.2.2. Исполняемые потоки	667
12.2.3. Блокированные и ожидающие потоки исполнения	668
12.2.4. Завершенные потоки исполнения	669
12.3. Свойства потоков исполнения	670
12.3.1. Прерывание потоков исполнения	670
12.3.2. Потокосы демоны	673
12.3.3. Именованные потоки исполнения	674
12.3.4. Обработчики необрабатываемых исключений	674
12.3.5. Приоритеты потоков исполнения	675
12.4. Синхронизация	676
12.4.1. Пример состояния гонок	676
12.4.2. Объяснение причин, приводящих к состоянию гонок	679
12.4.3. Объекты блокировки	681
12.4.4. Объекты условий	684
12.4.5. Ключевое слово synchronized	689
12.4.6. Синхронизированные блоки	693
12.4.7. Принцип монитора	694
12.4.8. Поля и переменные типа volatile	695
12.4.9. Поля и переменные типа final	697
12.4.10. Атомарность операций	697
12.4.11. Взаимные блокировки	699
12.4.12. Локальные переменные в потоках исполнения	702
12.4.13. Причины, по которым методы stop () и suspend () не рекомендованы к применению	703

12.5. Потокобезопасные коллекции	705
12.5.1. Блокирующие очереди	705
12.5.2. Эффективные отображения, множества и очереди	712
12.5.3. Атомарное обновление записей в отображениях	713
12.5.4. Групповые операции над параллельными хеш-отображениями	717
12.5.5. Параллельные представления множеств	719
12.5.6. Массивы, копируемые при записи	720
12.5.7. Алгоритмы обработки параллельных массивов	720
12.5.8. Устаревшие потокобезопасные коллекции	721
12.6. Задачи и пулы потоков исполнения	722
12.6.1. Интерфейсы Callable и Future	723
12.6.2. Исполнители	725
12.6.3. Управление группами задач	727
12.6.4. Архитектура вилочного соединения	732
12.7. Асинхронные вычисления	735
12.7.1. Завершаемые будущие действия	735
12.7.2. Составление завершаемых будущих действий	738
12.7.3. Длительные задачи в обратных вызовах пользовательского интерфейса	744
12.8. Процессы	751
12.8.1. Построение процесса	752
12.8.2. Выполнение процесса	753
12.8.3. Дескрипторы процессов	755
Глава 13. Библиотека JavaFX	759
13.1. Отображение данных на сцене	759
13.1.1. Первое JavaFX-приложение	759
13.2.2. Рисование геометрических форм	763
13.2.3. Текст и изображения	767
13.3. Обработка событий	771
13.3.1. Реализация обработчиков событий	772
13.3.2. Реагирование на изменения свойств	772
13.3.3. События от мыши и клавиатуры	775
13.4. Компоновка	782
13.4.1. Панели компоновки	783
13.4.2. Язык FXML	789
13.4.3. Стилиевые таблицы CSS	795
13.5. Элементы управления пользовательского интерфейса	800
13.5.1. Элементы управления вводом текста	800
13.5.2. Элементы управления выбором разных вариантов	804
13.5.3. Меню	811
13.5.4. Простые диалоговые окна	819
13.5.5. Специальные элементы управления	828
13.6. Свойства и привязки	832
13.6.1. Свойства в библиотеке JavaFX	832
13.6.2. Привязки	835
13.7. Длительные задачи в обратных вызовах пользовательского интерфейса	841
Приложение А. Ключевые слова Java	849
Предметный указатель	851