

Станислав Давыдов
Алексей Ефимов



IntelliJ IDEA

Профессиональное программирование на Java

- Описание интерфейса и настроек
- Эффективное создание и модификация кода
- Анализ кода на возможные ошибки
- Удобная работа с XML и компонентами J2EE
- Создание плагинов для IntelliJ IDEA

Наиболее
полное
руководство

+CD



В ПОДЛИННИКЕ®

УДК 681.3.068+800java
ББК 32.973.26-018.2
Д13

Давыдов С. В., Ефимов А. А.

Д13 IntelliJ IDEA. Профессиональное программирование на Java. —
СПб.: БХВ-Петербург, 2005. — 800 с.: ил.

ISBN 5-94157-607-2

Книга представляет собой исчерпывающее руководство программиста по использованию среды разработки IntelliJ IDEA для создания Java-приложений. Рассмотрены основные особенности работы с IntelliJ IDEA: создание кода на основе информации о классах, редактор с мощными вспомогательными возможностями, встроенная поддержка рефакторинга кода, средства структурного анализа и проверки кода на наличие логических ошибок, потенциально опасных конструкций и возможных способов улучшения, поддержка J2EE-приложений, поддержка XML, визуальный редактор пользовательского интерфейса, встроенная поддержка unit-тестирования с использованием JUnit, поддержка основных систем контроля версий (SourceSafe, CVS, StarTeam), полная поддержка Java 5, поддержка плагинов. Особое внимание уделено практическим приемам работы с IntelliJ IDEA. Компакт-диск содержит все необходимое для разработки Java-приложений.

Для программистов

УДК 681.3.068+800java
ББК 32.973.26-018.2

Группа подготовки издания:

Главный редактор	<i>Екатерина Кондукова</i>
Зам. гл. редактора	<i>Игорь Шишигин</i>
Зав. редакцией	<i>Григорий Добин</i>
Редактор	<i>Алия Амирова</i>
Компьютерная верстка	<i>Натальи Караваевой</i>
Корректор	<i>Зинаида Дмитриева</i>
Дизайн серии	<i>Инны Тачиной</i>
Оформление обложки	<i>Игоря Цырульникова</i>
Зав. производством	<i>Николай Тверских</i>

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 22.04.05.

Формат 70×100^{1/16}. Печать офсетная. Усл. печ. л. 64,5.

Тираж 3000 экз. Заказ №

"БХВ-Петербург", 194354, Санкт-Петербург, ул. Есенина, 5Б.

Санитарно-эпидемиологическое заключение на продукцию
№ 77.99.02.953 Д.006421.11.04 от 11.11.2004 г. выдано Федеральной службой
по надзору в сфере защиты прав потребителей и благополучия человека.

Отпечатано с готовых диапозитивов
в ГУП "Типография "Наука"
199034, Санкт-Петербург, 9 линия, 12

ISBN 5-94157-607-2

© Давыдов С. В., Ефимов А. А., 2005
© Оформление, издательство "БХВ-Петербург", 2005

Оглавление

Предисловие	15
От авторов	15
Для кого эта книга	16
Структура книги	16
Благодарности	18
Замечания и предложения	18
ЧАСТЬ I. ОБЩИЕ СВЕДЕНИЯ ОБ INTELLIJ IDEA.....	19
Глава 1. Прошлое и настоящее.....	21
1.1. Идеология работы с файлами	22
1.2. Логическая модель проекта	23
1.3. Идеология работы с пользователями	24
Глава 2. Установка и запуск.....	25
2.1. Регистрация	25
2.2. Комплект поставки	26
2.2.1. Установка под Windows	26
2.2.2. Установка под Linux	26
2.2.3. Установка под Mac OS X	27
2.2.4. Установка под Generic Unix	27
2.2.5. Структура директорий	27
2.3. Запуск	28
2.4. Директории настроек и временной информации	28
2.4.1. Директория с настройками	28
2.4.2. Директория с временными файлами	29

Глава 3. Интерфейс	30
3.1. Организация рабочего пространства приложения	30
3.2. Меню.....	32
3.2.1. Меню <i>File (Файл)</i>	32
3.2.2. Меню <i>Edit (Редактирование)</i>	42
3.2.3. Меню <i>Search (Поиск)</i>	50
3.2.4. Меню <i>View (Просмотр)</i>	62
3.2.5. Меню <i>Go To (Перейти к)</i>	71
3.2.6. Меню <i>Code (Код)</i>	75
3.2.7. Меню <i>Analyze (Анализ)</i>	86
3.2.8. Меню <i>Refactor (Рефакторинг)</i>	87
3.2.9. Меню <i>Build (Сборка)</i>	89
3.2.10. Меню <i>Run (Запуск)</i>	92
3.2.11. Меню <i>Tools (Инструменты)</i>	98
3.2.12. Меню <i>Window (Окно)</i>	108
3.2.13. Меню <i>Help (Помощь)</i>	110
3.3. Панель инструментов (<i>Tool Bar</i>).....	115
3.4. Инструментальные окна (<i>Tool Window</i>)	116
3.4.1. Режимы отображения	117
3.4.2. Состояния окна.....	118
3.4.3. Статусы окна	118
3.4.4. Размещение кнопок вызова.....	118
3.4.5. Дополнительные команды управления	119
3.4.6. Настройки инструментальных окон	119
3.4.7. Поиск в инструментальных окнах	119
3.4.8. Базовые инструментальные окна	119
3.5. Панель статуса (<i>Status Bar</i>).....	141
3.6. Окна редактирования.....	142
3.6.1. Закладки редакторов.....	142
3.6.2. Поля редактора.....	144
3.6.3. Дополнительные визуальные элементы	147
3.6.4. Команды редактора.....	147
Глава 4. Настройки.....	153
4.1. <i>IDE Settings (Настройки IDE)</i>	155
4.1.1. Диалог настроек <i>General (Общие)</i>	156
4.1.2. Диалог настроек <i>Appearance (Внешний вид)</i>	158
4.1.3. Диалог настроек <i>Editor (Редактор)</i>	161
4.1.4. Диалог настроек <i>Code Completion (Завершение кода)</i>	166
4.1.5. Диалог настроек <i>Errors (Ошибки)</i>	170
4.1.6. Диалог настроек <i>Colors & Fonts (Цвета и Шрифты)</i>	173
4.1.7. Диалог настроек <i>Global Code Style (Глобальные стили кода)</i>	179

4.1.8. Диалог настроек <i>JDK & Global Libraries</i> (<i>JDK и Глобальные библиотеки</i>)	195
4.1.9. Диалог настроек <i>Debugger</i> (<i>Отладчик</i>)	199
4.1.10. Диалог настройки <i>Resources</i> (<i>Ресурсы</i>)	202
4.1.11. Диалог настроек <i>File Types</i> (<i>Типы файлов</i>)	203
4.1.12. Диалог настроек <i>Local History</i> (<i>Локальная история</i>)	207
4.1.13. <i>Live Templates</i> (<i>"Живые шаблоны"</i>)	208
4.1.14. Диалог настройки <i>File Templates</i> (<i>Шаблоны файлов</i>)	221
4.1.15. Диалог настройки <i>Keypmap</i> (<i>Сочетания клавиши</i>)	228
4.1.16. Диалог настройки <i>External Tools</i> (<i>Внешние программы</i>)	234
4.1.17. Диалог настройки <i>Path Variables</i> (<i>Местоположения</i>)	241
4.1.18. Диалог <i>TODO</i> (<i>Заметки</i>)	242
4.1.19. Диалог настройки <i>Plugins</i> (<i>Плагины</i>)	247
4.1.20. Диалог настройки <i>Application Servers</i> (<i>Серверы приложений</i>)	250
4.1.21. Диалог настроек <i>Intention Power Pack</i> (<i>Дополнительный пакет "подсказок"</i>)	255
4.2. Настройки проекта	258
4.2.1. Диалог настройки <i>Paths</i> (<i>Пути</i>)	258
4.2.2. Диалог настройки <i>Compiler</i> (<i>Компилятор</i>)	259
4.2.3. Диалог настройки <i>Version Control</i> (<i>Контроль версий</i>)	263
4.2.4. Диалог настройки <i>Project Code Style</i> (<i>Стиль кода проекта</i>)	264
4.2.5. GUI Designer	266
ЧАСТЬ II. СОЗДАНИЕ КОДА	267
Глава 5. Управление проектами	269
5.1. Структура проекта	269
5.2. Создание проекта	269
5.2.1. Выбор JDK	270
5.2.2. Выбор количества модулей	271
5.2.3. Выбор типа модуля	271
5.3. Редактирование свойств модуля	289
5.3.1. Закладка <i>Paths</i> (<i>Пути</i>)	290
5.3.2. Закладка <i>Libraries</i> (<i>Библиотеки</i>)	293
5.3.3. Закладка <i>Dependencies</i> (<i>Зависимости</i>)	296
5.3.4. Закладка <i>Order/Export</i> (<i>Порядок/Экспорт</i>)	296
5.3.5. Закладка <i>Javadoc</i>	298
5.3.6. Закладка <i>Web Module Settings</i> (<i>Свойства Web-модуля</i>)	299
5.3.7. Закладка <i>J2EE Build Settings</i> (<i>Настройка сборки J2EE</i>)	301
5.3.8. Закладка <i>EJB Module Settings</i> (<i>Свойства EJB-модуля</i>)	304
5.3.9. <i>J2EE Application Module Settings</i> (<i>Свойства модуля J2EE приложения</i>)	305
5.4. Создание проекта из имеющихся файлов	307

Глава 6. Работа с кодом.....	311
6.1. Работа с редактором.....	312
6.1.1. Завершение кода (completion)	329
6.1.2. Подсветка.....	334
6.1.3. "Подсказки" (intention actions).....	336
6.1.4. Форматирование кода.....	341
6.1.5. Информация о коде	342
6.1.6. Буфер обмена.....	343
6.1.7. Сокращенное представление кода	343
6.2. Навигация по коду	344
6.2.1. Класс.....	344
6.2.2. Файл	344
6.2.3. Метод или поле	345
6.2.4. Переход к определению, имплементации, определению типа или базовому методу.....	345
6.2.5. Перемещение по коду "вперед-назад"	345
6.2.6. Список последних файлов.....	346
6.2.7. Закладки.....	346
6.2.8. Заметки (TODO)	346
6.2.9. Иерархия	346
6.3. Генерация кода	347
6.3.1. Шаблоны файлов	347
6.3.2. "Живые шаблоны"	347
6.3.3. Создание одготипных классов, интерфейсов и перечислений.....	347
6.3.4. Создание одготипных файлов.....	348
6.3.5. Создание методов <i>get</i> и <i>set</i>	348
6.3.6. Автоматическое создание конструкторов.....	348
6.3.7. Создание методов.....	348
6.3.8. Автоматическое создание блоков кода	348
6.3.9. Строки импорта	349
6.3.10. Комментарии	349
6.4. Анализ кода.....	350
6.4.1. Проверка кода (inspections).....	350
6.4.2. Анализ зависимостей.....	386
6.4.3. Анализ дубликатов	390
6.5. Структурный поиск.....	392
6.5.1. Редактирование переменных.....	393
6.5.2. Доступные шаблоны	395
6.5.3. Результаты поиска.....	396
6.5.4. Структурная замена	396
Глава 7. Создание интерфейса пользователя	398
7.1. Формы.....	398
7.1.1. Создание формы	399

7.1.2. Свойства компонент.....	415
7.1.3. Размещение компонент.....	420
7.2. Создание кода.....	425
7.3. Добавление собственных компонент.....	427
7.4. Дополнительные возможности.....	430
7.4.1. Настройка общих свойств визуального редактора.....	430
7.4.2. <i>Data Binding Wizard (Мастер связей)</i>	433
7.4.3. Компиляция форм.....	436
7.4.4. Интернализация (i18n).....	437
Глава 8. Рефакторинг.....	442
8.1. Рефакторинг <i>Rename (Переименовать)</i>	443
8.1.1. Переименование директорий.....	443
8.1.2. Переименование файлов.....	444
8.1.3. Переименование пакетов.....	444
8.1.4. Переименование классов.....	446
8.1.5. Переименование методов.....	446
8.1.6. Переименование полей класса.....	447
8.1.7. Переименование параметров вызова метода.....	448
8.1.8. Переименование локальных переменных.....	449
8.2. Рефакторинг <i>Change Signature (Изменить сигнатуру)</i>	449
8.3. Рефакторинг <i>Make Method Static (Сделать метод статичным)</i>	454
8.3.1. Простой случай.....	454
8.3.2. Сложный случай.....	456
8.4. Рефакторинг <i>Convert To Instance Method (Конвертировать в обычный метод)</i>	457
8.5. Рефакторинг <i>Move (Переместить)</i>	458
8.5.1. Перемещение директорий.....	459
8.5.2. Перемещение файлов.....	459
8.5.3. Перемещение пакетов.....	460
8.5.4. Перемещение классов.....	462
8.5.5. Перемещение членов класса.....	465
8.6. Рефакторинг <i>Copy (Копировать)</i>	467
8.6.1. Копирование директорий.....	467
8.6.2. Копирование файлов.....	468
8.6.3. Копирование классов.....	468
8.7. Рефакторинг <i>Safe Delete (Безопасное удаление)</i>	469
8.7.1. Удаление класса.....	469
8.7.2. Удаление метода.....	470
8.7.3. Удаление поля класса.....	470
8.8. Рефакторинг <i>Extract Method (Выделить метод)</i>	471
8.9. Рефакторинг <i>Replace Method Code Duplicates (Заменить методом повторяющийся код)</i>	474

8.10. Рефакторинг <i>Introduce Variable (Ввести переменную)</i>	476
8.11. Рефакторинг <i>Introduce Field (Ввести поле класса)</i>	477
8.11.1. Локальная переменная статического метода	477
8.11.2. Локальная переменная обычного метода	479
8.12. Рефакторинг <i>Introduce Constant (Ввести константу)</i>	480
8.13. Рефакторинг <i>Introduce Parameter (Ввести параметр)</i>	481
8.14. Рефакторинг <i>Extract Interface (Выделить интерфейс)</i>	482
8.15. Рефакторинг <i>Extract Superclass (Выделить базовый класс)</i>	485
8.16. Рефакторинг <i>Use Interface Where Possible</i> <i>(Использовать интерфейс, где это возможно)</i>	488
8.17. Рефакторинг <i>Pull Members Up (Перемещение членов класса вверх)</i>	490
8.18. Рефакторинг <i>Push Members Down (Перемещение членов класса вниз)</i>	492
8.19. Рефакторинг <i>Replace Inheritance With Delegation</i> <i>(Заменить наследование делегированием)</i>	495
8.20. Рефакторинг <i>Inline (Подстановка в код)</i>	497
8.20.1. Подстановка для метода	497
8.20.2. Подстановка для статического поля класса	499
8.21. Рефакторинг <i>Convert Anonynous to Inner</i> <i>(Конвертировать анонимный класс во внутренний)</i>	500
8.22. Рефакторинг <i>Encapsulate Fields (Инкапсулировать поля)</i>	502
8.23. Рефакторинг <i>Replace Temp with Query</i> <i>(Заменить временное использование вызовом)</i>	505
8.24. Рефакторинг <i>Replace Constructor With Factory Method</i> <i>(Заменить конструктор методом фабрики)</i>	507
8.25. Рефакторинг <i>Generify (Генерификация)</i>	508

Глава 9. Компиляция, запуск, отладка.....512

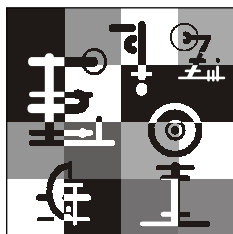
9.1. Компиляция	512
9.1.1. Компиляция Java-классов	512
9.1.2. Компиляция RMI.....	514
9.1.3. Валидация JSP	514
9.2. Типы приложений	514
9.2.1. Java-приложения (закладка <i>Application</i>)	516
9.2.2. Апплеты (закладка <i>Applet</i>)	517
9.2.3. Тесты (закладка <i>JUnit</i>)	520
9.2.4. Удаленные приложения (закладка <i>Remote</i>)	524
9.2.5. Web-приложения для Tomcat (закладка <i>Tomcat Server</i>)	526
9.2.6. Приложения для JSR45-совместимых серверов (JSR-45 Compatible Server).....	530
9.2.7. Приложения для сервера WebLogic (закладка <i>WebLogic Instance</i>)	533
9.3. Запуск	538
9.3.1. Java-приложения (закладка <i>Application</i>)	538
9.3.2. Апплеты.....	540

9.3.3. Тесты JUnit	544
9.3.4. J2EE-приложения.....	546
9.4. Отладка	549
9.4.1. Точки останова.....	550
9.4.2. Пошаговая отладка	561
9.4.3. Информация о потоках (закладка <i>Threads</i>).....	569
9.4.4. Информация о переменных (закладка <i>Frame</i>).....	571
9.4.5. <i>Watches (Наблюдения)</i>	576
9.4.6. Вычисление выражений.....	577
9.4.7. Функция <i>HotSwap</i>	579
Глава 10. J2EE	580
10.1. Работа с XML.....	580
10.1.1. Редактирование XML	580
10.1.2. Поддержка типизации	582
10.2. Серверы приложений.....	583
10.3. Работа с EJB.....	583
10.3.1. Создание EJB.....	585
10.3.2. Установка связей между EJB	598
10.4. Web-приложения	600
10.4.1. Сервлеты	603
10.4.2. Фильтры	605
10.4.3. JSP.....	606
10.5. Приложения J2EE.....	608
Глава 11. Поддержка новых возможностей Java 5.....	610
11.1. Generics.....	611
11.2. Цикл <i>for</i> для коллекций	612
11.3. Auto(un)boxing	613
11.4. Перечислимый тип <i>enum</i>	614
11.5. Переменное число параметров метода.....	616
11.6. Статические импорты	617
11.7. Аннотации	618
ЧАСТЬ III. ИНТЕГРАЦИЯ.....	621
Глава 12. Системы контроля версий	623
12.1. Утилита Diff.....	623
12.1.1. Сравнение двух файлов.....	623
12.1.2. Сравнение частей файлов.....	624
12.1.3. Представление результатов сравнения	624

12.1.4. Дополнительные команды	625
12.2. Локальная система контроля версий.....	625
12.2.1. Версии файла.....	626
12.2.2. Версии директорий	629
12.2.3. Метки.....	630
12.3. CVS.....	631
12.3.1. Настройки	631
12.3.2. Работа с проектом.....	641
12.3.3. Визуальное выделение файлов	648
12.3.4. Команды.....	648
12.4. SourceSafe.....	665
12.4.1. Настройки	665
12.4.2. Визуальное выделение файлов	667
12.4.3. Команды.....	668
12.5. StarTeam.....	672
12.5.1. Настройки	672
12.5.2. Визуальное выделение файлов	672
12.5.3. Команды.....	672
12.6. Дополнительные команды.....	676
Глава 13. Утилита Ant.....	677
13.1. Создание и подключение файлов сборки.....	677
13.2. Редактирование файлов сборки	680
13.2.1. Список доступных команд и параметров.....	680
13.2.2. Вывод дополнительной информации	681
13.2.3. Проверка ошибок.....	682
13.2.4. Редактирование зависимостей	682
13.2.5. Использование свойств	683
13.3. Настройка параметров	683
13.3.1. Общие свойства.....	684
13.3.2. Параметры для передачи файлу сборки	684
13.3.3. Выполнение	684
13.3.4. Дополнительные библиотеки.....	685
13.3.5. Фильтры	686
13.4. Привязка целей Ant.....	687
13.5. Запуск целей Ant.....	689
13.6. Генерация файла сборки для проекта.....	690
Глава 14. Тестирование.....	692
14.1. Представление тестов в проекте	692
14.2. Запуск тестов.....	693
14.3. Просмотр результатов	696

Глава 15. Плагины	698
15.1. Общие сведения о плагинах.....	698
15.1.1. Сообщество разработчиков плагинов IDEA.....	698
15.2. Разработка плагинов.....	698
15.2.1. Подготовка IntelliJ IDEA к разработке плагинов.....	699
15.2.2. Общая компонентная модель.....	709
15.2.3. Диагностика ошибок.....	715
15.2.4. Сохранение и восстановление состояния компонент.....	718
15.2.5. Система действий в IntelliJ IDEA.....	723
15.2.6. Стандартные визуальные компоненты IntelliJ IDEA.....	731
15.2.7. Редактор IntelliJ IDEA.....	746
15.2.8. Виртуальная файловая система IntelliJ IDEA (VFS).....	763
15.2.9. Система контроля версий IntelliJ IDEA (VCS).....	766
15.2.10. Структурный программный интерфейс IntelliJ IDEA (PSI).....	768
15.3. Структура директорий плагина.....	771
15.4. Дескриптор плагина.....	772
15.4.1. Требования к дескрипторам.....	772
15.4.2. Общая информация о плагине.....	772
15.4.3. Описания компонент.....	774
15.4.4. Описание действий.....	774
15.5. Заключение.....	777
ЧАСТЬ IV. БЛИЖАЙШЕЕ БУДУЩЕЕ	779
Глава 16. IDEA 5	781
16.1. Поддержка XML, HTML, CSS.....	781
16.1.1. XML.....	781
16.1.2. HTML.....	782
16.1.3. CSS.....	782
16.2. Поддержка J2EE.....	782
16.3. Работа с проектом.....	783
16.4. Настройки.....	783
16.5. Рефакторинг.....	783
16.6. Системы контроля версий.....	784
16.7. Плагины и OpenAPI.....	784
16.8. Поддержка J2ME.....	785
16.9. Отладка.....	785
16.10. Дополнительные функции редактора.....	785
16.11. Прочие улучшения.....	785
Заключение	787
Список ссылок	789
Содержимое компакт-диска	791
Предметный указатель	792

Глава 3



Интерфейс

Пользовательский интерфейс IntelliJ IDEA представляет собой стандартную форму MDI¹-интерфейса, в котором общее окно соответствует одному проекту.

3.1. Организация рабочего пространства приложения

Общий вид окна проекта в IntelliJ IDEA представлен на рис. 3.1.

Кроме стандартного для большинства приложений меню, панели инструментов (ToolBar), панели статуса (StatusBar) и рабочей области, в IntelliJ IDEA есть интересное интерфейсное дополнение — *инструментальные окна* (ToolWindow — по аналогии с панелью инструментов — ToolBar). Инструментальные окна имеют кнопки вызова, расположенные по периметру рабочей области, с пиктограммой, подписью и числовым обозначением (последнее не обязательно). Если на эти кнопки нажать, рядом с ними откроются окошки с некоторой вспомогательной функциональностью (рис. 3.2). Подробнее об инструментальных окнах написано в *разделе 3.4*.

Обзор интерфейса IntelliJ IDEA мы начнем с меню, как с наиболее часто употребляемого списка команд, доступных в IntelliJ IDEA.

¹ MDI — Multi Document Interface — многодокументный интерфейс.

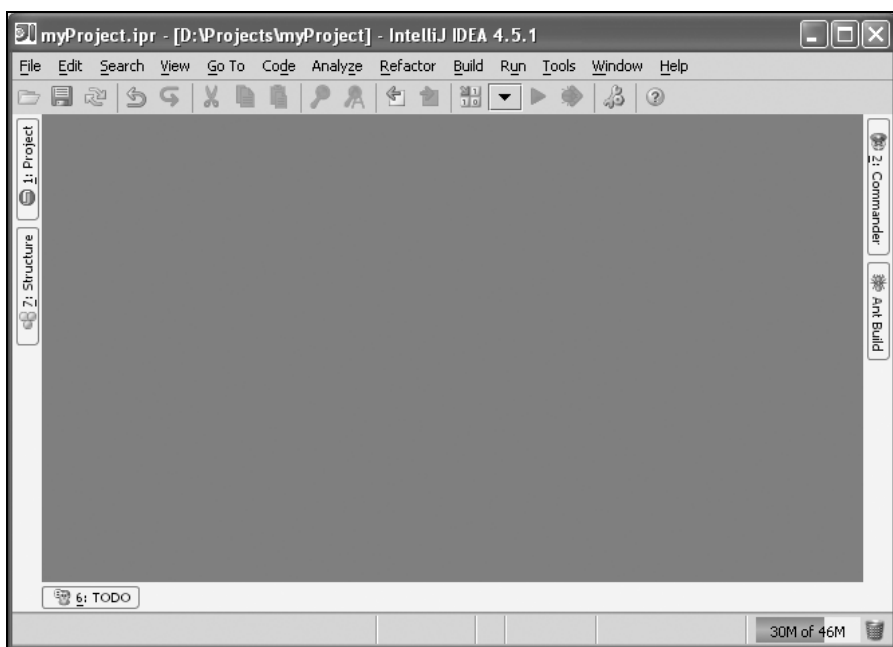


Рис. 3.1. Общий вид окна проекта в IntelliJ IDEA

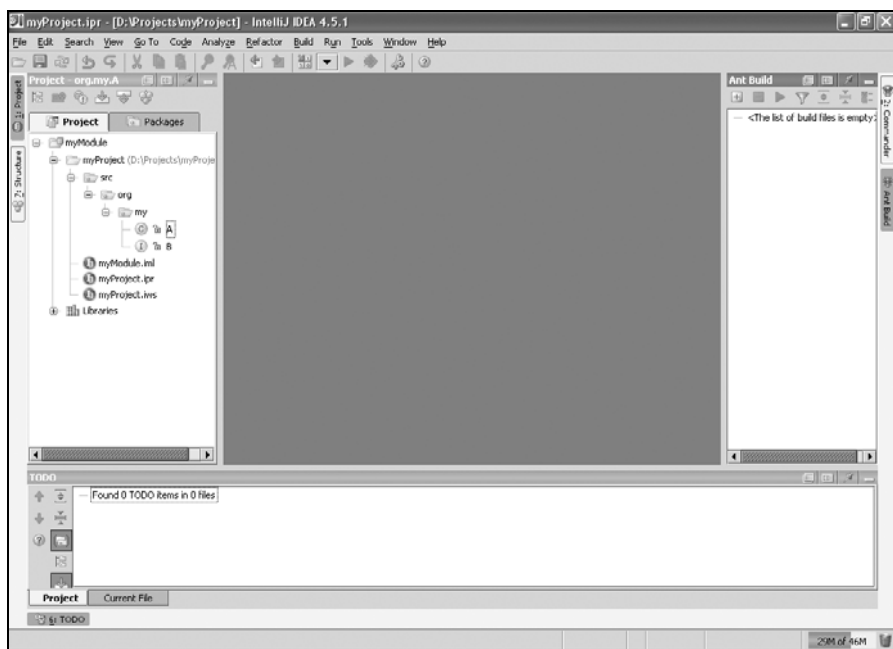


Рис. 3.2. Инструментальные окна

3.2. Меню

Главное меню IntelliJ IDEA состоит из следующих пунктов:

- **File** (Файл) — команды для работы с проектами, модулями, файлами;
- **Edit** (Редактирование) — стандартные команды для редактирования;
- **Search** (Поиск) — различные типы поиска и замены;
- **View** (Просмотр) — команды для отображения различных элементов интерфейса;
- **Go To** (Перейти) — команды навигации по проекту;
- **Code** (Код) — вспомогательные команды для написания кода;
- **Analyze** (Анализ) — команды для анализа кода;
- **Refactor** (Рефакторинг) — набор команд для проведения рефакторинга;
- **Build** (Сборка) — команды для компиляции и сборки проекта;
- **Run** (Запуск) — запуск и отладка приложений;
- **Tools** (Утилиты) — набор различных вспомогательных утилит;
- **Window** (Окно) — работа с окнами в IntelliJ IDEA;
- **Help** (Помощь) — контекстная помощь, справка о программе.

Пункты меню могут быть связаны с кнопками на панели инструментов, а также иметь *"горячие" клавиши* (hot keys) для быстрого вызова команд. В IntelliJ IDEA любой команде в меню можно назначить "горячую" клавишу. Для некоторых наиболее часто используемых команд "горячие" клавиши заданы по умолчанию. В описании таких команд мы будем указывать соответствующие им "горячие" клавиши в тексте сразу после названия.

Далее мы рассмотрим все приведенные пункты меню. Некоторые команды, присутствующие в меню, имеют весьма сложную функциональность, например, команды структурного поиска или рефакторинга. Таким командам посвящены отдельные главы или разделы, а в этой главе мы лишь приведем их краткое описание.

3.2.1. Меню *File* (Файл)

В меню **File** (Файл) представлены команды, с помощью которых можно выполнить некоторые действия с файлами — открыть, закрыть, вывести на печать и пр.

3.2.1.1. *New Project* (Новый проект)

Команда создания нового проекта (запускается мастер создания нового проекта, см. раздел 5.2).

3.2.1.2. *New Module (Новый модуль)*

Команда создания нового модуля в текущем проекте (запускается мастер создания нового модуля, см. раздел 5.3). Если ни одного проекта не открыто, эта команда недоступна.

3.2.1.3. *Open Project (Открыть проект)*

Команда открытия нового проекта из сохраненного на диске файла (рис. 3.3). Если один (или более) проект уже открыт, IntelliJ IDEA спросит, открывать ли новый проект в новом окне, либо в уже открытом.

3.2.1.4. *Open File (Открыть файл)*

Команда открытия файла в редакторе IntelliJ IDEA. Если открывается проектный файл, IntelliJ IDEA предложит открыть его как проект, либо как обычный файл.

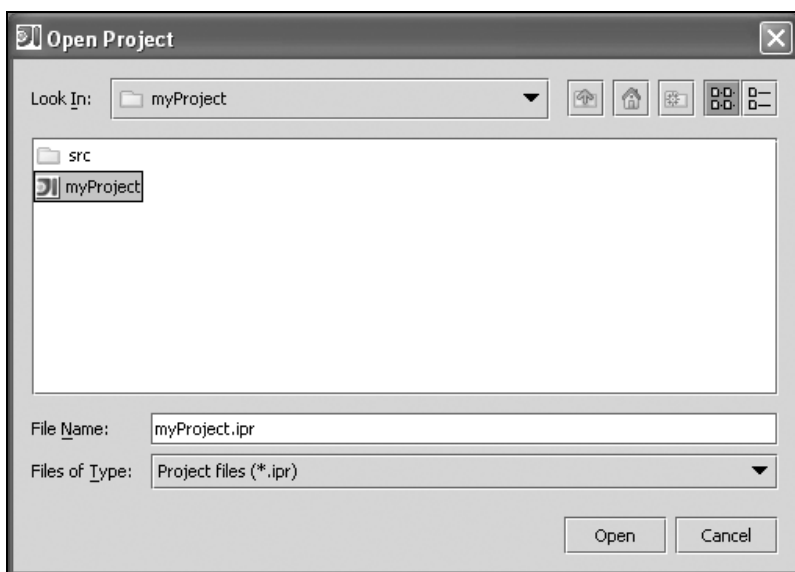


Рис. 3.3. Открытие проекта из сохраненного файла

Файлы проектов IntelliJ IDEA имеют расширение `ipr`. Если попытаться открыть такой файл с помощью команды **Open File** (Открыть файл), IntelliJ IDEA задаст вопрос о том, должен ли такой файл быть открытым как проект — полный аналог команды **Open Project** (Открыть проект), либо как обычный текстовый файл.

Надо заметить, что IntelliJ IDEA может открывать только те файлы, которые она "знает", т. е. те файлы, расширения которых описаны в типах файлов

в настройке **File Types** (Типы файлов) среди общих настроек IntelliJ IDEA (см. раздел 4.1.11).

3.2.1.5. **Reopen (Открыть заново)**

Подраздел меню **File** (Файл) содержит список проектов, с которыми вы недавно работали. IntelliJ IDEA запоминает последние 15 открытых проектов.

3.2.1.6. **Close Project (Закреть проект)**

Команда закрывает открытый проект. Если ни одного открытого проекта нет, команда недоступна.

3.2.1.7. **Close Active Editor (Закреть активную закладку редактора)**

"Горячие" клавиши: <Ctrl>+<F4>.

Команда закрывает закладку редактора, активную в данный момент (см. раздел 3.6). Этой команде соответствуют "горячие" клавиши: <Ctrl>+<F4>.

3.2.1.8. **Close All Editors (Закреть все окна редактора)**

Команда закрывает все открытые окна редактора.

3.2.1.9. **Close All Editors But Active (Закреть все окна редактора за исключением активного)**

Команда закрывает все открытые окна редактора за исключением активного.

3.2.1.10. **Команды разделения и объединения окон редактора**

Группа команд для разделения окон редактора насчитывает четыре команды:

- Split Tab Group Vertically** (Разделить группу закладок вертикально);
- Split Tab Group Horizontally** (Разделить группу закладок горизонтально);
- Move to Opposite Tab Group** (Переместить в противоположную группу закладок);
- Change Tab Group Orientation** (Изменить ориентацию группы закладок).

Более подробно эти команды будут описаны в разделе 3.6.

3.2.1.11. **Settings (Настройки)**

Команде соответствуют "горячие" клавиши: <Ctrl>+<Alt>+<S>.

Команда открывает диалог со всеми возможными настройками IntelliJ IDEA.

3.2.1.12. *Template Project Settings (Шаблон настроек проекта)*

Команда открывает диалог, с помощью которого можно задать настройки проекта "по умолчанию", т. е. те, что будут актуальны для вновь создаваемого проекта.

Подробнее об этих настройках написано в *разделе 4.2* и в *главе 5*.

3.2.1.13. *Save All (Сохранить все)*

"Горячие" клавиши: <Ctrl><S>.

Команда сохраняет все измененные документы, открытые в IntelliJ IDEA, включая настройки модулей и проектов.

3.2.1.14. *Export Settings (Экспорт настроек)*

Команда открывает диалог экспорта всех настроек IntelliJ IDEA (рис. 3.4), в котором можно выбрать те настройки, которые вы хотите экспортировать, а также место на диске, где должен быть создан файл с экспортированными настройками.



Рис. 3.4. Диалог экспорта настроек

В результате импорта в указанном месте создается JAR-файл, содержащий набор XML-файлов с настройками.

3.2.1.15. *Import Settings (Импорт настроек)*

Команда импортирования настройки IntelliJ IDEA из файла. Файл должен быть сохранен командой **Export Settings** (Экспорт настроек) (см. *раздел 3.2.1.14*).

3.2.1.16. Synchronize (Синхронизировать)

"Горячие" клавиши: <Ctrl>+<Alt>+<Y>.

Команда находит все файлы, которые были изменены извне, и перезагружает их. Обычно IntelliJ IDEA автоматически замечает изменившиеся файлы и перезагружает их, однако эта возможность доступна только при работе в операционных системах Windows и Linux. Так что команда иногда может оказаться полезной при работе с IntelliJ IDEA на других платформах, а также, если автоматическая синхронизация выключена (см. раздел 4.1.1).

3.2.1.17. Reload from Disk (Перезагрузить с диска)

Если некоторый файл был изменен внутри IntelliJ IDEA, но еще не был сохранен, с помощью этой команды можно загрузить с диска его предыдущую версию, при этом сделанные изменения будут утрачены.

3.2.1.18. Команды для работы с CVS

Группа команд для работы с CVS-репозиторием (Concurrent Version System — одна из версий системы контроля версий) содержит три команды:

- Check Out from CVS** (Извлечь из CVS). Команда извлечения файлов из CVS (в данном случае термин "check out" обозначает специфическую операцию для CVS). В случае, если среди этих файлов существует хотя бы один проект IntelliJ IDEA (т. е. файл с расширением ipr), IntelliJ IDEA предложит открыть этот проект;
- Import into CVS** (Импортировать в CVS). Команда позволяет импортировать существующий проект в CVS (в данном случае термин "import" также обозначает операцию, специфичную для CVS);
- Browse CVS Repository** (Просмотр CVS-репозитория). Аналог просмотра файловой системы, только для CVS.

Данные команды, несмотря на то, что они присутствуют в меню изначально (с момента установки IntelliJ IDEA), не являются базовыми. Эти команды добавлены в меню благодаря специальному плагину, входящему в поставку IntelliJ IDEA — CVS Integration (интеграция с CVS). Более подробно работа с CVS-репозиторием будет рассмотрена в разделе 12.3.2, а работа с плагином — в главе 15.

3.2.1.19. Export to HTML (Экспорт в HTML)

Команда позволяет экспортировать содержимое файла в HTML. Данная команда бывает очень полезной, если требуется опубликовать какие-либо исходные тексты на сайте в Интернете. При вызове этой команды открывается специальный диалог (рис. 3.5).

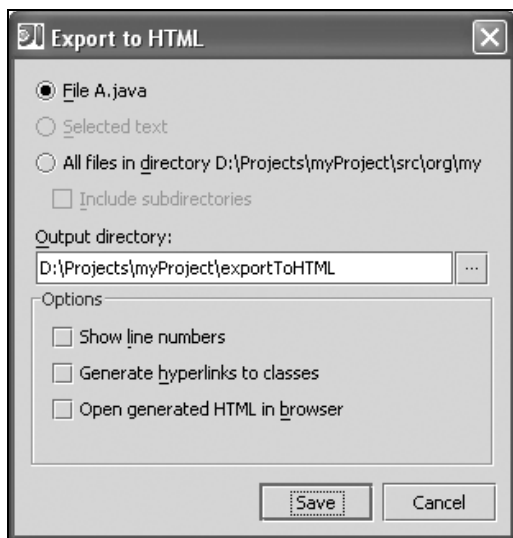


Рис. 3.5. Диалог экспорта в HTML

В диалоге можно задать следующие опции:

- объект экспорта:
 - текущий файл;
 - выделенный текст (эта опция активна, если в редакторе был выделен некоторый блок текста);
 - все файлы в директории (с возможностью включить все поддиректории, если они есть);
- путь на диске, куда должны быть записаны HTML-файлы;
- дополнительные настройки:
 - показывать номера строк (**Show line numbers**);
 - создавать гиперссылки на классы, которые встречаются в коде — в этом случае результатом экспорта будет полностью "кликабельная" (clickable) версия исходных файлов проекта с возможностью навигации между классами;
 - открыть сгенерированный HTML-код в браузере.

Самым главным достоинством данной операции является то, что сгенерированный HTML-код сохраняет все стилевые настройки, выделения и шрифты, имеющиеся в самой IntelliJ IDEA (см. раздел 4.1.7).

Кстати, при подготовке листингов к данной книге мы использовали именно эту функцию IntelliJ IDEA.

3.2.1.20. Print (Печать)

Команда вывода содержимого файла на печать. В случае вывода на печать также доступны расширенные настройки, показанные на рис. 3.7 и 3.8.

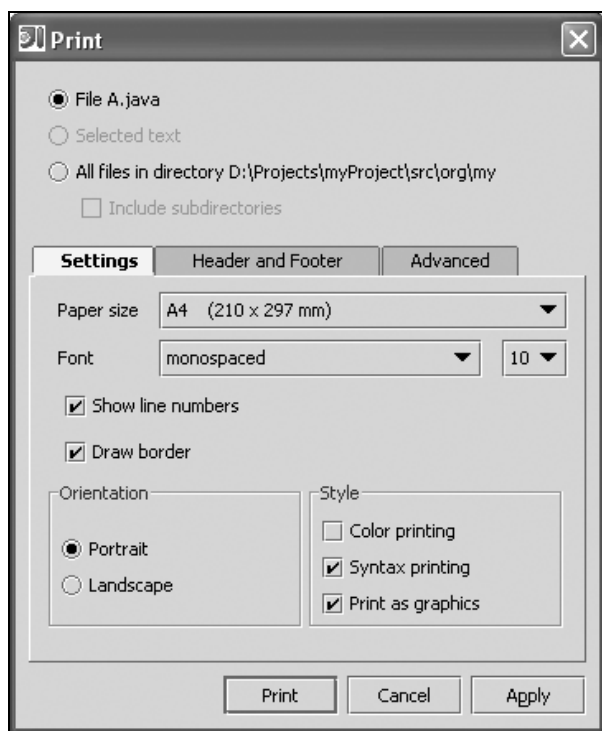


Рис. 3.6. Диалог вывода на печать: общие настройки

При активации команды вывода на печать появляется диалог, представленный на рис. 3.6.

Выбор содержимого для печати

Здесь так же, как и при экспорте в HTML, можно выбрать то, что будет напечатано:

- File** (Файл) — отдельный указанный файл;
- Selected text** (Выделенный текст) — блок текста, выделенный в редакторе IntelliJ IDEA;
- All files in directory** (Все файлы в директории) — все файлы в указанной директории. Для печати файлов из указанной директории можно указать дополнительную опцию **Include subfolders** (Включить поддиректории), чтобы были напечатаны все файлы из всех вложенных директорий.

Закладка *Settings* (Общие настройки)

Общие настройки печати состоят из следующих опций:

- Paper size** (Размер бумаги) — размер бумаги в принтере для печати;
- Font** (Шрифт) — тип и размер шрифта;
- Show line numbers** (Показывать номера строк) — выводить на печать также номера строк;
- Draw border** (Рисовать рамку) — содержимое файлов, выводимых на печать, будет окружено рамкой;
- Orientation** (Ориентация) — два варианта ориентации бумаги:
 - **Portrait** (Портрет) — портретная или книжная ориентация.
 - **Landscape** (Пейзаж) — пейзажная или альбомная ориентация.
- Style** (Стиль):
 - **Color printing** (Цветная печать) — цветовые стили кода будут сохранены при выводе на печать;
 - **Syntax printing** (Печать синтаксиса) — стилевые выделения синтаксических конструкций кода будут сохранены при выводе на печать;
 - **Print as graphics** (Печать содержимого, как графики) — данная опция бывает полезна, если принтер плохо поддерживает печатаемые символы (например, в том шрифте, который вы задали). Также эту опцию можно использовать в случае, если принтер не поддерживает локальные символы, встречающиеся в комментариях.

Закладка *Header and Footer* (Настройки "шапки" и "подвала")

"Шапка" и "подвал" — это постоянные элементы, которые будут вставлены в начало и конец печатной страницы (см. рис. 3.7).

Настройка "шапки" и "подвала" довольно проста. Предполагается, что "шапка" и "подвал" могут состоять максимум из двух строк. При этом если в "шапке" уже есть одна строка, то "подвалу" остается тоже только одна строка. Имеется возможность настроить только две строки, которые попадают в "шапку" и/или "подвал".

Каждая строка имеет следующие настройки:

- Text line** (Текстовая строка) — текст строки. Данная строка может содержать некоторые ключевые слова — макросы, используемые для подстановки некоторых значений. Всего есть два макроса:
 - **\$FILES** — название печатаемого файла;
 - **\$PAGES** — номер печатаемой страницы;

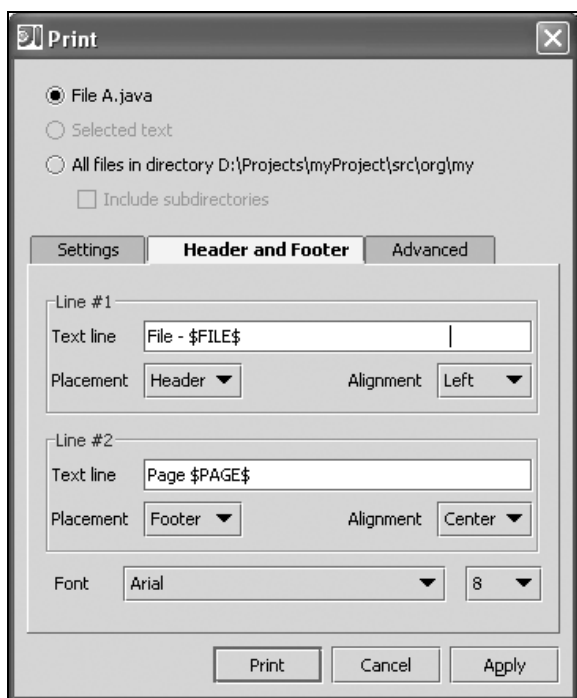


Рис. 3.7. Диалог вывода на печать: настройка "шапки" и "подвала"

- **Placement** (Размещение) — выпадающий список с двумя опциями:
 - **Header** ("Шапка") — размещать строку в "шапке";
 - **Footer** ("Подвал") — размещать строку в "подвале";
- **Alignment** (Выравнивание):
 - **Left** (Слева) — размещать строку с выравниванием по левому краю;
 - **Center** (По центру) — размещать строку по центру;
 - **Right** (По правому краю) — размещать строку с выравниванием по правому краю.

Для "шапки" и "подвала" также можно задать отдельные параметры шрифта (тип шрифта и его размер), которым они будут выводиться.

Закладка *Advanced* (Дополнительные настройки печати)

Дополнительные настройки печати представлены на рис. 3.8.

В качестве дополнительных настроек вывода на печать можно задать следующие опции:

- **Wrapping** (Перенос строк) — возможны два варианта переноса строк:
 - **No wrap** (Без переноса) — строки не переносятся;

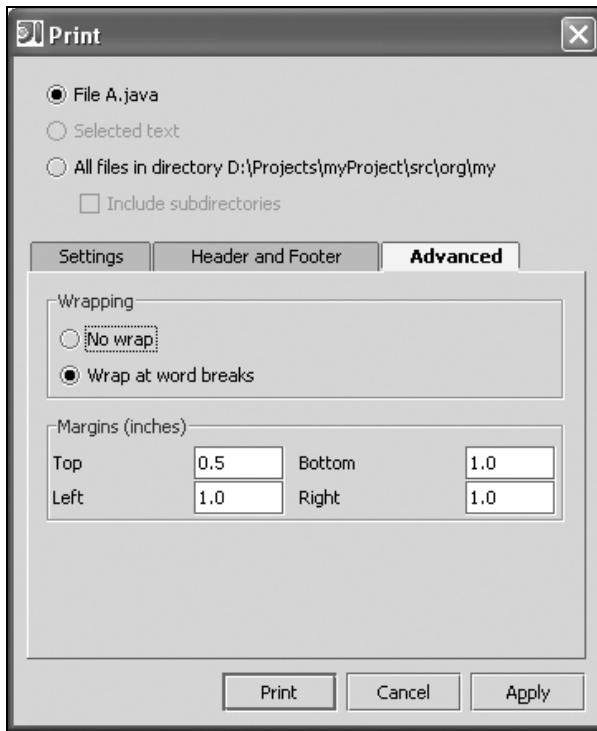


Рис. 3.8. Диалог вывода на печать: дополнительные настройки

- **Wrap at word breaks** (Перенос по разрывам строк) — IntelliJ IDEA будет делать переносы по разрывам в словах;
- ☐ **Margins** (Границы) — расстояние от края страницы до печатаемого содержимого;
 - **Top** (Верх) — верхняя граница;
 - **Left** (Слева) — левая граница;
 - **Bottom** (Низ) — нижняя граница;
 - **Right** (Право) — правая граница.

Размер отступа задается в дюймах.

3.2.1.21. *Exit* (Выход)

Данная команда завершает список команд в меню **File** (Файл). Она располагается последней в списке — там, где ей и положено находиться в уважающих себя приложениях ☺. Команда закрывает все открытые окна проектов и завершает работу IntelliJ IDEA.

3.2.2. Меню *Edit* (Редактирование)

Пункт меню **Edit** (Редактирование) содержит достаточно стандартный набор команд. Данные команды действуют как на текст в редакторе, так и на классы или файлы в инструментальных окнах **Project** (Проект) и **Commander** (Коммандер) (см. разделы 3.4.8.2, 3.4.8.3).

3.2.2.1. *Undo* (Отменить)

"Горячие" клавиши: <Ctrl>+<Z>.

Команда отмены последнего сделанного изменения. Под изменением понимается не только операция редактирования, но также и любая файловая операция, например создание или удаление файла.

3.2.2.2. *Redo* (Повторить)

"Горячие" клавиши: <Ctrl>+<Shift>+<Z>.

Команда повтора последнего изменения, отмененного с помощью команды **Undo** (Отменить) (см. раздел 3.2.2.1).

3.2.2.3. *Cut* (Вырезать)

"Горячие" клавиши: <Ctrl>+<X>.

Команда удаления в буфер обмена. Если в редакторе был выделен блок текста, удален будет именно он. Если выделения не было — удалена будет строка, в которой находится курсор. Вырезанная часть текста при этом попадает в буфер обмена.

Команда также действует на классы и файлы.

3.2.2.4. *Copy* (Копировать)

"Горячие" клавиши: <Ctrl>+<C>.

Команда копирования в буфер обмена. Если в редакторе был выделен блок текста, будет скопирован именно он. Если выделения не было — скопирована будет строка, в которой находится курсор.

Команда также действует на классы и файлы.

3.2.2.5. *Copy Path* (Копировать путь)

"Горячие" клавиши: <Ctrl>+<Shift>+<C>.

Команда копирует в буфер обмена путь текущего файла.

3.2.2.6. *Paste* (Вставить)

"Горячие" клавиши: <Ctrl>+<V>.

Команда вставки из буфера обмена. Команда действует как на текст в редакторе, так и на скопированные классы и файлы.

3.2.2.7. Paste... (Вставить...)

"Горячие" клавиши: <Ctrl>+<Shift>+<V>.

Команда расширенной вставки из буфера обмена, которая позволяет просмотреть все сделанные копирования и выбрать из них нужное (рис. 3.9). Команда действует только в редакторе.

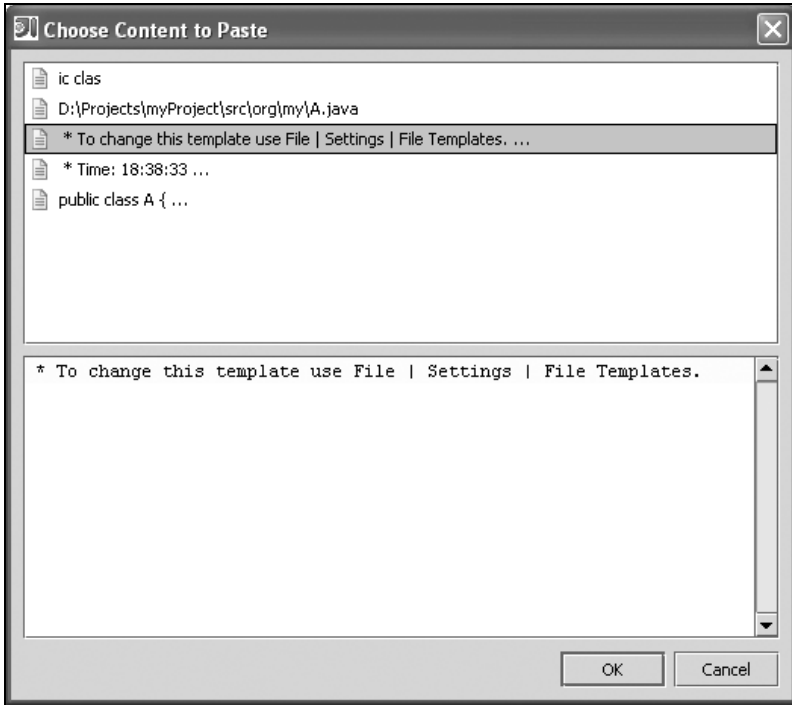


Рис. 3.9. Выбор содержимого буфера обмена для вставки

3.2.2.8. Column Mode (Режим колонки)

"Горячие" клавиши: <Ctrl>+<Shift>+<Insert>.

Команда переключения режима выделения со строчного (рис. 3.10) на колоночный с возможностью выделить вертикальный блок текста (рис. 3.11).



Рис. 3.10. Строчный режим выделения