

Эллеонора Керри

41 уровень

12.12.2017 76721 12

Топ-10 библиотек Java, которые помогут сэкономить время

Статья из группы Java Developer

42759 участников

Вы в группе

Для кого эта статья: кое-что будет интересно новичкам, которые изучают квест Java Syntax, но в основном изложенный материал лучше зайдёт тем, кто уже перевалил за экватор курса (разбирается с квестами Java Collections и Java Multithreading) и начинает интересоваться технологиями, выходящими за рамки Java Core.



За время своего существования Java изрядно обросла библиотеками. Но как понять, какие из них стоит использовать, а какие — нет? Отделить зёрна от плевел вечно занятому разработчику крайне непросто. В этой [статье](#) Мартин Пацов (Martin Patsov), Full Stack Developer компании Dreamix, расскажет о своей любви к библиотекам с открытым исходным кодом (open source), и предложит вам те из них, которые считает особенно полезными.

*Быть иль не быть, вот в чём вопрос:
Спасеньем обернётся иль провалом
Та самая библиотека Java...
Смелей! «СинийОры» уже были там
И опыт их тебе я передам.*

Пусть вас не вводит в заблуждение моё поэтическое отступление. Библиотек в Java действительно очень много, и разобраться с ними ох как непросто. Тем не менее, я их очень люблю. Меня даже можно назвать «библиотечным евангелистом» (*под «евангелистом» в сфере ИТ понимают вовсе не составителя канонический евангелий, а того, кто профессионально пропагандирует ту или иную технологию. Не путайте с обычными пиарщиками: чаще всего ИТ-евангелист досконально разбирается в продвигаемой им технологии. — прим. ред.*). Библиотеки — очень важная часть экосистемы open source. Вокруг них вырастают большие сообщества, которые их развивают и поддерживают. И это здорово. Главное достоинство

НАЧАТЬ ОБУЧЕНИЕ

уже написанный для определённых задач код — как можно чаще! Вы сэкономите время. Seriously, лучше потратить его на общение с семьей и друзьями.

Топ-10

В этой заметке я упомянул те Java-библиотеки, которые предпочитаю использовать в своей работе. Все они относятся к ПО с открытым исходным кодом. Выбирал я по принципу полезности, наличия хорошей документации и поддержки, а также регулярной обновляемости. Хочу ещё раз поставить акцент на том, что это — мой личный выбор, и он может существенно отличаться от вашего.

1. Стандартные библиотеки Java

Да-да, вам не показалась! Очень многие люди недооценивают, более того — не знают массы возможностей — стандартных библиотек Java (Java Standard Libraries), и не в курсе, как раскрыть их потенциал в программировании... а то и вовсе не используют их. Вот краткое описание некоторых из них:

- ***java.lang***. Эта библиотека всегда импортируется в любое Java-приложение, поскольку содержит всё то, без чего вы в принципе не сможете программировать (***String***, ***Double***, ***Enum***, ***Math*** и так далее).
- В ***java.util*** вы можете найти все коллекции и структуры данных, доступные в Java.
- Далее, у нас есть ***java.io*** для чтения файлов, работающих потоками, объектами класса Pipe и так далее.
- О первых трёх вы, скорее всего, уже слышали. А вот с ***java.nio***, альтернативой ***java.io***, знают уже не все. Эта библиотека отвечает за неблокирующий ввод-вывод, позволяя, как вы могли догадаться, интенсивно использовать операции ввода-вывода.
- ***java.net*** используется для работы с сокетами, создания подключений или коротких сетевых приложений.
- В Java у нас также есть библиотеки для работы с GUI: ***javax.swing*** (довольно устаревшее расширение ещё более старого ***java.awt***).
- Мы даже можем воспроизводить музыку и создавать MIDI-файлы с помощью ***java.sound***.

2. JHipster

JHipster — это платформа для быстрого развертывания, разработки и создания масштабируемых веб-серверов с высокой нагрузкой и использованием самых современных и модных технологий таких как Spring, Spring-MicroServices, Netflix,Docker, Kubernetes, AngularJs, Liquibase, MongoDB, Cassandra, Elasticsearch.



Честное слово, этот «Java-хипстер» определённо спас мне месяцы жизни. Я могу точно сказать, что именно JHipster — мой личный фаворит в списке и он занимает особое место в моем сердце. Этот инструмент — практически незаменим для генерирования эскиза проекта распределенного веб-сервера. Он умеет генерировать рот-файл с зависимостями, настраивать Elastic Search и Connection, вам остается только добавить бизнес-логику архитектуры. Основными и наиболее важными библиотеками. включенными в сгенерированный проект. являются:

НАЧАТЬ ОБУЧЕНИЕ

- [Angular](#)/ AngularJS - инфраструктура JavaScript

Здесь вы можете найти больше информации о [JHipster](#).

3. Maven

Maven — один из трёх самых популярных инструментов для сборки проектов на Java. Он отвечает за компиляцию, создание jar-файлов (Java-архивов), создание дистрибутива программы, генерацию документации.



Каждый современный проект состоит из разных файлов, модулей, технологий. Говоря очень просто, сборка проекта — подготовка его «рабочей» версии, то есть той программы, которую можно запустить на выполнение. Это можно делать «ручками», в командной строке, а можно написать специальные скрипты. Однако гораздо легче это проделывать с помощью специальных приложений. Называются они «системы автоматической сборки». Maven — одна из трёх таких программ, две другие — Gradle и Ant. На вход такие системы получают исходники проекта, а на выход выдают программу, которую можно запустить.

Структура проекта (зависимости, конфигурации, документация) описывается в файле pom.xml (POM – Project Object Model), который должен находиться в корневой папке проекта.

Честно, Maven — просто отличная штука, хоть поначалу и не очень понятная. Если вы никогда не использовали Maven раньше, вы многое потеряли. Откровенно говоря, я вообще не понимаю, как создавались Enterprise-приложения до появления Maven.

Почитать о Maven на русском можно [здесь](#).

4. Apache Commons

Apache Commons — фактически целый проект, ориентированный на создание библиотек Java. Или «большой набор маленьких Java-утилит» самого разного предназначения. К слову, утилиты Apache Commons лежат в основе нескольких очень известных проектов, например, Tomcat, Hibernate и других.

Вот краткий список некоторых из наиболее часто используемых библиотек Apache Commons:



НАЧАТЬ ОБУЧЕНИЕ

- [Commons CLI](#). Предоставляет API для анализа аргументов командной строки. Вряд ли стоит даже в мыслях допускать создание приложения без возможности передавать параметры и контролировать его поведение!
- [Commons CSV](#). Независимо от того, что вы разрабатываете, в какой-то момент вам придется столкнуться с необходимостью использования csv-файлов. То есть, их придётся открывать, читать, редактировать, сохранять и создавать. Я предлагаю использовать формат RFC 4180 из класса CSVFormat и кодировку UTF-8 при сохранении / создании файлов.
- [Commons IO](#). Используется для упрощения выполнения операций ввода-вывода. Присмотритесь, по крайней мере, к ReversedLinesFileReader, он того стоит!

Почитать об Apache Commons (на английском) можно [здесь](#).

5. Guava

Guava — набор основных библиотек Google для Java. Они содержат новые типы коллекций (например, multimap и multiset), неизменяемые коллекции, библиотеку графов, функциональные типы? утилиты для параллелизма, ввода/вывода, хэширования, обработки строк и многое другое. Все эти инструменты используют специалисты Google в своей работе, а это уже хорошая рекомендация.



В какой-то мере Guava является альтернативой Apache Commons.

На самом деле, о Guava можно говорить долго (обязательно обратите на неё внимание!), но здесь я упомяну только одну полезную штуку из этой библиотеки. Возможно, вы уже сталкивались с задачами упорядочивания своих коллекций или сравнения содержимого в них? А что насчёт расширенной сортировки и сравнения по нескольким столбцам с несколькими условиями? Компонент **ComparisonChain** используется для реализации расширенной и сложной сортировки для коллекций.

Почитать о Guava (на английском) можно [здесь](#).

Твой Java-дайджест

Полезные статьи о Java: обучение, программирование, карьера

Введите свой e-mail

6. google-gson

Библиотека **google-gson** полезна для преобразования объектов Java в JSON и наоборот. Она особенно пригодится при разработке мобильных приложений и создании/использовании REST API, а также в любом случае, когда вам нужно преобразовать объект Java в его представление JSON и наоборот.

НАЧАТЬ ОБУЧЕНИЕ



О JSON (JavaScript Object Notation), Guava, Apache Commons и прочих интересных вещах вы узнаете из квеста [Java Collections](#).

Почитать о google-gson (на английском) можно [здесь](#).

7. Hibernate-ORM

Hibernate — библиотека, предназначенная для решения задач объектно-реляционного отображения (ORM). Hibernate ORM используется для хранения данных в реляционных базах данных. Она обеспечивает более простой и абстрактный способ сделать это для разработчиков. Он использует JDBC в своей реализации. Hibernate также является реализацией спецификации JPA.



Почитать о Hibernate ORM (на английском) можно [здесь](#).

8. Mockito

Пусть название Mockito не вводит вас в заблуждение. Речь не о коктейле, а о библиотеке для мок-объектов. Мок-объекты — это объекты, которые имитируют поведение реального объекта по какой-то заданной схеме. Например, для модульного тестирования такие «поддельные» объекты могут симулировать поведение бизнес-объектов. Ну а мок-библиотека Mockito повышает удобство создания и использования мок-объектов.



НАЧАТЬ ОБУЧЕНИЕ

9. JUnit

[JUnit](#) — свободный Java-фреймворк для создания модульных тестов (они же — юнит-тесты). Эта библиотека — одна из самых популярных, её используют почти все. Рекомендуем вам разобраться с ней как можно быстрее, поскольку очень часто начинающих Java-разработчиков и Java Trainee заставляют «покрывать код тестами». И, надо сказать, это далеко не самое плохое начало карьеры.



Больше информации о Junit [здесь](#).

10.Log4j и Slf4j

Эти два фреймворка созданы для скрытия реализации рутинных операций по журналированию определённых событий, которые происходят во время работы Java-приложений. Slf4j представляет собой абстракцию для других фреймворков журналирования (того же Log4j).



Больше информации о [Log4j](#) и [Slf4j](#).

Что еще почитать:

[Топ-3 Blockchain-библиотек для Java-разработчиков](#)

+133

Комментарии (12)

популярные

новые

старые

JavaCoder

Введите текст комментария

НАЧАТЬ ОБУЧЕНИЕ

Андрей

Уровень 27, Москва, Россия

EXPERT

6 сентября 2020, 00:06

В общем даже на 40-ом уровне я, вернувшись к этой статье, буду понимать, что толком даже не начинал учиться...

Ответить

+7

Максим

Уровень 40, Беларусь

EXPERT

14 декабря 2017, 00:51

Ипать... Сколько ж мне еще всего надо выучить)

Ответить

+26

Вячеслав Чернышов

Backend Developer в Дом.рф

13 декабря 2017, 10:29

Указать в качестве вспомогательной библиотеки JDK и забыть Spring - мощно.

Ответить

+3

LuneFox

инженер по сопровождению в BIFIT

EXPERT

15 января, 08:49

Spring упоминается в JHipster :)

Ответить

+1

hidden #2595317

Уровень 45

21 февраля, 18:25

В этом месте он уже отключился.)

Ответить

+2

Игорь

Уровень 0

12 декабря 2017, 17:31

Какими продуктами(библиотеками) более удобно пользоваться для работы с базами данных? Опыт очень большой. Только все с десктопными приложениями. Хочу попробовать проект с помощью IDE NetBeans фирмы ORACLE

Ответить

+2

MJM

Уровень 15, Нижний Новгород

12 декабря 2017, 18:10

Эээ... не совсем понятно, что вы имеете в виду. Если вы хотите создать Java-проект и подключить в нём базы данных, посмотрите на драйверы JDBC (они в стандартную джаву входят, java.sql).

Ответить

+6

Игорь

Уровень 0

13 декабря 2017, 05:48

Действительно, хочу, для начала, понять создание приложения с подключением к СУБД и вводом/выводом данных. Понять механизм. Дальше, думаю, разберусь.

Ответить

0

Вячеслав Чернышов

Backend Developer в Дом.рф

13 декабря 2017, 10:31

Очень большой опыт работы с десктопными приложениями и полное отсутствие опыта работы с БД? Впрочем, дальнейшие комментарии всё объясняют.

Ответить

+9

Игорь

Уровень 0

14 декабря 2017, 05:11

Так, расслабься.
Кто тебе сказал, что я недоволен? Опять пытаешься гадать, комментировать.
У меня два сына. Один пошел в первый класс, второй окончил университет. Я думаю, даже мой первоклассник объяснил тебе, как себя вести, тем более с человеком, которого ты пытаешься оскорбить, заведомо зная, что тебе ничего не будет.
В твоих комментариях не нуждаюсь.
Свободен. Надеюсь, намек понятен?

Ответить

+5

Игорь

Уровень 0

14 декабря 2017, 05:40

Умение читать комментарии, возможно тебя самого впечатляет. Вот только выводы при себе оставляй. Я задал вопрос. А тут начали разводить холивар.
Насчет работы с БД.
Я, вроде, не писал про "полное отсутствие опыта работы с БД".
У меня, действительно, большой опыт работы с ними. Причем, не только с реляционными.
Насчет приложений. Могу сказать, их тоже достаточно.
Такое ощущение, что достигнутый здесь уровень некоторые путают с уровнем IQ. Не надо заблуждаться. Похоже, это обратно пропорционально. По крайней мере, судя по общению в данной теме. Только **MJM** попытался уточнить вопрос и подсказать(мне, кстати, этого было достаточно). Остальное вода и попытки как-то зацеить. Зачем?
Может быть, это тенденция данного ресурса? Тогда я тут не задержусь. Есть много, более практичных и доброжелательных.

Ответить

+5

ОБУЧЕНИЕ

- Курсы программирования
- Курс Java
- Помощь по задачам
- Подписки
- Задачи-игры

СООБЩЕСТВО

- Пользователи
- Статьи
- Форум
- Чат
- Истории успеха
- Активности

КОМПАНИЯ

- О нас
- Контакты
- Отзывы
- FAQ
- Поддержка



JavaRush — это интерактивный онлайн-курс по изучению Java-программирования с нуля. Он содержит 1200 практических задач с проверкой решения в один клик, необходимый минимум теории по основам Java и мотивирующие фишки, которые помогут пройти курс до конца: игры, опросы, интересные проекты и статьи об эффективном обучении и карьере Java-девелопера.

ПОДПИСЫВАЙТЕСЬ

ЯЗЫК ИНТЕРФЕЙСА

 Русский 

