

Обзор Java: введение, преимущества и история создания языка программирования

Прежде, чем перейти к обучению, начнём с введения в java-программирование, разберёмся кратко, что это за такой язык программирования. Немного истории языка java. Язык Java разработан компанией Sun Microsystems, создателем которого был Джеймс Гослинг, и выпущен в 1995 году в качестве основных компонентов компании Sun Microsystems — Java платформ (Java 1.0 [J2SE]).

По состоянию на 2017 год последней версией Java Standard Edition является 8 (J2SE). С развитием Java, и её широкой популярностью, несколько конфигураций были построены для различных типов платформ. Например: J2EE — приложения для предприятий, J2ME — для мобильных приложений.

Sun Microsystems переименовала прежнюю версию J2 и ввела новые: Java SE, Java EE и Java ME. Введение в программирование Java различных версий подтверждало знаменитый слоган компании «**Напиши один раз, запускай везде**».

История создания языка Java

История создания языка Java начинается в июне 1991 года, когда Джеймс Гослинг создал проект для использования в одном из своих многочисленных сет-топ проектов. Язык, который рос вне офиса Гослинга, как дуб, **Oak** - первоначальное название Java до 1995 года, после в дальнейшем история Java продолжалась под именем **Green**, а позже был переименован как Java.

Но официальной датой создания языка Java считается 23 мая 1995 года, после выпуска компанией Sun первой реализации Java 1.0. Она гарантировала «**Напиши один раз, запускай везде**», обеспечивая недорогой стоимостью на популярных платформах.

13 ноября 2006 года, Sun выпустила большую часть как свободное и открытое программное обеспечение в соответствии с условиями GNU General Public License (GPL).

После 8 мая 2007 года судьба Java сложилась иначе. Компания завершила процесс, делая все чтобы исходный код был бесплатным и открытым, кроме небольшой части кода, на который компания не имела авторских прав.

Преимущества языка Java: краткий обзор

Объектно-ориентированный: в джава все является объектом. Дополнение может быть легко расширено, так как он основан на объектной модели.

Платформонезависимый: в отличие от многих других языков, включая C и C++, Java, когда был создан, он не компилировался в платформе конкретной машины, а в независимом от платформы байт-коде. Этот байт код распространяется через интернет и интерпретируется в Java Virtual Machine (JVM), на которой он в настоящее время работает.

Простой: процессы изучения и введение в язык программирования Java остаются простыми. Если Вы понимаете основные концепции объектно-ориентированного программирования, то он будет прост для Вас в освоении.

Безопасным: методы проверки подлинности основаны на шифровании с открытым ключом.

Архитектурно-нейтральный: компилятор генерирует архитектурно-нейтральные объекты формата файла, что делает скомпилированный код исполняемым на многих процессорах, с наличием системе Java Runtime.

Портативный: архитектурно-нейтральный и не имеющий зависимости от реализации аспектов спецификаций — все это делает Java портативным. Компилятор в Java написан на ANSI C с чистой переносимостью, который является подмножеством POSIX.

Прочный: прилагает усилия, чтобы устранить ошибки в различных ситуациях, делая упор в основном на время компиляции, проверку ошибок и проверку во время выполнения.

Многопоточный: функции многопоточности, можно писать программы, которые могут выполнять множество задач одновременно. Введение в язык Java этой конструктивной особенности позволяет разработчикам создавать отложенные интерактивные приложения.

Интерпретированный: Java байт-код переводится на лету в машинные инструкции и нигде не сохраняется. Делая процесс более быстрым и аналитическим, поскольку связывание происходит как дополнительное с небольшим весом процесса.

Высокопроизводительный: введение Just-In-Time компилятора, позволило получить высокую производительность.

Распространенный: предназначен для распределенной среды интернета.

Динамический: программирование на Java считается более динамичным, чем на C или C++, так как он предназначен для адаптации к меняющимся условиям. Программы могут выполнять обширное количество во время обработки информации, которая может быть использована для проверки и разрешения доступа к объектам на время выполнения.

После небольшого введения, обзора преимуществ и истории Java, приступим к нашему обучению.

Инструменты, которые понадобятся

Для выполнения примеров, рассмотренных в этом учебнике, Вам нужен компьютер Pentium 200 МГц с минимальной оперативной памятью 64 Мб (рекомендуется оперативная память 128 Мб).

Вам также понадобится следующее программное обеспечение:

- Linux 7.1, Windows 95/98/2000/7/8 и выше или другая операционная система.
- JDK 5 и выше.
- Notepad или любой другой текстовый редактор.

Самоучитель обеспечит необходимыми навыками для создания GUI, сетевых и веб-приложений.

Ознакомившись с особенностями истории появления и преимуществами языка Java, Вы закончили урок введения в язык программирования.

Следующий урок будет Вас вести к изучению языка и документации. Проинструктирует Вас о том, как установить и подготовить среду для разработки приложений.