

УДК 004.432.45

*Матвеев Д. А., студент 1 курса магистратуры, гр. ИСТМз-21
ФГОБУ ВО «Поволжский государственный университет
телекоммуникаций и информатики», Россия, г. Самара*

**АНАЛИЗ ВОЗМОЖНОСТИ ИСПОЛЬЗОВАНИЯ ОБЪЕКТНО-
ОРИЕНТИРОВАННОГО ЯЗЫКА ПРОГРАММИРОВАНИЯ JAVA
ДЛЯ РЕШЕНИЯ ПРОИЗВОДСТВЕННЫХ ЗАДАЧ.**

Аннотация: В статье рассматриваются основные возможности языка Java. Акцентируется внимание на некоторых особенностях применения языка. Показываются некоторые технологии, сформированные на основе языка Java.

Ключевые слова: языки программирования, объектно-ориентированные языки, джава, разработка программного обеспечения.

*Matveev D. A., 1st year Master's student,
of gr. ISTMZ-21 FGOBU in «Volga State University of
Telecommunications and Informatics», Russia, Samar*

**ANALYSIS OF THE POSSIBILITY OF USING THE OBJECT-
ORIENTED PROGRAMMING LANGUAGE JAVA FOR SOLVING
SOME PRODUCTION PROBLEMS.**

Abstract: The article discusses the main features of the Java language. Attention is focused on some features of the use of the language.

Keywords: programming languages, object-oriented languages, java, software development.

Среди объектно-ориентированных языков программирования особое место занимает язык под названием Java. Ключевой особенностью языка является кроссплатформенность, обусловленная двухступенчатой обработкой кода. На первом этапе текст программы, написанный программистом, переводится компилятором в специальный байт-код. Затем, виртуальная машина JVM через соответствующий интерпретатор кода выполняет байт-код. Таким образом, достигается возможность

написания кода программ единожды, без учета специфики всех возможных платформ, и запускать итоговую программу на различных платформах, где присутствует JVM. Эта особенность Java получила название «скомпилируй единожды, запускай везде» (compile once, run anywhere).

Поскольку Java является объектно-ориентированным языком программирования, то при написании кода используются различные абстракции. Следовательно, качество и тип абстракции напрямую определяют сложность решаемых задач[1]. Объектно-ориентированный подход в Java не накладывает ограничений на тип решаемой проблемы, предоставляя программисту большой простор для выбора элементов, характеризующих решаемую задачу. С точки зрения концепции ООП все является объектом. Стоит отметить, что каждый объект не вечен и обладает некоторым временем жизни. В отличие от других языков программирования, Java берет на себя отслеживание времени жизни объекта, удаляя его, в случае отсутствия необходимости его использования.

Современный прогресс привел к появлению многоядерных процессоров, позволяющих производить обработку данных в параллельном режиме. В результате уменьшается время обработки данных. Одной из характерных задач, требующих параллельного режима является отображение пользовательского интерфейса или обработка веб-сервером многочисленных запросов. Однако многопоточная обработка является сложным делом. Java предоставляет большое количество встроенных инструментальных средств для решения задач параллельной обработки данных. Основной единицей выступает поток, или Thread, позволяющий обрабатывать данные, не зависимо от других потоков. Для обеспечения корректного использования общего доступа к данным в Java существует специальный механизм синхронизации, который предоставляет удобные средства, такие как, различные планировщики, потокобезопасные

структуры данных, элементы управления потоками на основе семафоров, барьеров и мониторов[2].

Java широко применяется для решения задач в области интернета, интрасетей и серверных решений. Часто можно встретить необходимость обеспечения противоречивых требований существующих корпоративных информационных систем. При этом необходимо корректно координировать корпоративные данные, которые хранятся в разных местах и обрабатываются различными системами. Для обеспечения этих требований, появились API-интерфейсы и различные реализации. Наиболее известной реализацией является платформа Java EE. Эта платформа предоставляет стандартные способы работы с транзакциями, с общением и сохраняемостью[3]. С развитием технологий, на смену этой платформе пришел каркас Spring. Развиваясь, Spring стал полноценной технологией, состоящей из целого ряда отдельных проектов [4]. Эти проекты позволяют решать различные задачи, такие как обеспечение авторизации и аутентификации, доступ к базам данных, обмен сообщениями через специальные брокеры сообщений, реактивное взаимодействие и многое другое.

Подводя итог, можно отметить, что на сегодняшний день объектно-ориентированный язык программирования Java обладает широкими возможностями для решения различных производственных задач. Сформировано большое сообщество вокруг этого языка, которое позволяет, как вносить изменения и улучшения в сам язык, так и в сформированные на его основе платформы.

Использованные источники:

1. Эккель, Б., Философия Java. 4-е полное издание. [Текст] / Б. Эккель- СПб.: Питер, 2017. – 1168 с.
2. Хорстманн, К., Java. Библиотека профессионала, том 1. Основы. 10 изд.: Пер. с англ [Текст] / К. Хорстманн- М.: ООО «И.Д. Вильямс», 2017. – 864 с.

3. Гонсалвес, Э., Изучаем JavaEE7. [Текст] / Э. Гонсалвес - СПб.: Питер, 2017. – 640 с.

4. Козмина, Ю., Харроп, Р., Шеффер, К., Хо, К., Spring 5 для профессионалов.: пер с англ. [Текст] / Ю. Козмина, Р. Харроп, К. Шеффер, К. Хо – СПб.: ООО «Диалектика», 2019. – 1120 с.