

Торайғыров университетінің
ҒЫЛЫМИ ЖУРНАЛ

НАУЧНЫЙ ЖУРНАЛ
Торайғыров университета

ТОРАЙҒЫРОВ УНИВЕРСИТЕТІНІҢ ХАБАРШЫСЫ

Физика, математика және компьютерлік
ғылымдар сериясы
1997 жылдан бастап шығады



ВЕСТНИК ТОРАЙҒЫРОВ УНИВЕРСИТЕТА

Серия: Физика, математика
и компьютерные науки
Издается с 1997 года

ISSN 2959-068X

№ 2 (2022)
Павлодар

**НАУЧНЫЙ ЖУРНАЛ
ТОРАЙГЫРОВ УНИВЕРСИТЕТА**

Серия: Физика, математика и компьютерные науки
выходит 4 раза в год

СВИДЕТЕЛЬСТВО

о постановке на переучет периодического печатного издания,
информационного агентства и сетевого издания
№ KZ91VPY00046988
выдано

Министерством информации и общественного развития
Республики Казахстан

Тематическая направленность
публикация материалов в области физики, математики,
механики и информатики

Подписной индекс – 76208

[https://doi.org/ 10.48081/JVDF6096](https://doi.org/10.48081/JVDF6096)

Бас редакторы – главный редактор

Тлеукунов С. К., *ф.-м.г.д., профессор*

Заместитель главного редактора Испулов Н. А., *ф.-м.г.к., профессор*

Ответственный секретарь Жумабеков А. Ж., *магистр*

Редакция алқасы – Редакционная коллегия

Esref Adali,

Qadir Abdul,

Домбаев К. М.,

Демкин В. П.,

Жумадиллаева А. К.,

Ибраев Н. Х.,

Косов В. Н.,

Сеитова С. М.,

Шоканов А. К.,

Омарова А. Р.,

PhD докторы, профессор (Турция);

PhD докторы, профессор (Пакистан);

ф.-м.г.д., профессор;

ф.-м.г.д., профессор (Российская Федерация);

т.г.к., кауымд. профессор;

ф.-м.г.д., профессор;

ф.-м.г.д., профессор;

пед.г.д., профессор;

ф.-м.г.к., профессор

технический редактор

За достоверность материалов и рекламы ответственность несут авторы и рекламодатели

Редакция оставляет за собой право на отклонение материалов

При использовании материалов журнала ссылка на

«Вестник Торайгыров университета» обязательна

***С. Н. Талипов**

Торайгыров университет, Республика Казахстан, г. Павлодар

ПРЕПОДАВАНИЕ ПРОМЫШЛЕННОГО ПРОГРАММИРОВАНИЯ НА JAVA В ВУЗЕ

Статья предназначена в первую очередь студентам-программистам, а также всем, кто интересуется современным промышленным программированием на Java.

В настоящее время Java является основным языком разработки программ для промышленности и крупного бизнеса. Это связано с кроссплатформенностью языка, его бесплатностью, а также заложенным в него объектно-ориентированным принципам разработки программного обеспечения. Поэтому очень важно знать все его основные технологии и возможности. Данные технологии являются надстройкой над основой языка Java и объединяются под термином «промышленное программирование».

В данной статье рассматриваются все основные технологии промышленного программирования на Java, которые актуальны на данный момент и которые были лично использованы и опробованы автором при разработке коммерческих программ для нефтеперерабатывающей отрасли Республики Казахстан за последние 5 лет.

В представленной статье рассмотрены технологии сборки программ на Java из исходного кода, базы данных и технологии работы с ними, технологии создания графического интерфейса взаимодействия с пользователем, сетевые технологии, расширенные технологии ООП, библиотеки для импорта/экспорта данных в офисные форматы, а также общесистемные технологии.

Благодаря информации, представленной в статье, инженер-программист будет знать, с помощью чего можно сделать реальный промышленный проект на Java.

Все рассмотренные в данной статье технологии являются базовыми и основными для разработки промышленных программ на Java. Существуют и другие технологии, но те, что рассмотрены,

в обязательном порядке должны быть известны каждому современному программисту на Java.

Ключевые слова: Java, промышленное программирование, базы данных, объектно-ориентированное программирование, программа, технологии.

Введение

В настоящее время Java является основным языком разработки программ для промышленности и крупного бизнеса. Это связано с кроссплатформенностью языка, его бесплатностью, а также заложенным в него объектно-ориентированным принципам разработки программного обеспечения. Поэтому очень важно знать все его основные технологии и возможности. Данные технологии являются надстройкой над основой языка Java и объединяются под термином «промышленное программирование».

Материалы и методы

В данной статье рассматриваются все основные технологии промышленного программирования на Java, которые актуальны на данный момент и которые были лично использованы и опробованы автором при разработке коммерческих программ для нефтеперерабатывающей отрасли Республики Казахстан за последние 5 лет.

Результаты и обсуждение

Для начала рассмотрим технологии сборки программ из исходного кода. В Java используются три основных способа сборки программ из исходных кодов (исходников): Ant, Maven и Gradle. Технология сборки определяет, где искать нужные модули к программе и как именно их объединить в единое целое. Технологии сборки реализуются через сборщики программ, которые компилируют Java-код, делают копирование/удаление/перемещение/создание файлов и каталогов, автоматический запуск тестов и др.

Самый древний сборщик программ – это Ant («муравей») [1]. Данный способ собирает программу из исходников и готовых вспомогательных библиотек формата «jar», которые должны быть доступны локально в конкретной подпапке. Способ хорош тем, что для сборки не нужен Интернет, а минусом можно отметить устаревание со временем библиотек и отсутствие их авто обновления.

Более продвинутый способ, чем Ant – это Maven («собиратель знания») [2]. При данном способе нужные библиотеки автоматически скачиваются из Интернета в соответствии с данными из файла сборки «pom.xml». Пример данного файла:

```
<dependencies>  
  <dependency>
```

```
<groupId>org.hibernate</groupId>
<artifactId>hibernate-core</artifactId>
<version>5.4.21.Final</version>
</dependency>
<dependency>
  <groupId>mysql</groupId>
  <artifactId>mysql-connector-java</artifactId>
  <version>8.0.23</version>
</dependency>
</dependencies>
```

После выполнения всех указанных требований в файле «pom.xml» программа будет откомпилирована и сгенерирована нужная документация.

Самый совершенный и современный способ сборки программ на Java, а также и Android – это Gradle [3]. При данном способе нужные библиотеки автоматически скачиваются из Интернета в соответствии с данными из файла сборки «build.gradle». Пример данного файла:

```
repositories {
  mavenCentral()
}
dependencies {
  implementation group: <org.hibernate>, name: <hibernate-core>, version:
<5.4.21.Final>
  implementation group: <mysql>, name: <mysql-connector-java>, version:
<8.0.23>
}
```

Особенность сборщика Gradle в том, что он поддерживает каскадную модель разработки, определяя, какие компоненты сборки не изменились и какие задачи, зависящие от этих частей, не требуют перезапуска.

Базы данных и технологии работы с ними

Базы данных (БД) в Java – это одна из самых важных тем, после объектно-ориентированного подхода (ООП) к созданию программ. Все промышленные программы на Java используют базы данных в своей работе. Наиболее востребованные системы управления базами данных (СУБД) – это локальная Derby (или SQLite), MySQL (или платный MS SQL), PostgreSQL и др.

Derby – это простая локальная СУБД, которая сама написана на Java, имеет небольшое занимаемое место на диске и позволяет взаимодействовать Java-приложениям через промышленный стандарт взаимодействия JDBC (Java DataBase Connectivity) [4]. База данных формата Derby представляет собой папку с файлами определенной структуры. Derby используется в

переносимых программах с локальной БД, к которой имеется монопольный не сетевой доступ.

Как альтернатива данной СУБД может рассматриваться SQLite, как более универсальная и используемая в Android. Пример фрагмента программы с использованием Derby [4]:

```
Connection connection = DriverManager.getConnection(«jdbc:derby://localhost:1527/disks», authorization);
java.sql.Statement statement = connection.createStatement(ResultSet.
TYPE_SCROLL_SENSITIVE,
statement.execute(«select ID, NAME_DISK, PRICE_PUR, PRICE_SEL
from disk_1»);
ResultSet table = statement.getResultSet();
table.first();
for (int j = 1; j <= table.getMetaData().getColumnCount(); j++) {
System.out.print(table.getMetaData().getColumnName(j) + «\t»); }
```

СУБД MySQL [5] является основной для многих сайтов и промышленных программ. Данная СУБД имеет развитые типы данных, представления, триггеры, хранимые процедуры, функции, индексы и инструментальные средства манипулирования данными. MySQL работает по клиент-серверному механизму и не является локальной СУБД. Данная СУБД используется для сложных программ с многопользовательскими данными, с использованием транзакций и работы по сети.

СУБД PostgreSQL [6] подобна по характеристикам и возможностям MySQL и является основной для промышленных программ с требованиями полной лицензионной чистоты для любого коммерческого и не коммерческого использования.

Для работы с СУБД в Java, помимо простейших возможностей JDBC, используются еще две основные технологии: Hibernate [7] и Spring [8].

Hibernate – это промышленная технология работы с базами данных, которая позволяет работать с записями БД как с обычными объектами Java через т.н. объектно-реляционные отображения. В данной технологии при манипуляции с объектами автоматически происходят соответствующие нужные изменения в таблицах БД, от программиста требуется лишь настроить связь между классами его программы и таблицами БД через конфигурационные файлы. Пример фрагмента программы с использованием Hibernate:

```
Session session = HibernateUtil.getSessionFactory().openSession();
UserRole userRole = new UserRole();
userRole.setName(“R114”);
userRole.setAccessCodes(“1,2,5”);
session.save(userRole);
```

```
userRole = (UserRole) session.get(UserRole.class, userRole.getId());
userRole.setAccessCodes("2");
session.save(userRole);
List<UserRole> list = (List<UserRole>) session.createCriteria(UserRole.
class).list();
for (UserRole role : list) {
    System.out.println(role);
}
session.close();
```

Промышленная технология работы с базами данных Spring позволяет удобно работать с данными БД через готовые методы данного фреймворка. Это самая распространенная и удобная промышленная технология для работы с БД в Java. Технология Spring позволяет создавать объекты Java не только обычным способом через код программы, но и из xml-файлов так называемых «бинов». Пример бина:

```
<bean id="dataSource" class="org.springframework.jdbc.datasource.
DriverManagerDataSource">
    <property name="driverClassName" value="com.mysql.cj.jdbc.Driver"/>
    <property name="url" value="jdbc:mysql://localhost:3306/tsn_demo"/>
    <property name="username" value="root" />
    <property name="password" value="root" />
</bean>
<bean id="customerDAO" class="kz.tsn.spring.PersonDAO">
    <property name="dataSource" ref="dataSource" />
</bean>
```

Фрагмент основного кода программы с использованием Spring выглядит следующим образом:

```
@Override
public void append(String firstName, String lastName, int age) {
    JdbcTemplate insert = new JdbcTemplate(dataSource);
    insert.update("INSERT INTO person (FIRSTNAME, LASTNAME, AGE)
VALUES(?,?,?)", new Object[]{firstName, lastName, age}); }
@Override
public void deleteByLastName(String lastname) {
    JdbcTemplate insert = new JdbcTemplate(dataSource);
    insert.update("DELETE FROM person WHERE LASTNAME LIKE ?",
new Object[]{"%" + lastname + "%"}); }
```

От программиста при использовании фреймворка Spring требуется знать его многочисленные классы и методы, а также учитывать особенности конкретной версии как Java, так и самого Spring.

Технологии создания графического интерфейса взаимодействия с пользователем

Многие программы на Java нуждаются в графическом интерфейсе для взаимодействия с пользователем. Раньше для этих целей использовались библиотеки классов AWT, потом классы Swing, а сейчас наиболее популярен фреймворк JavaFX [9].

Данный фреймворк имеет наибольшие возможности в создании интерфейса программы, но имеется и недостаток: JavaFX еще не входит в JDK и его нужно отдельно устанавливать как на компьютере разработчика программы, так и на компьютере у пользователя. Также для проектирования окон программы в JavaFX приходится использовать стороннее приложение SceneBuilder, что тоже не очень удобно. Тем не менее, данный фреймворк стал стандартом в современных программах на Java. Пример внешнего вида программы на Java с использованием JavaFX показан на рисунке 1:

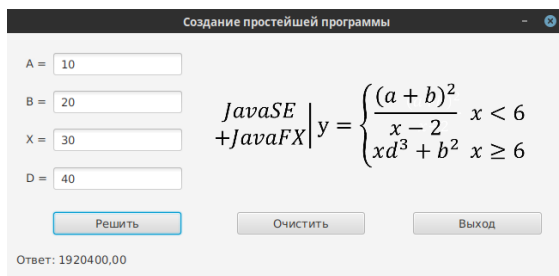


Рисунок 1 – Интерфейс с использованием JavaFX

Сетевые технологии

Современные промышленные программы используют следующие сетевые технологии [4, 10-11]:

- GSON, JSON – использование текстового формата данных для передачи информации по сети, пример формата JSON: “{“time”: {“updated”: “Jun 11, 2022 18:02:00 UTC”, “updatedISO”: “2022-06-11T18:02:00+00:00”, “updateduk”: “Jun 11, 2022 at 19:02 BST”}, ...}”;
- JSOUP – парсинг данных из содержимого html-страниц (классы org.jsoup.*);
- SOCKETS – создание клиент-серверных программ на основе сокетов TCP/IP (классы java.net.*);
- SSL – использование зашифрованной передачи данных по сети с использованием цифровых сертификатов;

- JSP, SERVLET – создание сервлетов, т.е. Java-программ, работающих с пользователем через его браузер и активные JSP-страницы. Пример работы сервлета с активными JSP-страницами показан на рисунке 2;
- RMI – создание клиент-серверных Java-программ для связи между собой (классы java.rmi.*);
- SOAP – создание клиент-серверных программ для связи различных программ между собой по сети через пересылку скрытых xml-файлов (классы javax.jws.);
- Telegram bot API – создание ботов для Telegram-мессенджера (классы org.telegram.telegrambots.*);

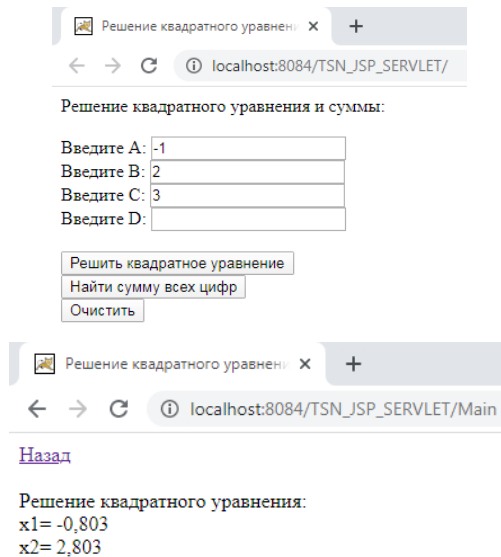


Рисунок 2 – Интерфейс с использованием JavaFX

Расширенные технологии ООП

Промышленные программы используют следующие современные ООП-технологии в своем коде [4, 12]:

- аннотации, пример аннотации: "@NonNull String sayMessage(@NonNull String message);";
- лямбда-выражения (функциональное программирование), например "INumericTest isEven = (n) -> (n % 2) == 0;";
- паттерны проектирования (шаблоны для профессионального каркаса ПО): AbstractFactory, Adapter, Builder, Decorator, Facade, FactoryMethod,

Memento, Observer, Prototype, Singleton, Strategy. Пример использования паттерна в коде программы:

```
" new Hi("0").add("11").add("222").add("3333").add("4444").prn();"
```

Библиотеки для импорта/экспорта данных в офисные форматы

Для создания отчетов в программах на Java часто используется экспорт данных в форматы современных офисных документов (doc, docx, xls, xlsx), а для ввода данных в программу наоборот, импорт данных из офисных форматов. Наиболее просто и часто это осуществляется через библиотеки Apache POI [13]. Пример записи данных в XLSX-файл с использованием данных библиотек:

```
public static void writeXLSXFile(String xlsxFilename) throws IOException {  
    XSSFWorkbook wb = new XSSFWorkbook(new  
FileInputStream(xlsxFilename));  
    XSSFSheet sheet = wb.getSheetAt(0);  
    for (int r = 0; r < 3; r++) {  
        XSSFRow row = sheet.createRow(r + 9);  
        for (int c = 0; c < 3; c++) {  
            row.createCell(c + 8).setCellValue(String.valueOf(2 * (r + c)));  
        }  
    }  
    FileOutputStream fos = new FileOutputStream(xlsxFilename);  
    wb.write(fos);  
    fos.flush();  
    fos.close();  
}
```

Системные технологии

Промышленные программы используют следующие современные системные технологии в своей работе [4, 14-15]:

- автотестирования кода JUNIT4 (класс org.junit.Test). Пример тестирования кода показан на рисунке 3;
- локализация строк и интерфейсов для отображения на нужном пользователю языке (классы java.util.Locale, java.util.ResourceBundle);
- логирование кода для выявления ошибок исполнения через log-файлы (класс java.util.logging). Пример содержимого log-файла с логами показан на рисунке 4;
- хранение и чтение данных с файлов настроек и xml-файлов (классы java.util.prefs.Preferences, java.util.Properties);
- сериализация данных (объекты как текст в файлы) и десериализация данных (текст из файлов как объекты) (классы java.io.*);

- исполнение стороннего кода, написанного на языке C и C++ (вызов системных библиотек и модулей) (классы com.sun.jna.*,jdk.incubator.foreign.*);
- работа с потоками данных для производительной работы (классы java.util.concurrent.*).

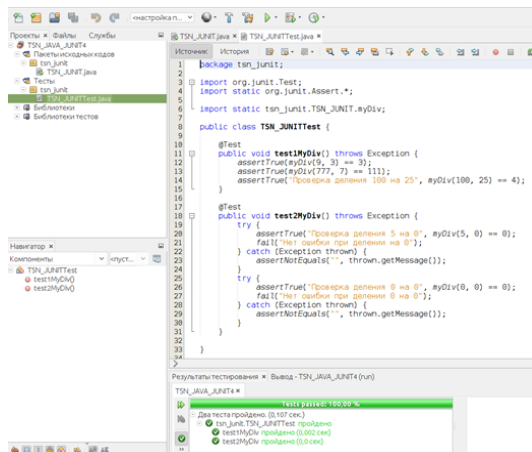


Рисунок 3 - Пример тестирования кода

```
2019-12-03 18:24:39 [INFO ] java_logs.MY_JAVA_CLASS - Start
2019-12-03 18:24:39 [INFO ] java_logs.MY_JAVA_CLASS - Запись лога с уровнем INFO (информационная)
2019-12-03 18:24:39 [WARNING] java_logs.MY_JAVA_CLASS - Запись лога с уровнем WARNING (Предупреждение)
2019-12-03 18:24:39 [SEVERE ] java_logs.MY_JAVA_CLASS - Запись лога с уровнем SEVERE (серьезная ошибка)
2019-12-03 18:24:39 [INFO ] java_logs.MY_JAVA_CLASS - Some message
2019-12-03 18:24:39 [SEVERE ] java_logs.MY_JAVA_CLASS - My Exception: ERR!
2019-12-03 18:24:39 [INFO ] java_logs.MY_JAVA_CLASS - End
2019-12-03 18:24:39 [INFO ] MY_JAVA_CLASS:49 - Start log4j
2019-12-03 18:24:39 [INFO ] MY_JAVA_CLASS:50 - Hi Logger info!
2019-12-03 18:24:39 [WARN ] MY_JAVA_CLASS:51 - Warning!
2019-12-03 18:24:39 [ERROR ] MY_JAVA_CLASS:52 - Error!
2019-12-03 18:24:39 [FATAL ] MY_JAVA_CLASS:53 - Fatal error!
2019-12-03 18:24:39 [INFO ] MY_JAVA_CLASS:54 - End log4j
```

Рисунок 4 – Пример содержимого файла с логами

Выводы

Все рассмотренные в данной статье технологии являются базовыми и основными для разработки промышленных программ на Java. Существуют и другие технологии, но те, что рассмотрены, в обязательном порядке должны быть известны каждому современному программисту на Java.

В Торайгыров университете г. Павлодара эти технологии студенты изучают на дисциплинах «Программирование на Java», «Современное объектно-ориентированное программирование», «Промышленное программирование», «Разработка многоуровневых приложений» и после активно используют в своей профессиональной деятельности и дальнейшей учебе.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 **Steve Loughran, Erik Hatcher**. Ant in Action. – Manning Publications, 2007. – 600 p.
- 2 **Jonathan Lalou**. Apache Maven Dependency Management. – Packt Publishing, 2013. - 158 p.
- 3 **Benjamin Muschko**. Gradle in Action Author. – Manning Publications, 2014. – 480 p.
- 4 **Training courses by Talipov S. N.** [Электронный ресурс]. – Режим доступа: <https://github.com/tsnsoft?q=java>
- 5 **Joel Murach**. Murach's MySQL (3rd Edition). – Mike Murach & Associates, 2019. - 628 p.
- 6 **Hans-Jurgen Schonig**. Mastering PostgreSQL 13. – Packt Publishing, 2020. – 476 p.
- 7 **Christian Bauer, Gavin King, Gary Gregory**. Java Persistence with Hibernate 2nd Edition. – Manning, 2015. - 608 p.
- 8 **Шефер Крис, Хо Кларенс, Харрон Роб**. Spring 4 для профессионалов, 4-е изд. : Пер. с англ. – М. : ООО «И.Д. Вильямс», 2015. – 752 с.
- 9 **Прохоренок, Н. А.** JavaFX. – СПб. : БХВ-Петербург, 2020. – 768 с.
- 10 **David, R. Heffelfinger**. Java EE 7 Development with NetBeans 8. – Packt Publishing, 2015. – 364 p.
- 11 **Гонсалвес, Э.** Изучаем Java EE7. – СПб. : Питер, 2014. – 640 с.
- 12 **Фримен, Э., Робсон, Э., Сьерра, К., Бейтс, Б.** Head First. Паттерны проектирования. – СПб. : Питер, 2022. – 640 p.
- 13 **Vitalii, Serdiuk**. Apache POI Handbook : Java + Apache POI tutorial. – Kindle Store, 2018. – 560 p.
- 14 **Catalin Tudose**. JUnit in Action, Third Edition. – Kindle, 2022. – 560 p.
- 15 **Javier Fernandez Gonzalez**. Mastering Concurrency Programming with Java 9. – Packt Publishing, 2017. – 516 p.

REFERENCES

- 1 **Steve Loughran, Erik Hatcher**. Ant in Action. – Manning Publications, 2007. – 600 p.
- 2 **Jonathan Lalou**. Apache Maven Dependency Management. – Packt Publishing, 2013. - 158 p.
- 3 **Benjamin Muschko**. Gradle in Action Author. – Manning Publications, 2014. – 480 p.
- 4 **Training courses by Talipov S.N.** [Electronic resource]. – Access mode: <https://github.com/tsnsoft?q=java>

- 5 **Joel Murach**. Murach's MySQL (3rd Edition). – Mike Murach & Associates, 2019. – 628 p.
- 6 **Hans-Jurgen Schonig**. Mastering PostgreSQL 13. – Packt Publishing, 2020. – 476 p.
- 7 **Christian Bauer, Gavin King, Gary Gregory**. Java Persistence with Hibernate 2nd Edition. – Manning, 2015. – 608 p.
- 8 **Shefer Kris, Xo Klarens, Xarrop Rob**. Spring 4 dlya professionalov, 4-e izd. [Spring 4 for professionals, 4th ed.: Per. from English]. – Moscow : ООО «I.D. Vil'yame», 2015. – 752 p.
- 9 **Proxorenok, N. A.** JavaFX [JavaFX]. – SPb. : BXV-Peterburg, 2020. – 768 p.
- 10 **David, R. Heffelfinger**. Java EE 7 Development with NetBeans 8. – Packt Publishing, 2015. – 364 p.
- 11 **Gonsalves E.** Izuchaem Java EE7 [Learning Java EE7]. – SPb. : Piter, 2014. – 640 p.
- 12 **Frimen, E., Robson, E., S'erra, K., Bejts, B.** Head First. Patterny' proektirovaniya [Head First. Design patterns]. – SPb. : Piter, 2022. – 640 p.
- 13 **Vitalii Serdiuk**. Apache POI Handbook: Java + Apache POI tutorial. – Kindle Store, 2018. – 560 p.
- 14 **Catalin Tudose**. JUnit in Action, Third Edition. – Kindle, 2022. – 560 p.
- 15 **Javier Fernandez Gonzalez**. Mastering Concurrency Programming with Java 9. – Packt Publishing, 2017. – 516 p.

Материал поступил в редакцию 07.06.22.

*С. Н. Талипов

Торайгыров университеті, Қазақстан Республикасы, Павлодар қ.
Материал баспаға 07.06.22 түсті.

ЖОО-ДА JAVA-ДА ӨНЕРКӘСІПТІК БАҒДАРЛАМАЛАУДЫ ОҚЫТУ

Мақала ең алдымен бағдарламалаушы студенттерге, сондай-ақ Java-да заманауи өнеркәсіптік бағдарламалауға қызығушылық танытқандарға арналған.

Қазіргі уақытта Java өнеркәсіп пен ірі бизнеске арналған бағдарламаларды әзірлеудің негізгі тілі болып табылады. Бұл тілдің кросс-платформасына, оның тегін болуына, сондай-ақ бағдарламалық жасақтаманы әзірлеудің объектіге бағытталған принциптеріне байланысты. Сондықтан оның барлық негізгі

технологиялары мен мүмкіндіктерін білу өте маңызды. Бұл технологиялар Java тілінің негізіне қондырма болып табылады және «өнеркәсіптік бағдарламалау» терминімен біріктіріледі.

Бұл мақалада қазіргі уақытта өзекті болып табылатын және соңғы 5 жыл ішінде Қазақстан Республикасының мұнай өңдеу саласы үшін коммерциялық бағдарламаларды әзірлеу кезінде автордың жеке өзі қолданған және сынақтан өткізген Java-да өнеркәсіптік бағдарламалаудың барлық негізгі технологиялары қарастырылады.

Ұсынылған мақалада бастапқы кодтан Java бағдарламаларын құрастыру технологиялары, мәліметтер базасы және олармен жұмыс істеу технологиялары, пайдаланушының графикалық интерфейсін құру технологиялары, жселілік технологиялар, кеңейтілген ООР технологиялары, кеңсе форматтарына деректерді импорттау/экспорттау кітапханалары, сонымен қатар Жүйелік технологиялар қарастырылған.

Арқасында мақаласында ұсынылған ақпарат, инженер-программист білуге болады, ненің көмегімен жасауға болады, нақты өндірістік жоба Java.

Осы мақалада қарастырылған барлық технологиялар Java-да өнеркәсіптік бағдарламаларды әзірлеу үшін негізгі және негізгі болып табылады. Басқа технологиялар бар, бірақ қарастырылған технологиялар Java-дағы әрбір заманауи бағдарламашыға белгілі болуы керек.

Кілтті сөздер: Java, өнеркәсіптік бағдарламалау, мәліметтер базасы, объектіге бағытталған бағдарламалау, бағдарлама, технологиялар.

*S. N. Talipov

Toraighyrov University, Republic of Kazakhstan, Pavlodar.

Material received on 07.06.22.

TEACHING INDUSTRIAL PROGRAMMING IN JAVA AT A UNIVERSITY

The article is intended primarily for students-programmers, as well as anyone interested in modern industrial programming in Java.

Currently, Java is the main programming language for industry and large businesses. This is due to the cross-platform nature of the language, its free of charge, as well as the object-oriented principles of software development embedded in it. Therefore, it is very important to know all

its basic technologies and capabilities. These technologies are an add-on to the basis of the Java language and are combined under the term «industrial programming».

This article discusses all the main technologies of industrial programming in Java, which are relevant at the moment and which were personally used and tested by the author in the development of commercial programs for the oil refining industry of the Republic of Kazakhstan over the past 5 years.

This article discusses technologies for building Java programs from source code, databases and technologies for working with them, technologies for creating a graphical user interface, network technologies, advanced OOP technologies, libraries for importing/exporting data to office formats, as well as system-wide technologies.

Thanks to the information provided in the article, a software engineer will know how to make a real industrial project in Java.

All the technologies discussed in this article are basic and basic for the development of industrial programs in Java. There are other technologies, but those that are considered must necessarily be known to every modern Java programmer.

Keywords: Java, industrial programming, databases, object-oriented programming, software, technologies.

Теруге 07.06.2022 ж. жіберілді. Басуға 30.06.2022 ж. қол қойылды.

Электрондық баспа

8,05 Mb RAM

Шартты баспа табағы 8,57. Таралымы 300 дана. Бағасы келісім бойынша.

Компьютерде беттеген: З. С. Исакова

Корректор: А. Р. Омарова

Тапсырыс № 3960

Сдано в набор 07.06.2022 г. Подписано в печать 30.06.2022 г.

Электронное издание

8,05 Mb RAM

Усл.печ.л. 8,57. Тираж 300 экз. Цена договорная.

Компьютерная верстка З. С. Исакова

Корректор: А. Р. Омарова

Заказ № 3960

«Toraighyrov University» баспасынан басылып шығарылған

«Торайғыров университеті» КЕ АҚ

140008, Павлодар к., Ломов к., 64, 137 каб.

«Toraighyrov University» баспасы

«Торайғыров университеті» КЕ АҚ

140008, Павлодар к., Ломов к., 64, 137 каб.

67-36-69

e-mail: kereku@tou.edu.kz

www.vestnik.tou.edu.kz

<https://vestnik-pm.tou.edu.kz/>