

## Почтовые уведомления в Java

*Еровлев Павел Андреевич*  
*Приамурский государственный университет имени Шолом-Алейхема*  
*Студент*

*Еровлева Регина Викторовна*  
*Приамурский государственный университет имени Шолом-Алейхема*  
*Студент*

### Аннотация

В данной статье рассматривается как создать почтовые уведомления. Для реализации будет использоваться язык программирования Java и фреймворк Spring Boot. Конечным результатом будут являться работающие почтовые уведомления.

**Ключевые слова:** Java, Почта, Spring Boot

## Mail Notification in Java

*Erovlev Pavel Andreevich*  
*Sholom-Aleichem Priamursky State University*  
*Student*

*Eroleva Regina Victorovna*  
*Sholom-Aleichem Priamursky State University*  
*Student*

### Abstract

This article discusses how to create mail notifications. For implementation, the Java programming language and the Spring Boot framework will be used. The end result will be working email notifications.

**Keywords:** Java, Mail, Spring Boot

### 1 Введение

#### 1.1 Актуальность

Электронная почта — важная функция любого корпоративного Java-приложения. Любому приложению потребуется функция электронной почты для отправки напоминаний, счетов, платежей, подтверждений, паролей, предупреждений и некоторых других видов системных уведомлений.

#### 1.2 Обзор исследований

А.Н. Иванов и П.С. Власюк описали в своей работе реализацию безопасного веб-приложения на языке программирования Java [1].

М.А.Потовиченко, М.В. Привалов и С.В. Корнев рассмотрели разработку программного продукта, который обеспечивает учет данных посещения занятий студентов, а также защиту их работ [2]. В.А. Сухомлин описал кратко в своей статье принцип работа комплексной программы дополнительного образования, ориентированную на подготовку разработчиков Java enterprise приложений [3]. Д.В. Козырев, Л.А. Володченкова разработали программный интерфейс серверной части для облачного хранилища данных [4]. Ю.А.Флёрв и Л.Л. Вышинский разработали программу для организации взаимодействия с Web-клиентами в программных комплексах [5].

### **1.3 Цель исследования**

Цель исследования – используя язык программирования Java, реализовать почтовые уведомления.

## **2 Материалы и методы**

Работа по реализации почтовых уведомлений происходит с использованием языка программирования Java, фреймворка SpringBoot и IDE IntelliJIdea.

## **3 Результаты и обсуждения**

Вряд ли найдется какое-либо корпоративное приложение на Java, которое не отправляет электронную почту. Это может быть по функциональным причинам или нефункциональным требованиям, но время от времени необходимо будет писать код для отправки электронных писем, но в Java это не так просто.

Необходимо включить «Java Mail API» и изучить некоторые ключевые классы и способы их настройки для отправки электронных писем из программ Java.

Но для использования будет рассмотрена библиотека «JavaMailSenderImpl». Она упрощает отправки электронных писем в Java.

Это производственная реализация интерфейса «JavaMailSender», которая поддерживает как «SimpleMailMessage» из «Spring», так и «MimeMessages» из «JavaMail API».

Лучшее в этом классе то, что он позволяет определять настройки почты как свойства bean-компонента, а также использовать предварительно сконфигурированный сеанс «JavaMail», который можно получить из среды JNDI сервера приложений.

Начнем реализовывать отправки электронной почты из приложения Java с использованием Spring Framework и его служебных классов, таких как «JavaMailSenderImpl», еще одного класса шаблона, который устраняет большую часть ошибок, связанной с отправкой электронных писем с использованием почтового API Java.

Для начала необходимо настроить «JavaMailSenderImpl» как компонент Spring в «ApplicationContext.xml» (рис.1)

```
1 <bean id="mailSender" class="org.springframework.mail.javamail.JavaMailSenderImpl">
2   <property name="host" value="smtp.gmail.com"/>
3   <property name="port" value="25"/>
4   <property name="username" value="pasandreg@gmail.com"/>
5   <property name="password" value="#password"/>
6   <property name="javaMailProperties">
7     <props>
8       <prop key="mail.transport.protocol">smtp</prop>
9       <prop key="mail.smtps.auth">true</prop>
10      <prop key="mail.smtp.starttls.enable">true</prop>
11      <prop key="mail.smtp.ssl.enable">>false</prop>
12      <prop key="mail.debug">true</prop>
13    </props>
14  </property>
15 </bean>
16
17
```

Рисунок 1 – ApplicationContext.xml

Далее необходимо настроить тестовое приложение со ссылкой на «JavaMailSenderImpl» (рис.2).

```
1 <bean id="javaApp" class="com.javarevisited.demo.SpringEmailDemo">
2   <property name="mailSender" ref="mailSender"/>
3 </bean>
4
```

Рисунок 2 – Настройка приложения

Теперь создадим класс для обработки и отправки сообщений (рис.3).

```
9 package com.javarevisited.demo;
10
11 public class SpringEmailDemo {
12
13   private JavaMailSender mailSender;
14
15   public MimeMessage getMailMessage() throws MessagingException {
16     final MimeMessage mimeMessage = this.mailSender.createMimeMessage();
17     MimeMessageHelper helper = new MimeMessageHelper(mimeMessage, false);
18     helper.setFrom("pasandreg@gmail.com");
19     helper.setTo("tankolet123@mail.ru");
20     helper.setCc("reginasemchenko@mail.ru");
21     helper.setBcc("pavel.erovlev@gmail.com");
22     helper.setSubject("Сообщение из Java приложения");
23     helper.setText("Это тестовое сообщение", true);
24     return mimeMessage;
25   }
26
27
28   public void sendMail() throws MailException, MessagingException {
29     mailSender.send(getMailMessage());
30   }
31
32
33 }
34
```

Рисунок 3 – Класс для отправки сообщений

Метод «setFrom» устанавливает почту отправителя, «setTo» указывает кому будет отправлено сообщение, «setCc» позволяет поставить адрес в копию письма, а «setBcc» установит адрес в скрытую копию получателя.

После выполнения класса на все адреса отправились сообщения (рис.4-5).

## Сообщение из Java приложения

-  Pavel Сегодня, 19:39  
Кому: вам, Регина Семченко

Это тестовое сообщение

Рисунок 4 – Сообщение для адреса tankolet123@mail.ru

По данным из сообщения видно, что скрытого получателя не видно.

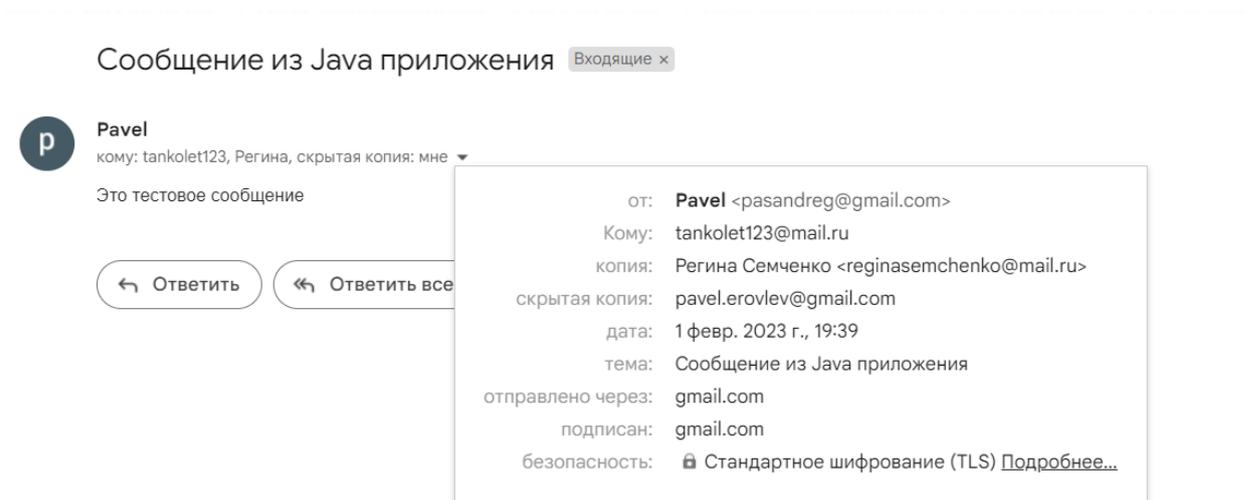


Рисунок 5 – Сообщение для скрытого получателя

### Выводы

В данной статье был рассмотрен процесс реализации по отправки электронных писем с Java с помощью среды Spring.

Spring отлично подходит для отправки всех видов электронных писем, таких как электронное письмо с вложением, электронное письмо с содержимым HTML, электронное письмо с изображениями и так далее.

### Библиографический список

1. Иванов А.Н., Власюк П.С. Разработка приложения с использованием веб-технологий java // Тихоокеанский государственный университет. 2020. С. 261-266.
2. Потовиченко М.А., Привалов М.В., Корнев С.В. Компьютеризированная

- подсистема учета текущей успеваемости студента в условиях вуза // Информатика, управляющие системы, математическое и компьютерное моделирование. 2019. № 2. С. 71-75.
3. Сухомлин В.А. Подготовка разработчиков корпоративных java-приложений в режиме дистанционного обучения // Информатика, управляющие системы, математическое и компьютерное моделирование. 2013. № 9. С. 175-180.
  4. Володченкова Л.А., Козырев Д.В. Разработка серверной части программного приложения для удаленного хранения данных// Математические структуры и моделирование. 2020. № 1 (53). С. 108-138.
  5. Флёров Ю.А. и Вышинский Л.Л. Программа организации интерфейса web-серверов с java-приложениями // Аллея науки. 2016. № 7. С. 58-61.