

Platformy Programistyczne Java

Laboratorium 6

JavaMail API

```
<dependency>  
  <groupId>javax.mail</groupId>  
  <artifactId>javax.mail-api</artifactId>  
  <version>1.5.5</version>  
</dependency>
```

MIME: Multipurpose Internet Mail Extension

- Standard stosowany przy wysyłaniu poczty elektronicznej
- Tradycyjny email umożliwiał tylko przesyłanie tekstu
- MIME definiuje mechanizmy do przesyłania innego rodzaju informacji:
 - Tekstu w językach używających innego kodowania znaków niż ASCII,
 - Plików zawierających obrazy, dźwięki, filmy, programy, etc.

Krok 1: Konfiguracja serwera SMTP dla gmail'a

```
Properties props = System.getProperties();  
// host usługi SMTP  
props.put("mail.smtp.host", "smtp.gmail.com");  
// port SSL - 465  
props.put("mail.smtp.socketFactory.port", "465");  
// StartTLS - ustanawia szyfrowanie TLS w połączeniu sieciowym  
props.put("mail.smtp.socketFactory.class", "javax.net.ssl.SSLSocketFactory");  
// wymusza autentykację (login i hasło do konta na Gmailu)  
props.put("mail.smtp.auth", "true");
```

Krok 2: Stworzenie sesji

```
Session emailSession = Session.getInstance(props, new UserAuthenticator());
```

```
// przykładowa implementacja autentykacji
```

```
class UserAuthenticator extends Authenticator {  
  
    @Override  
    protected PasswordAuthentication getPasswordAuthentication() {  
        dialog.pack();  
        dialog.setVisible(true);  
        String login = dialog.getLogin();  
        String password = dialog.getPassword();  
        PasswordAuthentication auth = null;  
        return new PasswordAuthentication(login, password);  
    }  
}
```

Krok 3: Stworzenie wiadomości

```
MimeMessage message = new MimeMessage(emailSession);  
  
message.addRecipient(RecipientType.TO, new InternetAddress("x@x.com"));  
  
message.setSubject("Platformy programistyczne");  
  
message.setFrom(new InternetAddress("y@y.com", "Nazwa nadawcy"));
```



Ta nazwa wyświetli się odbiorcy

Adres nadawcy zostanie nadpisany przez
adres email podany przy autentykacji

Krok 4: Treść maila w formacie HTML

```
BodyPart bodyPart = new MimeBodyPart();

/* Łączenie wielu String'ów używając StringBuilder'a jest wydajniejsze niż
używanie operatora + */
StringBuilder emailBody = new StringBuilder();
emailBody.append("Tutaj jakaś treść mail'a");

// dla każdej linii ASCII Art
    emailBody.append(liniaAsciiArt).append("<br/>");

bodyPart.setContent(emailBody.toString(), "text/html");
```

Krok 5: Załącznik

```
BodyPart attachmentPart = new MimeBodyPart();
```

```
Path pathToFile = ... // ścieżka do pliku, np. graficznego
```

```
// obsługa danych z pliku - żeby BodyPart wiedział, co z tym zrobić
```

```
DataSource source = new FileDataSource(pathToFile.toFile());
```

```
attachmentPart.setDataHandler(new DataHandler(source));
```

```
// przyjazna pliku, w przeciwnym razie nazwą pliku będzie cała ścieżka
```

```
attachmentPart.setFileName(pathToFile.getFileName().toString());
```


Krok 6: Pełna zawartość maila

```
Multipart container = new MimeMultipart();  
  
container.addBodyPart(bodyPart); // treść maila z kroku 4  
  
container.addBodyPart(attachmentPart); // załącznik z kroku 5  
  
message.setContent(container);
```

Krok 7: Wysłanie

```
// import  
  
import javax.mail.Transport;  
  
...  
  
// wysyłanie  
  
Transport.send(message);
```

Krok 8: Ustawienie Gmail'a

Ustawienia -> Logowanie i bezpieczeństwo -> Połączone aplikacje i witryny

Logowanie i bezpieczeństwo

Logowanie się w Google

Aktywność urządzeń i powiadomienia

Połączone aplikacje i witryny

Zezwalaj na mniej bezpieczne aplikacje: WŁĄCZONE



Niektóre aplikacje i urządzenia firm innych niż Google korzystają z mniej bezpiecznych technologii logowania, co może obniżyć bezpieczeństwo Twojego konta. Możesz wyłączyć dostęp takich aplikacji (i to zalecamy) lub nadal ich używać mimo zagrożeń.