

CPE 372 Object Oriented Analysis and Design

Exercise 1

Creating a Simple Class

In the **Lecture1** subdirectory under **demos**, you will find two source files for two Java classes: **IOUtils.java** and **EmailTester.java**. Download these files to your computer. Be sure you have installed the Java 8 SDK.

The first class provides a number of static methods to help you do terminal input/output in Java.

The second class is a tester class for a class called **EmailMessage**. *Your task in this exercise is to write this Java class.*

Below you will find a sample run of **EmailTester**. The program should loop, creating email messages, until the user says “N” to the “Continue?” question. Red text is entered by the user.

```
[goldin@rinjani Lecture1]$ java EmailTester
==> Enter To email address
john@gmail.com
==> Enter From email address
helen@kmutt.ac.th
==> Enter email subject
Let's be friends
==> Enter message text below. Type END to finish.
Dear John,
I have seen you walking around the campus very often.
You look like a really interesting person.
Can we be friends? Would you like to meet for a coffee?
Please let me know!

All the best,
Helen
END

Send?
Y
-----
[Message created at 2017-06-26 17:10:15]
-----
TO:    john@gmail.com
FROM:   helen@kmutt.ac.th
SUBJECT: Let's be friends
- - - - -
Dear John,
I have seen you walking around the campus very often.
You look like a really interesting person.
Can we be friends? Would you like to meet for a coffee?
Please let me know!
```

All the best,
Helen

[Message sent at 2017-06-26 17:11:26]

Sent message to john@gmail.com

Continue?

N

The **EmailMessage** class should have the member variables and the methods needed by the code in **EmailTester**. Specifically it needs the following member data:

- A string to store the date and time the message is first created. Use the **IOUtils.getDateTime()** method to get this string. You must initialize this string in the constructor method for **EmailMessage**.
- Strings to hold the to address, from address, and subject line.
- Some way to store a multiple line message body.

Your class also needs the following methods. If you look at the code for **EmailTester.java**, you will get some idea what arguments and return values these methods should have.

- setAddress()
- setFromAddress()
- setSubject()
- addToBody()
- send() - This should print out the entire message to the terminal as shown above.
- getAddress() -- it would be a good to have "get" functions for the other data too.

Compile your class and test it, until you get output roughly like the text above. *Do not change EmailTester.java!* Your class should work with my test class.

Upload your source file EmailMessage.java.

Do not forget to follow the Java coding standards.