Java Advanced CXEC-320

## Final Hands-On TEST - Wednesday, April 16, 2014

Student:				

## Be sure to read the following general instructions carefully:

- This lab test must be completed individually by all the students.
- Save your program periodically just in case that your PC crashes.

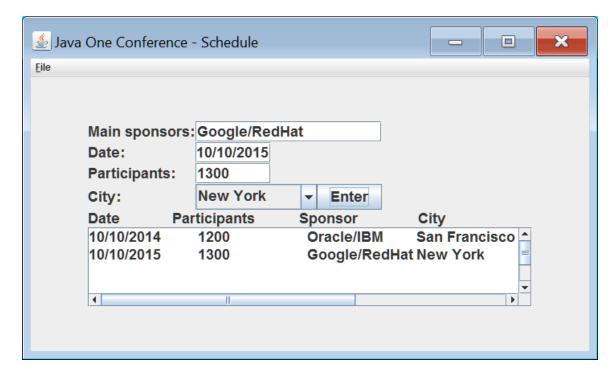
## Exercise 1

Start a new Eclipse Java project. Name the project:

YourFullName\_CXEC320FinalTest. For example: JohnSmith\_CXEC320FinalTest. DO NOT create a package in this project. Let Eclipse create the default package.

Write a GUI application in Java that helps organizers to keep track of **Java One** conferences. The minimum information that is required is shown in the following picture:

The user should be able to enter the *sponsor name* of Java One conference, *date*, *number of participants*, *and location*. The information should be displayed in a JTextArea component which has scrolling abilities, and at the same time should be stored in a file in serialized form (whenever user clicks on button Enter).



You will need to create the following classes:

Final Hands-On Test Page 1 of 2

Java Advanced CXEC-320

**ConferenceInfo** - to encapsulate the information (*String date*, *int nrOfParticipants*, *String sponsor*, *String city*). Provide the necessary *constructors* and *setter/getter* methods for this class.

**DataInfo** - to encapsulate input/output capabilities. This class will define the following methods:

```
public static void openFile()
public static void addRecords(ConferenceInfo info)
public static void closeFile()
```

Use **FileOutputStream/ObjectOutputStream** to create the file, save a record, and close the file.

**ConferenceGUI** - implements the GUI shown above and interacts with **ConferenceInfo** and **DataInfo** classes to store conference the information in the file.

Use the most appropriate layout manager classes to implement the layout of this GUI. Provide validation for user input.

**Bonus marks** (5): Allow the user to exit the application by clicking on File/Exit menu item, and use a GridbagLayout manager to implement the UI.

## **Evaluation:**

Total	- 70 marks
5. User friendliness	- 5 marks
4. Exception Handling/Validation	- 10 marks
3. Java APIs	- 20 marks
2. GUI	- 20 marks
1. OOD	- 15 marks

You should submit the entire project using <u>Assignment link</u> on Blackboard immediately after demonstration.

Final Hands-On Test Page 2 of 2