

Distributed Systems

Exercise Sheet 5, Tuesday, 16:00

Klingemann, SS 2023

Deadline: 13th June 2023

2nd Assessed Exercise

1. Date-Service using Java RMI

Understand the example code and test the programs. Client and server can run on the same machine. The nameservice has to be started by means of the rmiregistry command.

2. Distributed system for the management of wardrobes and their content using Java RMI

Extend your simple system for the management of wardrobes and their content from Sheet 3 to create a client-server-based system using Java RMI. Therefore, you have to transform the objects of the classes wardrobe and piece of clothing into remote objects. All wardrobe- and piece of clothing-objects exist only on the server! For each piece of clothing there exists a separate object. Similar to the description in Sheet 3, we require that each wardrobe has two private attributes: a name and a set of piece of clothing objects. This set should represent the pieces of clothing that are stored in the wardrobe. (Note that it is a set of objects and not just a number!) Each piece of clothing has three private attributes: a category (e.g., trouser, shirt), a colour and a size.

A wardrobe should have methods with the following functionality. All these methods can be invoked from the client.

- Search for a piece of clothing with a particular category and colour. You can assume that there exists at most one. The method returns a reference to the corresponding piece of clothing object.
- Add a new piece of clothing. The method has three parameters: a category, a colour and a size. A corresponding piece of clothing object is created and added to the set of pieces of clothing of the wardrobe.
- Return the set of all piece of clothing objects of the wardrobe. (The return value has to be a collection of references to a piece of clothing objects and not just a number!)
- Return the name of the wardrobe.

A piece of clothing should have methods with the following functionality. All these methods can be invoked from the client.

- Return the category
- Return the colour
- Return the size
- Change the size

Implement a client that is calling all methods of the objects on the server in a sensible manner. In particular, your client should be able to calculate based on the methods above the total number of all pieces of clothing with the colour “red”. Your application should use two wardrobe objects. The nameservice has to be started by means of the rmiregistry command.

Organisational matters

- You have to solve the exercise completely on your own! (No working in groups!)
- It is necessary but not sufficient to present a working program. Moreover, you have to be able to explain all parts of your program, be able to answer questions with respect to your program and make small extensions of your program.
- Your program has to be created completely within the exercise slot.
- If you violate one of the rules above, this implies that you definitely fail in this exercise.
- You can only present solutions that correspond to the exercise slot you are assigned to.
- It is in your responsibility to present your solution in time before the deadline. The assessment of your solution can only be guaranteed if you finish your program 60 minutes before the end of the exercises.
- To take part in the exam it is required to solve at least three of five assessed exercise sheets.