This lab session will give you practical experience with critical sections and mutual exclusion in Java. You will also revisit the Java thread life cycle.

For that purpose we want to implement the FSP model of the museum admission control system that we introduced in tutorial session 1. That means that we want the museum director, the entrance turnstile, the exit turnstile and the controller to be separate threads.

The director uses an applet that has two buttons to open and close the museum. The applet should display the number of visitors who have entered the museum, the number of visitors who are in the museum, and the number of visitors who have left the museum. Also it should show whether the entrance and exit are currently open.

**Exercise 1:**
Design classes and their relationships that are needed for the implementation of the museum admission control system. (Hint: opening and closing of the entrance exit turnstiles can be implemented as starting and stopping of threads)

**Exercise 2:**
What are the critical sections in your implementation? How do you synchronise them?

**Exercise 3:**
Implement the design in Java.