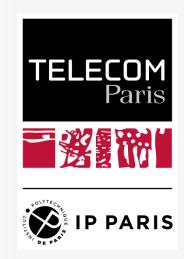
Java Threads ->

Basic LABS

Lab sources: <gitlab address>



Exo1: Basic Multi-Threads

- Define a Thread subclass called MyThread
 - Each instance receives a String name upon creation
 - The run() method
 - Prints a message iteratively, for 100 times
 - Each message: <the thread's name> and <iteration index: 1..100>
 - Sleeps for 50ms between iterations
 - Prints an END message before exiting
- In a main method, create 2 instances of MyThread and start them
 - What is the order of the message printouts?
 - Execute several times: is the order the same each time?
 - Same questions when: adding more threads; changing the sleep delays.



Exo2: Command Buffer

- Define a CommandsBuffer class as shown in the slides
 - Stores commands in an array: String[] commands
 - Provides push and pop methods to add and remove commands to/from the buffer, respectively
- Define two Thread types: PushThread and PullThread
 - Both take a CommandBuffer instance as input upon creation
 - Both act on the buffer for 100 iterations, with a sleep of 50ms in-between iterations
 - PushThread calls push on the buffer
 - PullThread calls pull on the buffer
- The main method creates an instance of each Thread type and starts them; use join at the end.
- Questions:
 - As the two threads push and pull 100 commands each, the buffer should be empty at the end is this so?
 - Remove all synchronisation commands from the push and pull methods (synchronize, wait and notifyall) is the outcome any different? why?
 - Leave synchronisation and remove the wait and notify commands from the push method what happens?