|  |  |
| --- | --- |
| **Exemplo 2 -sleep** |  |
| public class RandomLetters implements Runnable {  @Override public void run() {  try {  while(true) {  System.out.print(" " + (char)('A' + new Random().nextInt(26)));  Thread.sleep(200);  }  } catch (InterruptedException e) {  System.out.println("\n" + Thread.currentThread().getName() + " interrupted.");  System.out.println("INTERRUPTED flag: " + Thread.currentThread().isInterrupted());  }  System.out.println("Thread " + Thread.currentThread().getName() + " is DONE!");  }  } | public class InterruptSleepExample {  public static void main(String[] args) {  Runnable runnable = new RandomLetters();  Thread t1 = new Thread(runnable);  t1.start();  try { Thread.sleep(2000); } catch (InterruptedException ignored) {}  t1.interrupt(); // sets interrupt flag in t1  System.out.println("\nThread " + Thread.currentThread().getName() + " is DONE!");  }  } |
| **//Exemplo- redefinição do INTERRUPT**  public void run() {  while(true) { // loop continua mesmo com interrupçãoo  try {  // chamadas externas que poderão lidar com o INTERRUPT  Thread.sleep(100);  } catch(InterruptedException e) {  System.out.println("Thread interrompido que não será finalizado."));  Thread.currentThread().interrupt(); // IMPORTANTE!  }  }  System.out.println("Thread finalizado."));  } |  |
| **//Estrutura padrão para métodos run():**  public void run() {  try {  while(true) {  Thread.sleep(100);  }  } catch(InterruptedException e) {  System.out.println("Thread interrompido."));  }  System.out.println("Thread finalizado."));  } |  |
| **// Delay - o exemplo espera 10 segundos**  public class SimpleDelay implements Runnable {  int delay;  public SimpleDelay(int delay) {  this.delay = delay;  }  @Override public void run() {  System.out.println(Thread.currentThread().getName() + " started.");  try { Thread.sleep(delay); } catch (InterruptedException ignored) {}  System.out.println(Thread.currentThread().getName() + " finished.");  }  } | public class SleepExample {  public static void main(String[] args) {  Runnable r1 = new SimpleDelay(2000);  Runnable r2 = new SimpleDelay(5000);  new Thread(r1).start();  new Thread(r2).start();  try { Thread.sleep(10000); } catch (InterruptedException ignored) {}  System.out.println("Thread main is DONE!");  }  } |