

## Resumo da aula

### Transcrição

Vamos revisar o conteúdo rapidamente. Nesse exemplo, tivemos a necessidade de sincronizar o acesso. Quando compartilhamos um objeto com dois *threads*, em alguns casos gostaríamos de garantir que essas linhas de código sejam executadas de uma só vez (ou seja, de forma atômica).

Para isso, aprendemos o método `synchronized()`, que é baseado em uma chave ( `this` ), que também é chamada de **Mutex**. Cada objeto no mundo Java já possui essa chave para sincronizar o acesso. Dessa forma, o *thread* é executado continuamente, sem nunca liberar a chave e travando a aplicação.

Na próxima aula, trabalharemos outro exemplo, talvez um pouco mais real, com um objeto compartilhado. Depois, continuaremos trabalhando na classe `Banheiro`.

Faça os exercícios e até a próxima!