

## Mãos à obra: Distribuindo tarefas

Vamos implementar a nossa classe responsável por distribuir as tarefas em diferentes threads. Para deixar claro, as tarefas a executar serão apenas simuladas nesse curso, mas dentro de um projeto real poderiam executar um acesso demorado ao banco de dados ou um processamento em lote (por exemplo, uma leitura de um arquivo grande) ou executar chamadas de vários serviços web. Enfim, há várias aplicações do nosso projeto no mundo real :) Mãos à obra!

1) No projeto `servidor-tarefas` crie uma nova classe `DistribuirTarefas` no pacote `br.com.alura.servidor` que implementa a interface `Runnable`.

```
public class DistribuirTarefas implements Runnable {  
  
    @Override  
    public void run() {  
    }  
}
```

2) Dentro dessa classe crie um atributo `Socket socket` e receba-o no construtor da classe.

```
private Socket socket;  
  
public DistribuirTarefas(Socket socket) {  
    this.socket = socket;  
}
```

3) Implemente o método `run` exibindo uma mensagem avisando que estamos distribuindo as tarefas para um cliente e simule um processamento pesado colocando a `Thread` para dormir por 20 segundos:

```
public void run() {  
  
    System.out.println("Distribuindo as tarefas para o cliente " + socket);  
  
    try {  
        Thread.sleep(20000);  
    } catch (InterruptedException e) {  
        e.printStackTrace();  
    }  
}
```

4) Vamos também modificar a nossa classe `ServidorTarefas` para utilizar o nosso distribuidor de tarefas recém criado. No método `main` dentro do `while(true)` instancie um objeto `distribuirTarefas` passando para ele o `socket` obtido do servidor:

```
// na classe ServidorTarefas, dentro do main, dentro do while(true)  
DistribuirTarefas distribuirTarefas = new DistribuirTarefas(socket);
```

5) Ainda na classe `ServidorTarefas` crie uma nova thread que usa o objeto `distribuirTarefas` e de `start()` para iniciar a mesma:

```
Thread threadCliente = new Thread(distribuirTarefas);  
threadCliente.start();
```

6) Para testar, primeiramente verifique se não há nenhum outro servidor ou cliente rodando dentro do Eclipse (*não deveria ter nenhum console **ativo** no Eclipse*). Depois disso, rode a classe `ServidorTarefas` e depois execute pelo menos 2 clientes (não deveria ter nenhuma demora na hora de estabelecer a conexão entre cliente e servidor).