02

Achando o erro

Transcrição

De volta à aplicação web, o cliente nos reportou erro durante o cadastro de convidado. Vamos simular este cadastro mais uma vez, para confirmarmos o erro de validação dos dados, que ainda não foi resolvido. Para fins de teste, utilizamos então dados de um segundo usuário para cadastro, verificando-se que os dados foram armazenados com sucesso e aparecem na lista.

Portanto, descobrimos que não são todos os usuários que passam por este problema. Precisaremos debugar o método responsável por salvar os dados dos convidados para sabermos o que está acontecendo, sendo necessário o código fonte desta aplicação, para conseguirmos debugar.

Com o Tomcat pronto, necessitaremos do código localmente para encontrar a linha que contém o erro. O código em questão se encontra no GitHub (http://github.com/alura-cursos/listavipServlet), cujo *import* realizarei por meio do Git.

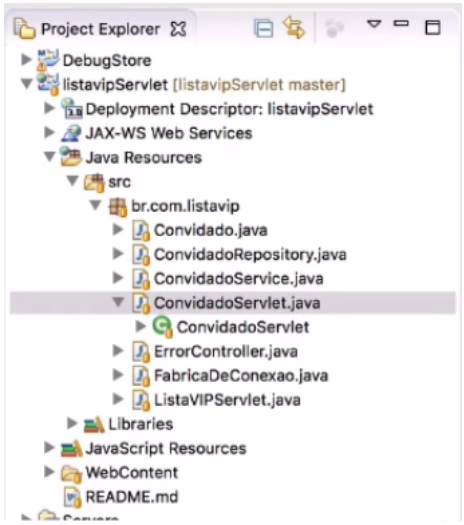
Com o Eclipse aberto, clicaremos com o botão direito do mouse em "Import > Import...". Na janela que se abriu, buscaremos a opção "Git", clicando em "Next" em seguida e escolhendo "Clone URI", apertando "Next" depois.

Copiaremos o link do GitHub ((http://github.com/alura-cursos/listavipServlet)http://github.com/alura-cursos/listavipServlet (http://github.com/alura-cursos/listavipServlet)) à janela aberta do Eclipse (é preciso estar online neste momento). Pergunta-se qual *branch* (ou ramo) se quer pegar, sendo que o "master" já vem selecionado. Não fazemos nenhuma alteração, e a próxima janela se refere ao diretório local em sua máquina, em que o arquivo a ser baixado será salvo.

Na próxima janela, manteremos o *checkbox* de "*import as general project*" marcado, pois será um projeto genérico, com mesmo nome utilizado no GitHub, `listavipServlet`. Clicaremos em "Finish", e o arquivo será baixado e colocado no *workplace* do Eclipse.

Por ser um projeto web, não há a opção de baixarmos neste modo pelo Git. Para isto, clicaremos em cima do arquivo, no lado esquerdo da tela, selecionando *"Properties*". Na janela nova, selecionamos "*Project Facets*", clicando no link "*Convert to faceted form...*", que realizará a conversão para uma aplicação web. Ali, duas opções precisam estar marcadas: Java, na versão `1.8`, e *Dynamic Web Module*, na versão `3.0`. Por fim, apertamos "*Apply*" e "OK" em seguida. O projeto foi baixado e transformado em um projeto web.

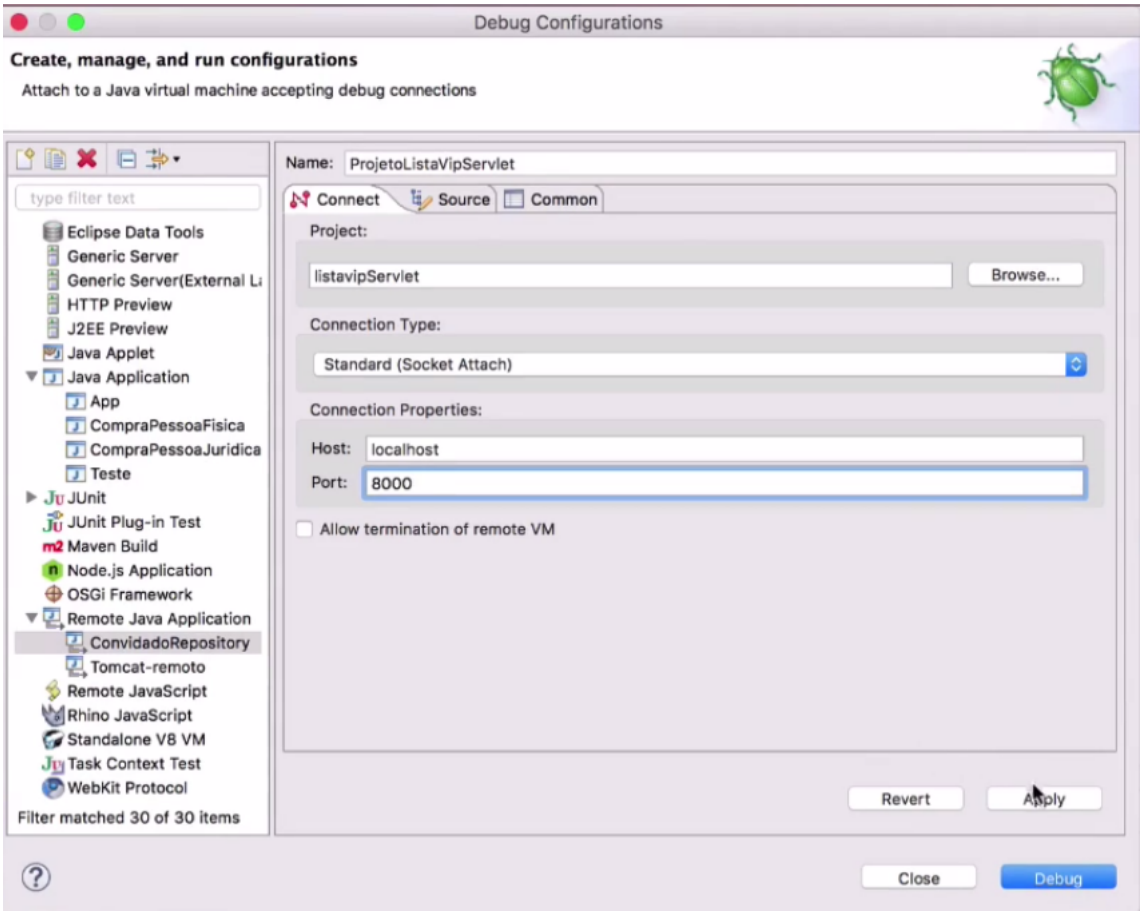
Agora, temos o Tomcat rodando de um lado, sendo iniciado em modo JPDA na porta `8000`, e podemos finalmente partir para o *Debug*. Vamos começar analisando o arquivo `ConvidadoServlet.java`:



Clicaremos no método `salvar()` localizado na linha `31`, mantendo o "Ctrl" (no Linux ou Windows, ou "Cmd" no Mac) pressionado, e entraremos nele. Ali, é criada uma conexão, o `insert` é executado na tabela `convidado`, fazendo-se um `PrepareStatement` ... É aqui que precisamos debugar, acrescentando-se um *breakpoint* ali.

Feito isto, precisaremos entrar no modo *Debug*, porém, não será um *Debug* normal, já que neste caso não existe método `main`, tampouco se trata de algo que possamos conectar a um servidor dentro do Eclipse, por exemplo. O container está em outro local, e só sabemos que iremos conectar a ele pela porta `8000`.

Como faremos este *Debug* remoto? Fizemos uma configuração específica para isto, então clicaremos com o lado direito do mouse no botão "*Debug* ConvidadoRepository.java", selecionando a opção "*Debug Configurations*", indo à "*Remote Java Application*" e apertando o lado direito do mouse novamente, selecionando "*New*". Faremos uma nova conexão, denominada "ProjetoListaVipServlet":

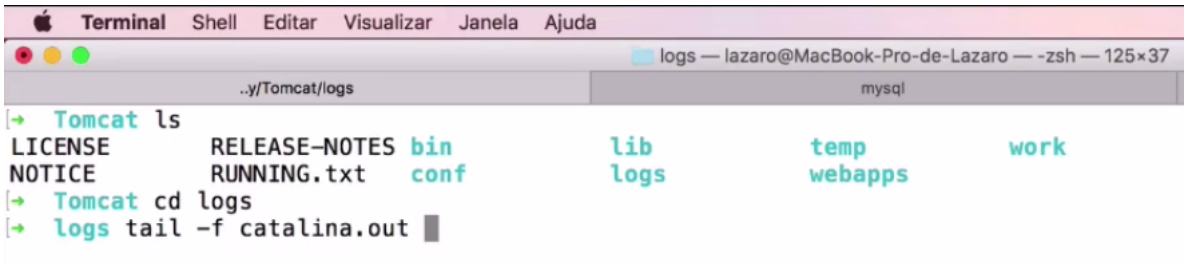


O *Host* se refere à localização do Tomcat; se ele for encontrado em outro IP, basta inseri-lo ali. Clicaremos em "*Apply*" para salvar estas configurações, e em "*Debug*" posteriormente. Feito isto, executaremos a aplicação novamente, tentando cadastrar aqueles dados repassados pelo cliente que está com problemas.

Voltamos à aplicação, submeteremos os dados do usuário e a janela é alternada automaticamente para o Eclipse, que nos pergunta se queremos mudar a perspectiva para *Debug*. Selecionamos "*Yes*", e repetiremos os passos vistos em vídeos anteriores, com `F6` ou "*Step Over*" para caminharmos linha a linha.

Criou-se a conexão, que verificamos na aba "*Variables*", uma *string* `insert` , o `preparedStatement` , e a execução. Houve um erro no `execute()` , pulando-se para o final. O programador colocou até um *Try/Catch*, que foi completamente omitido. É uma má prática, visto que foi colocado um *Try/Catch* para tratar uma exceção que, quando lançada, não foi nem printada, ou seja, não conseguimos nem ver esta exceção no log.

Vamos deixar a aplicação seguir para ver se aparece mais algo, a partir de `F8` (ou "*Resume*"). Nada apareceu, então vamos acessar o log do Tomcat. Para isto, abriremos o terminal, digitando `cd ..` para sairmos da pasta `bin` , onde estávamos anteriormente, para acessarmos a pasta `logs` :



O comando `tail -f` serve para pegar o final do arquivo, que no caso é `catalina.out` , arquivo de saída do log. Não aparece nenhum erro de log, pois o erro não foi printado, como dissemos.

Não é a melhor opção, porém é o básico: vamos imprimir este `StackTrace()`, acrescentando as seguintes linhas após o `catch` na aba `ConvidadoRepository.java`:

```
e.printStackTrace();
```

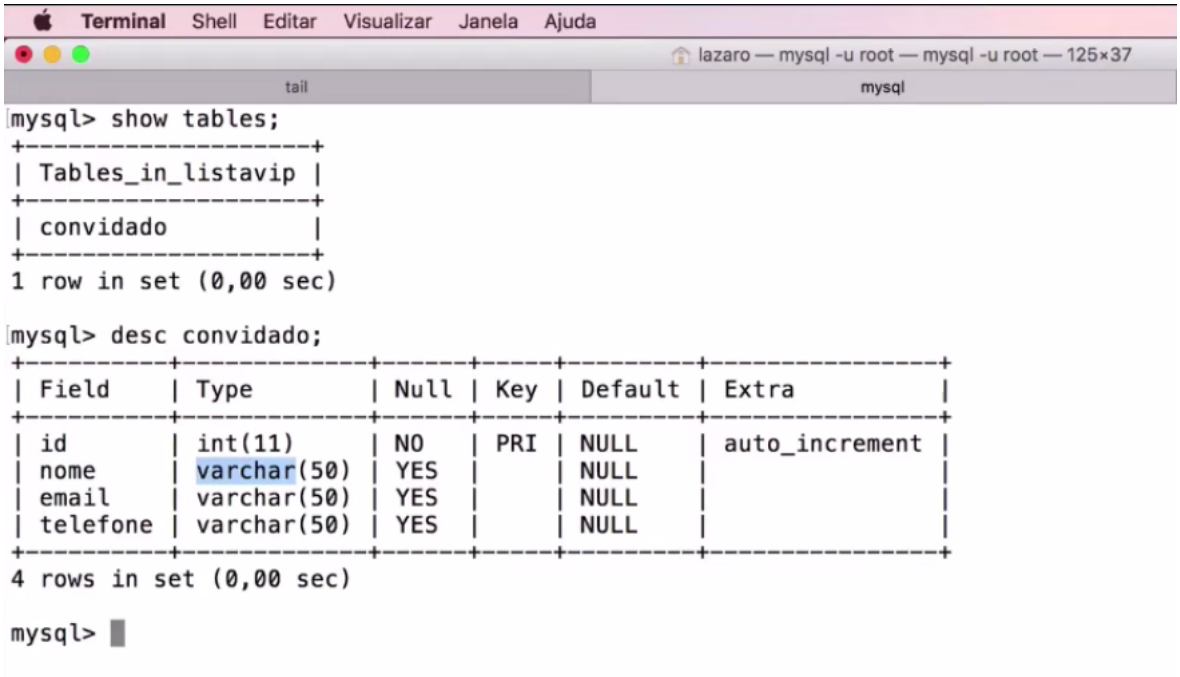
Salvaremos e rodaremos a aplicação mais uma vez, voltando à página de cadastro e inserindo todos aqueles dados do usuário novamente. Somos levados de volta ao Eclipse, e vamos pulando linha a linha, desta vez passando de `execute` para o Try, pelo menos. Houve um *print*, então apertaremos o "*Resume*" e deixaremos a aplicação rodar.

No terminal, verificaremos se o arquivo `catalina.out` está rodando. Encontraremos a mensagem `Data truncation: Data too long for column 'nome' at row 1`, ela nos indica que o nome do convidado está muito comprido.

Na aba "mysql" do terminal, digitaremos `show tables;` para visualizarmos como está a coluna "nome" e saber quantos caracteres podemos colocar ali. Digitaremos:

```
desc convidado;
```

O limite de caracteres permitido nesta coluna é de `50`, e o nome do usuário que estávamos tentando cadastrar possui `55`.



Encontramos o erro a partir do *Debug* remoto, no "mysql". Corrigiremos isto mais adiante, até a próxima.