

Introdução ao AngularJS

Professor Anderson Henrique



Testando nosso projeto CRUD

Vamos utilizar a ferramenta POSTMAN no nosso controller ChamadoController testando as operações GET, POST, PUT e DELETE

Antes vamos alterar o nosso arquivo **pom.xml**, pois, está faltando as **dependências** que produzem e consomem dados do tipo **json**

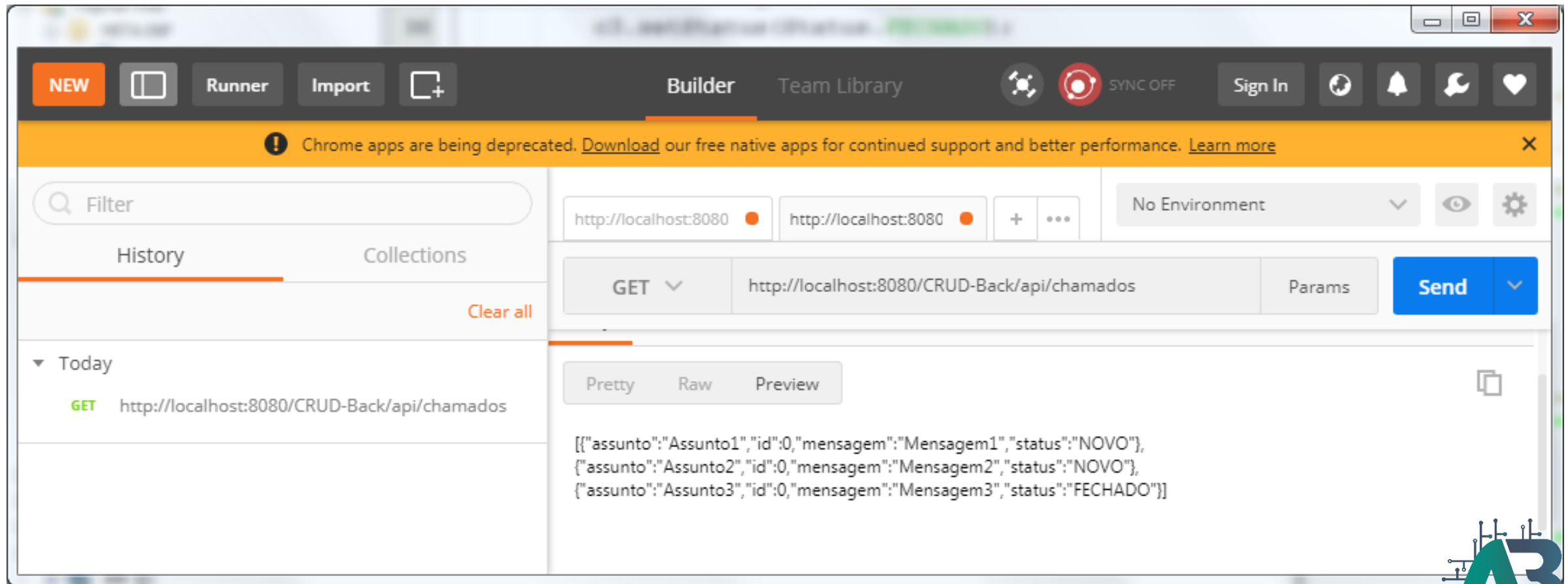
Esta dependência está no groupId: org.glassfish.Jersey.media, artifactId: jersey-media-moxy, version: 2.0

```
30  <dependency>
31      <groupId>org.glassfish.jersey.media</groupId>
32      <artifactId>jersey-media-moxy</artifactId>
33      <version>2.0</version>
34  </dependency>
```

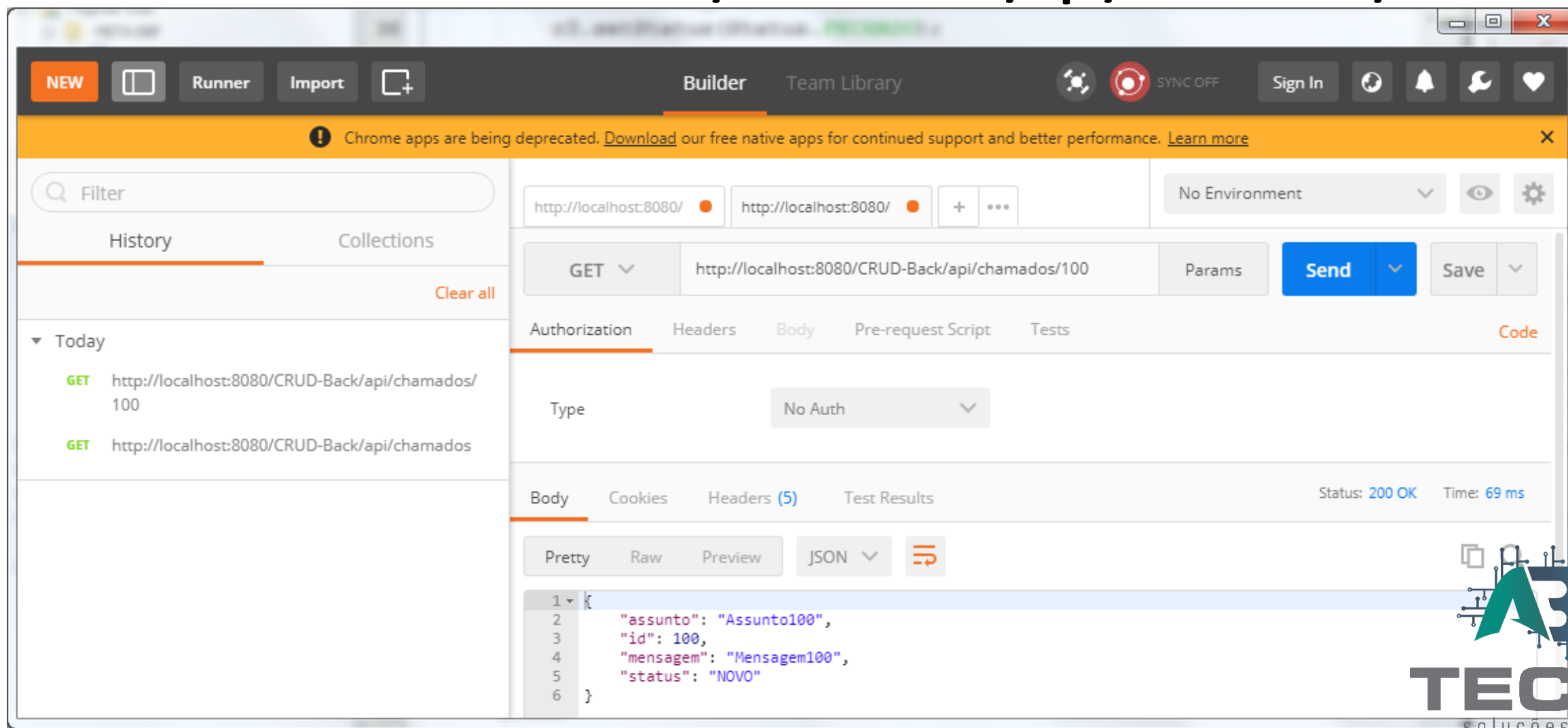
Vamos depurar (debugar) o projeto, depois que for construído vamos abrir o POSTMAN, podemos digitar na barra de URL também do navegador:

localhost:8080/CRUD-Back/api/chamados

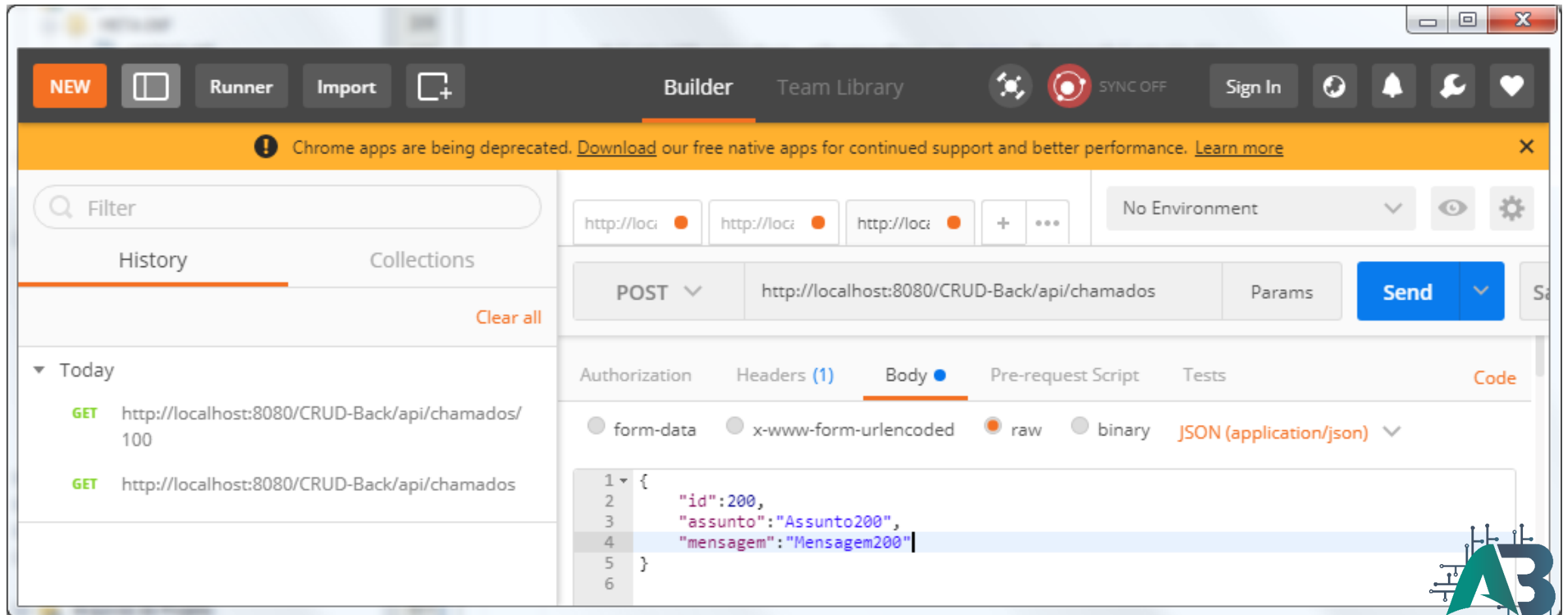
Este é o resultado no POSTMAN, retorna todos os chamados quando utilizamos o método listChamados



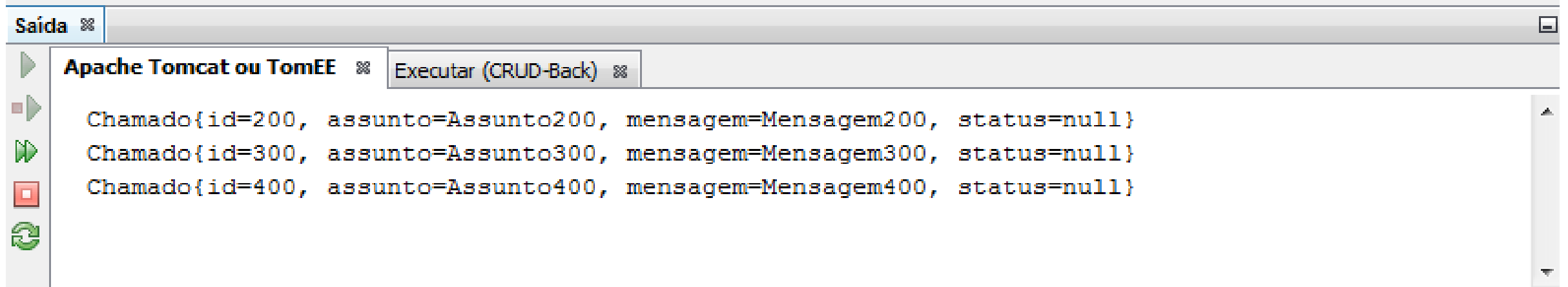
Vamos testar o método getChamado, para isso vamos digitar na URL o caminho **localhost:8080/CRUD-Back/api/chamados/100**



Para testar o método **create**, vamos utilizar o **POSTMAN** para isso digitando na URL o caminho:

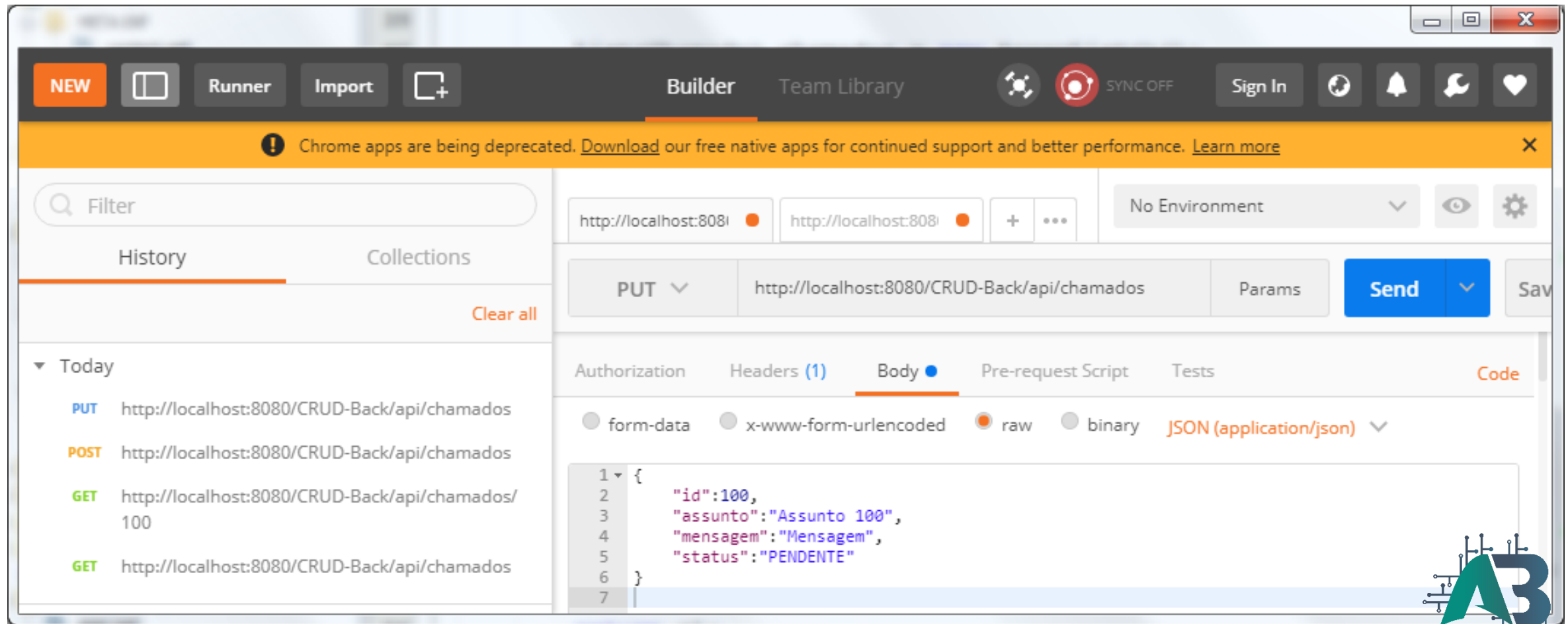


No **console** do IDE Netbeans vai **printar** o chamado que foi criado, perceba que o atributo **status** veio com o valor NULL, vamos alterar isso posteriormente

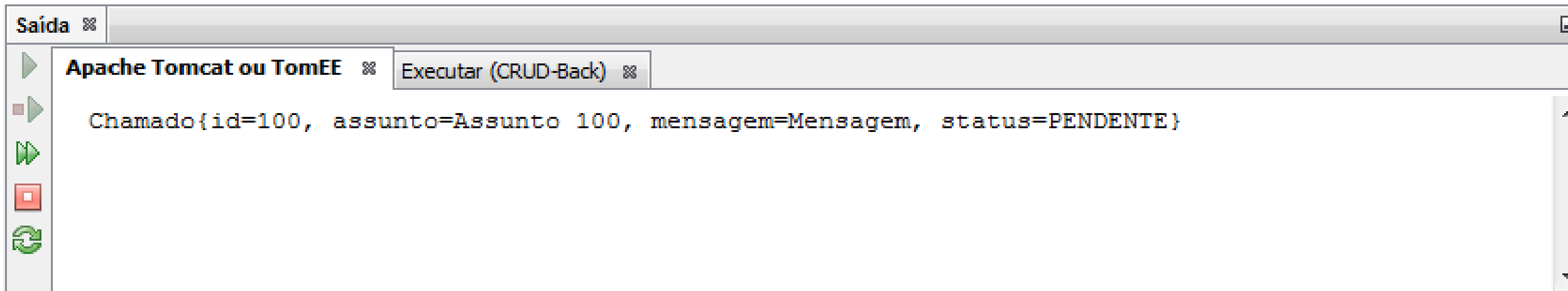


```
Saída
Apache Tomcat ou TomEE
Executar (CRUD-Back)
Chamado{id=200, assunto=Assunto200, mensagem=Mensagem200, status=null}
Chamado{id=300, assunto=Assunto300, mensagem=Mensagem300, status=null}
Chamado{id=400, assunto=Assunto400, mensagem=Mensagem400, status=null}
```

Para testar o método **update**, também utilizaremos o **POSTMAN** digitando na URL o caminho:



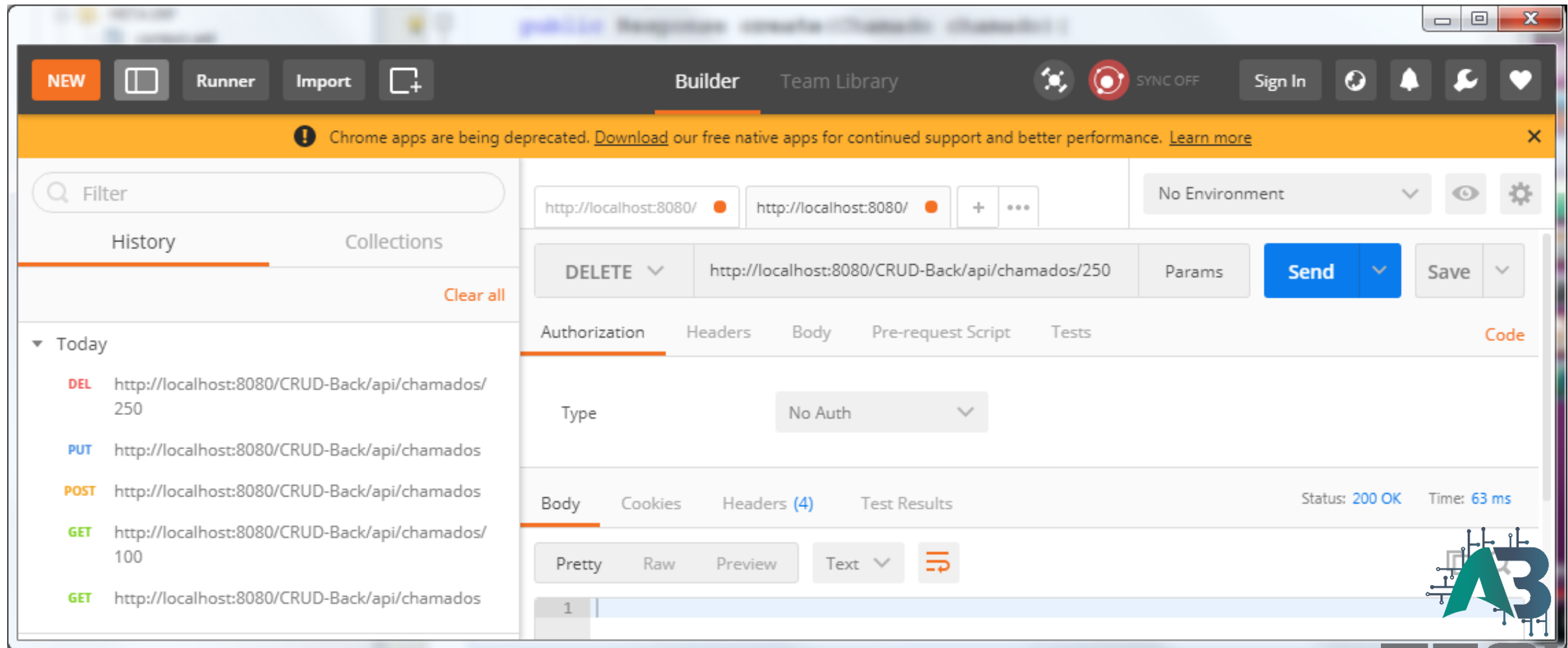
No **console** do IDE Netbeans vai **printar** o chamado que foi atualizado, perceba que o atributo **status** veio com o valor PENDENTE



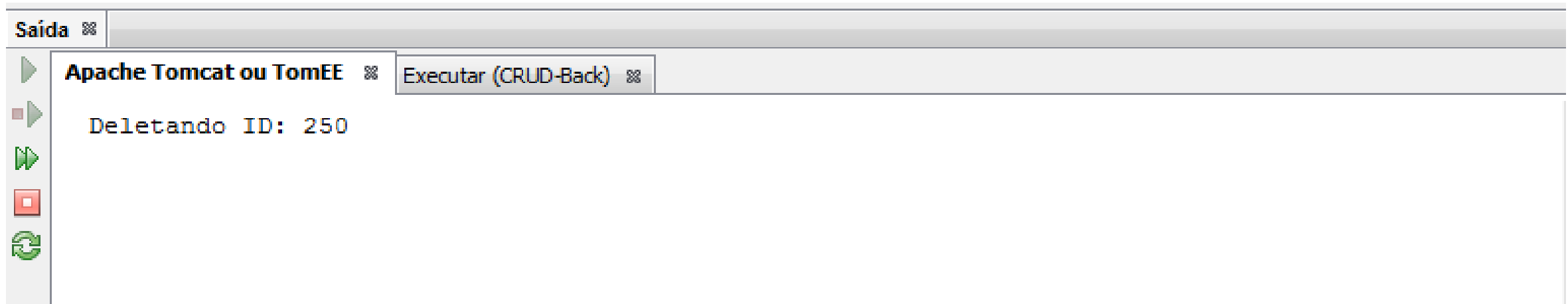
The screenshot shows the NetBeans IDE's console window. The title bar reads 'Saída' with a small icon. Below it, a tab is labeled 'Apache Tomcat ou TomEE' with a small icon, and the active sub-tab is 'Executar (CRUD-Back)' with a small icon. The console area displays the following Java object representation: `Chamado{id=100, assunto=Assunto 100, mensagem=Mensagem, status=PENDENTE}`. On the left side of the console, there is a vertical toolbar with icons for running (a green play button), stepping through code (a green play button with a square), stopping (a red square), and refreshing (a green circular arrow).

```
Chamado{id=100, assunto=Assunto 100, mensagem=Mensagem, status=PENDENTE}
```

Para finalizar vamos testar o método **delete**, também utilizaremos o **POSTMAN** digitando na URL o caminho:



No **console** do IDE Netbeans vai **printar** o id do chamado que foi excluído



Dúvidas?

Professor Anderson Henrique



Para a próxima aula

01 – Construindo um FRONT-End com AngularJS

02 – Integrando nosso BACK-End com o FRONT-End

Professor Anderson Henrique