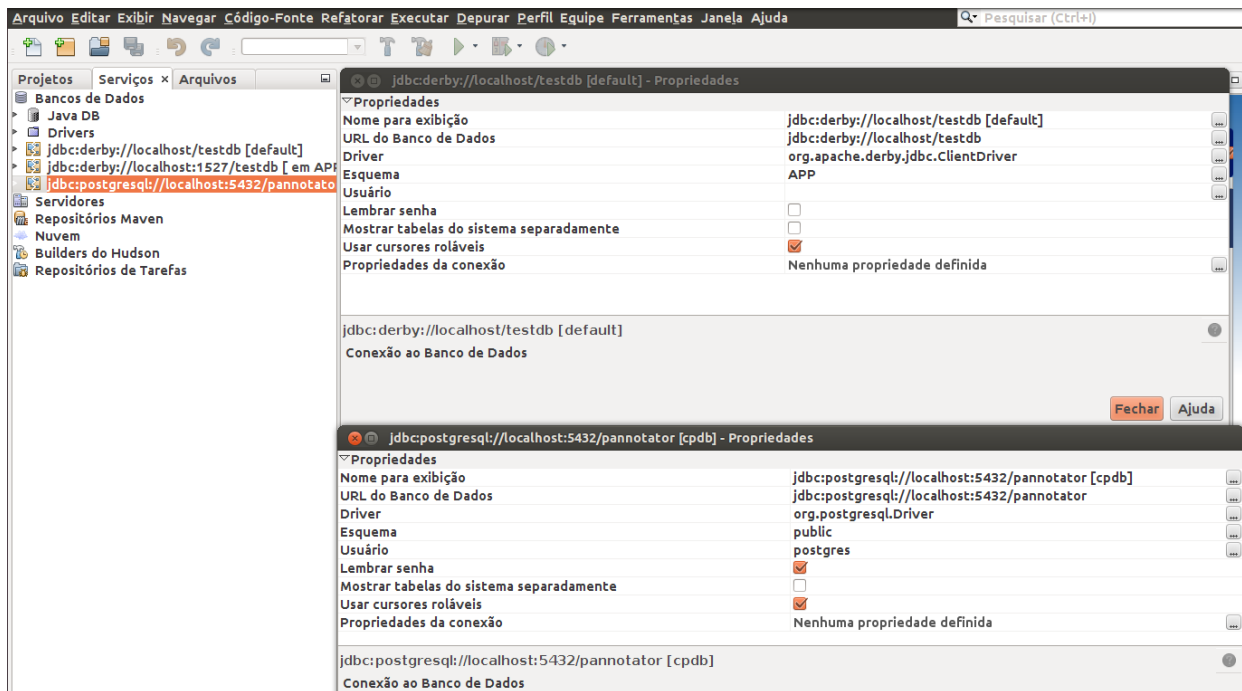


## Sistemas de Bancos de Dados

### Aula 08 – Estabelecendo conexões

O objetivo desta aula é criarmos uma conexão do JDBC ao banco de dados *postgres*. Antes de iniciar, convém salientar que *NetBeans* também possui uma interface de conexão a bancos de dados similar à do programa *pgadmin3*. Caso você tenha instalado o programa *NetBeans* então poderá optar por usá-lo. Se você estiver com dificuldades para se conectar utilizando o *pgadmin3* ou via o seu programa via JDBC uma opção é utilizar o *NetBeans* para tentar encontrar que tipo de erro de conexão você está enfrentando.

- 1) Na aba Serviços do *NetBeans*, opção *Drivers*, você tanto pode utilizar os *drivers* mapeados por padrão quanto se conectar a um novo banco de dados. No exemplo abaixo foram feitas duas conexões ao banco de testes do tutorial e uma conexão a um banco do SGBD PostgreSQL.



Perceba que o número da porta de conexão é essencial, nos exemplos abaixo estão sendo utilizadas as portas padrão de cada SGBD, portas 1527 para o Derby e 5432 para o *postgres*;

- 2) Baixe o arquivo *postgresql-42.2.4.jar* do moodle. Depois abra o terminal de comandos do linux e posicione-se no diretório em que baixou o arquivo *postgresql-42.2.4.jar* para esta aula. Para que não se recorda: use para isso o comando *cd*;
- 3) Copie o drive do *postgres* de nome *postgresql-42.2.4.jar* para a pasta raiz do tutorial do java digitando o comando *cp* abaixo:
  1. *cp postgresql-42.2.4.jar /home/myuser/mydir/JDBCTutorial/*
- 4) Faça uma cópia do arquivo *mysql-sample-properties.xml*, que está na pasta *properties*, com o nome de *postgres-properties.xml*. Esse novo arquivo será utilizado para se conectar ao banco de dados do *postgres*. Terminada a cópia: edite o novo arquivo com os parâmetros corretos, lembre-se do usuário e senha do usuário *postgres* e da porta de conexão padrão. Se você não estiver conseguindo se lembrar, tente criar uma conexão usando, por exemplo, o *NetBeans*. Parâmetros iguais devem ser configurados no arquivo *postgres-properties.xml*.

Sistemas de Bancos de Dados  
Aula 08 – Estabelecendo conexões

- 5) Modifique o programa *JDBCUtilities.java* para incorporar estas linhas que o habilitam a se conectar a um banco de dados *postgres*. Veja que no código existem duas funções com uma estrutura *if* para se conectar ao *derby* e ao *mysql*. Altere ambas as funções acrescentando este código:

```
else if (this.dbms.equals("postgresql")) {  
    currentUrlString = "jdbc:" + this.dbms + "://" + this.serverName + ":" +  
this.portNumber + "/" + this.dbName;  
    conn = DriverManager.getConnection(currentUrlString, connectionProps);  
    this.urlString = currentUrlString + this.dbName;  
    conn.setCatalog(this.dbName);  
}
```

Perceba que em uma função a variável *currentUrlString* está declarada e na outra não. Na função sem declaração é necessário dizer qual é o tipo da variável ou teremos um erro de compilação. Acrescente o tipo *String*, antes da variável, na função na qual a variável não foi declarada.

- 6) Crie duas novas classes a partir da classe *MyQueries.java* (*MyQueries2.java* e *MyQueries3.java*) para realizar as seguintes consultas e exibir os resultados na tela da linha de comando. Para esse fim é necessário tanto mudar o nome do arquivo quanto das referências à classe que está definida dentro do arquivo ou a Java vai acusar erro de compilação. Estas são as consultas que os novos programas devem fazer:
1. Retorne os nomes de todos os clientes que possuem apenas depósitos, bem como a soma de depósitos de cada cliente (Implementar no *MyQueries2.java*);
  2. Retorne os nomes dos clientes que possuem depósitos e empréstimos (ambos) com as respectivas somas (Implementar no *MyQueries3.java*);
- 7) Modifique o arquivo '*comp*' da aula anterior para criar uma variável *postgres* com a configuração adequada à estrutura de diretórios de sua máquina. Use a variável *derby* como exemplo de como deve ficar a variável *postgres* dentro do arquivo *comp*.
1. *postgres=/home/myuser/mydir/JDBCTutorial/postgresql-42.2.4.jar*
  2. Acrescente a variável *\$postgres* no parâmetro *-classpath* (*-cp*) ao acionar o compilador java (denominado *javac*) e também ao acionar a máquina virtual java (denominado *java*). Modifique o script *comp* para este fim.
  3. Lembre-se que se o arquivo *comp* foi copiado agora e ainda não utilizado então ele ainda não tem permissão de execução. Neste caso o comando *chmod 755 comp* vai habilitá-lo para este fim;
- 8) Compile primeiro a classe *JDBCUtilities.java* que foi modificada nas etapas anteriores:  
*./comp JDBCUtilities properties/javadb-sample-properties.xml*
- 9) Compile e execute em seguida as duas classes *MyQueries2* e *MyQueries3*.

Lembre-se que o novo arquivo *postgres-properties.xml* (passo 4) deve ser passado para todos os programas compilados e executados com o programa *comp*.