

Como usar os dados de um arquivo .odb em modo multiusuário

Por Johnny Furusato

Neste tutorial mostrarei como usar de modo compartilhado um arquivo .odb, tornando possível que mais de uma pessoa possa trabalhar com os mesmos dados atualizados ao mesmo tempo.

Um aviso importante: mantenha backups dos seus arquivos .odb antes de iniciar os procedimentos abaixo. Há sério risco de que eles se tornem inacessíveis alterando-se a versão utilizada do hsqldb do seu LibreOffice (1.8.0) e passando a usar do desenvolvedor (2.3.X). Não me responsabilizo por qualquer eventual perda de dados ou prejuízo que possa

ocorrer decorrente deste experimento. Como vimos recentemente na LM-ED14, o Base pode ser gerado como um banco de dados HSQLDB incorporado.

Isso facilita o desenvolvimento de aplicações rápidas, mas voltado para uso em um ambiente monousuário e com grande possibilidade de travamentos e instabilidades conforme o arquivo cresça e seja usado com outras aplicações carregadas na memória.

Como pode ocorrer também a perda de dados nestas situações, isso



prejudicou severamente o desenvolvimento e a importância do Base dentro dos projetos Office (Apache OpenOffice e LibreOffice).

Basta comparar a evolução de ferramentas como o Calc e o Writer em relação ao Base para comprovarmos o pouco desenvolvimento deste último.

Alternativas a esses problemas conhecidos de instabilidade do Base foram propostas tais como o uso do Firebird incorporado e a utilização apenas como uma ferramenta de front-end acessando os dados em um outro SGBD como MySQL, PostgreSQL, etc. Isso pode ser obtido por ODBC, JDBC ou outro driver específico.

Esta última forma tem sido a minha utilização preferida em pequenos projetos, onde alguns formulários feitos no Base são utilizados para manipulação de dados e estes ficam protegidos em um SGBD mais robusto.

Seguindo nessa linha é o que faremos nesse tutorial.

Rodaremos o **HSQLDB** externamente como um servidor de banco de dados

autônomo ao LibreOffice Base.

Trabalharemos em um ambiente com 3 computadores em rede, sendo que um deles terá o papel de servidor onde ficarão centralizados e armazenados os dados. Os outros dois serão os computadores clientes que manipularão essas informações.



Neste tutorial não vou entrar em detalhes sobre a rede local.

Também não é necessário que o computador que fará o papel de servidor, tenha um sistema operacional voltado para esse fim.

Basta que compartilhe uma pasta em rede e sejam atribuídas as permissões de acesso adequadas aos demais.



Note que no meu ambiente de testes, utilizei três computadores com o Windows, por isso as telas podem ser diferentes do seu caso.

No computador servidor, que passaremos a chamar de “**ServidorDB**”, crie uma pasta. No exemplo, criei essa pasta no drive D com o nome de **TESTANDO**. Compartilhe a pasta para os clientes Maria e João.

Na pasta crie o arquivo **.odb** que será usado como padrão e ficará o **hsqldb** que rodará em modo servidor.

Para usar o arquivo em rede é necessário que o arquivo seja separado em uma parte de dados – database, e outra com o front-end – onde ficam os formulários, consultas, etc.

Após a criação deste modo, a pasta poderá ser transportada e usada de modo portátil (armazenada em um pendrive ou hd externo) ou usada compartilhada com o hsqldb ativado em modo servidor ou web.

Há vários modos de construir esse arquivo.

Utilizarei uma biblioteca (library) criada por **Villeroy** a **FreeHSQLDB**.

Lembre-se de fazer uma cópia de backup do seu arquivo .odb com o HSQLDB incorporado antes de iniciar.

Utilizei a estrutura do banco de dados estoque mostrada na LibreOffice Magazine – Edição 14, página 73.

Tenho assim as cinco tabelas no arquivo: Cliente, DetalhePedido, Pedidos, Fornecedor e Produto.

Primeiro, faça o download da última versão do HSQLDB e extraia no seu computador. Por exemplo: C:\hsqldb.

Não é necessário nenhum procedimento adicional para a instalação.

Utilizando um software descompactador zip (por exemplo: WINZIP, 7-ZIP, WINRAR, etc) abra e extraia a **pasta database** do seu arquivo **odb** e renomeie todos os arquivos (script, data, properties) adicionando um prefixo que será o nome do seu arquivo de banco de



dados do hsqldb (prefixo.script, prefixo.data, prefixo.properties).

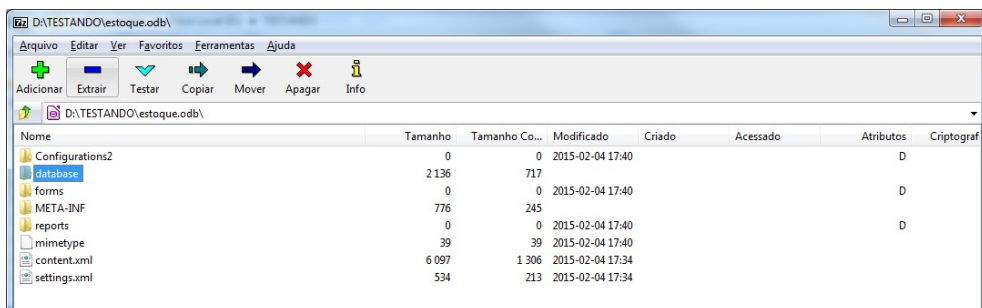
No meu caso, uso o prefixo **dbservidor**.

Note também que não tenho ainda dados gravados, por isso a ausência dos **arquivos data e backup**.

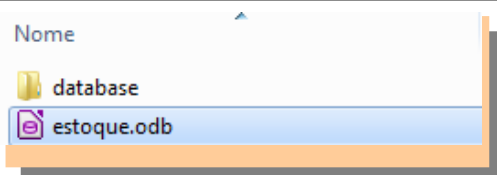
Mas não se preocupe com isso. Quando colocar dados em suas tabelas, os arquivos serão criados automaticamente.

A **pasta database** com os nomes de arquivos com prefixos acrescentados vai se parecer com a imagem abaixo.

Nome	Data de modificaç...	Tipo	Tamanho
dbservidor.properties	04/02/2015 17:40	Arquivo PROPERT...	1 KB
dbservidor.script	04/02/2015 17:40	Arquivo SCRIPT	2 KB



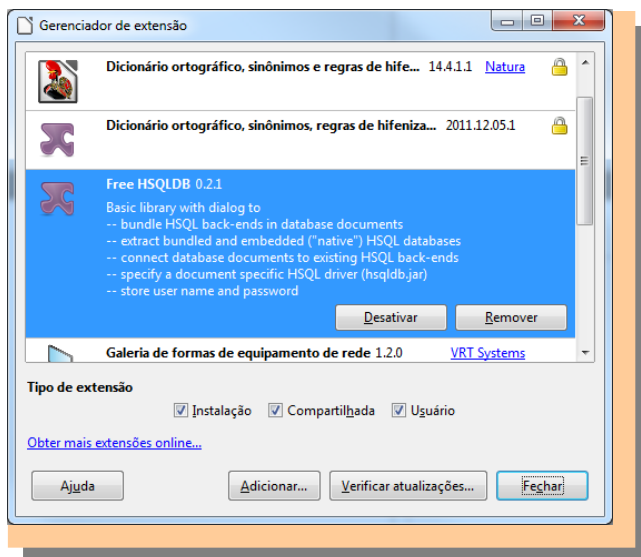
O arquivo estoque.odt original



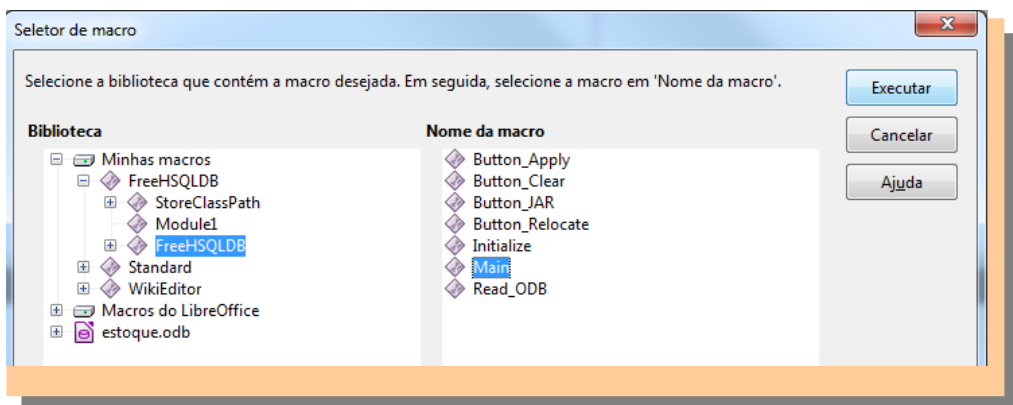
Instale a biblioteca FreeHSQLDB como uma extensão no LibreOffice. Ele está disponível em: <https://forum.openoffice.org/en/forum/download/file.php?id=17008>



Certifique-se de fechar qualquer outro documento do LibreOffice já aberto. Depois de baixar o arquivo clique duplo sobre ele. Será aberta a caixa de dialogo **Gerenciador de extensão** do LibreOffice.



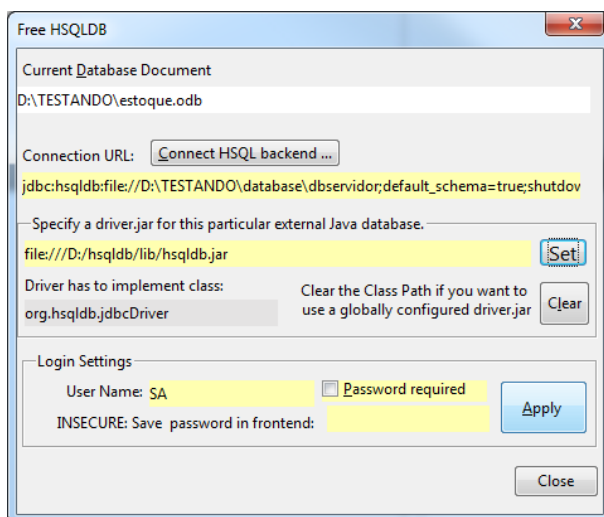
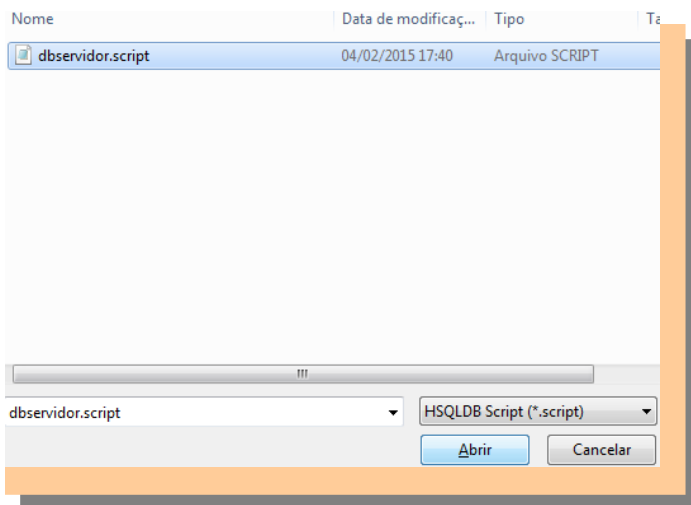
Abra o seu arquivo odb. Com o LibreOffice Base aberto vá em **Ferramentas** > **Macros** > **Organizar macros** > **LibreOffice Basic...** e chame a macro **FreeHSQldb**.





Clique no botão **Set** e aponte para o `hsqldb.jar` que você instalou acima.

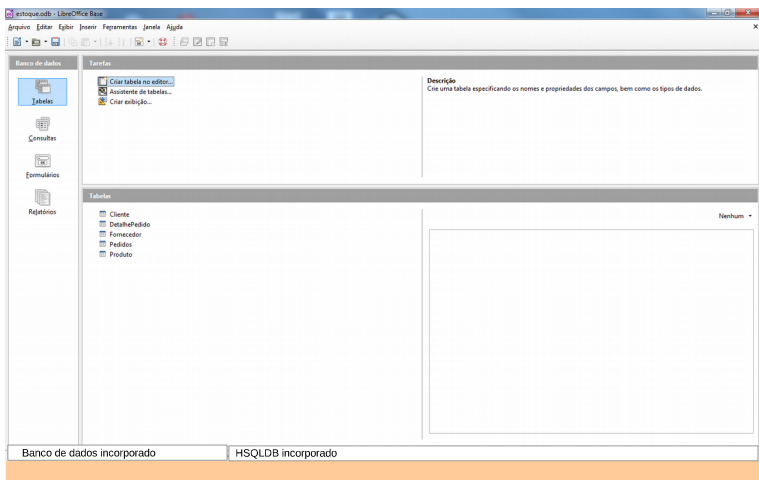
Clique no botão **Connect HSQL backend** para apontar para `prefixo.script` extraído e renomeado acima.



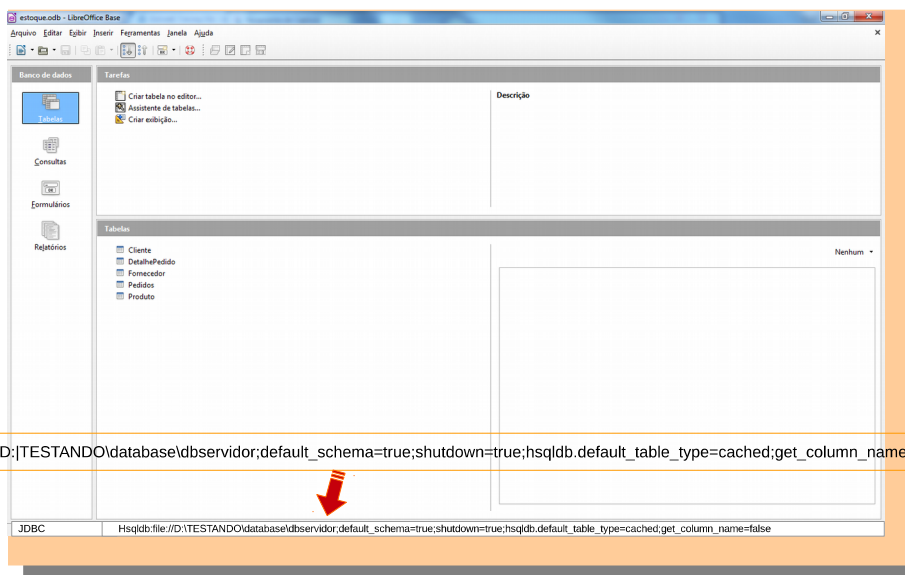


Salve o seu arquivo odb.

Verifique se na barra de status houve alteração da conexão e se o seu banco de dados funciona como o incorporado.



Note como ficou a barra de status com a conexão **JDBC**, com a parte dos dados separados.





Agora você tem na pasta “**TESTANDO**”, um arquivo **odb** e uma pasta **database**.

Repare que a pasta **database** contida dentro do arquivo odb continua existindo, de modo imutável, enquanto a outra externa acumulará todas as alterações de dados das tabelas.

Se você teve sucesso até aqui, agora só falta configurar o HSQLDB em modo servidor e permitir que os clientes acessem o arquivo odb.

Rodando o HSQLDB em modo servidor

No site **hsqldb.org** você encontra os manuais de instalação e utilização do HSQLDB.

Estudando esse material você poderá adaptar o HSQLDB às suas necessidades, pois talvez não estejam contempladas neste tutorial.

A versão do hsqldb.jar que usarei é a 2.3.2.

A pasta **TESTANDO** e o HSQLDB foram instalados no meu drive D.

Para inicializar o HSQLDB você pode rodar no prompt de comando algo do tipo:

```
java -cp hsqldb.jar org.hsqldb.Server -database.0 file:dbservidor;server.dbname.0 dbservidor
```

O comando é comprido, mas para facilitar a manutenção um método interessante é criar dois arquivos .bat na pasta database.

Um deles terá a função de inicializar o servidor HSQLDB e receberá o nome de **up.bat**.

Copie e cole o seguinte texto.

```
@ECHO OFF
```

```
REM este arquivo inicializa o servidor hsqldb e habilita acesso ao BD dbservidor.
```

```
REM adapte as linhas abaixo conforme a sua necessidade
```

```
REM javapath é a localização do seu executável Java.exe
```

```
REM jarpath é a localização do seu arquivo hsqldb.jar
```

continua...



```
set javapath=Java\jre6\bin\java.exe
set jarpath=D:\hsqldb\lib\sqltool.jar
IF EXIST "%PROGRAMFILES(X86)%" GOTO Win64
:Win32
set javapath=%PROGRAMFILES%\%javapath%
GOTO FIM
:Win64
set javapath=%PROGRAMFILES(X86)%\%javapath%
:FIM
REM SHUTDOWN DA VERSÃO 1.8
REM SE USAR ESTA VERSÃO, COMENTE AS LINHAS FINAIS DO ARQUIVO
REM E DESCOMENTE A SEGUINTE
REM ::: Servidor local com user=SA, sem password (APENAS HSQLDB 1.8)
REM "%javapath%" -cp "%jarpath%" org.hsqldb.util.ShutdownServer
"%javapath%" -jar "%jarpath%" --sql "shutdown;" --inlineRC
url=jdbc:hsqldb:hsq://localhost/,User=SA,Password=
```

Abra uma janela de comando e execute o up.bat. Você deve ter uma mensagem, como a que aparece na figura abaixo.

```
cmd Prompt de Comando - up.bat
D:\TESTANDO\database>up.bat
[Server@16672d61: [Thread[main,5,main]]: checkRunning(false) entered
[Server@16672d61: [Thread[main,5,main]]: checkRunning(false) exited
[Server@16672d61: Startup sequence initiated from main() method
[Server@16672d61: Could not load properties from file
[Server@16672d61: Using cli/default properties only
[Server@16672d61: Initiating startup sequence...
[Server@16672d61: Server socket opened successfully in 55 ms.
[Server@16672d61: Database [index=0, id=0, db=file:dbserverid, alias=1] opened su
cessfully in 399 ms.
[Server@16672d61: Startup sequence completed in 464 ms.
[Server@16672d61: 2015-02-05 16:22:31.271 HSQLDB server 2.3.2 is online on port
9001
[Server@16672d61: To close normally, connect and execute SHUTDOWN SQL
[Server@16672d61: From command line, use [Ctrl]+[C] to abort abruptly
```



```
set javapath=Java\jre6\bin\java.exe
set jarpath=D:\hsqldb\lib\hsqldb.jar
IF EXIST "%PROGRAMFILES(X86)%" GOTO Win64
:Win32
set javapath=%PROGRAMFILES%\%javapath%
GOTO FIM
:Win64
set javapath=%PROGRAMFILES(X86)%\%javapath%
:FIM
"%javapath%" -cp "%jarpath%" org.hsqldb.Server -database.0
file:dbserveridor;server.dbname.0
dbserveridor;hsqldb.default_table_type=cached;shutdown=true
```

O outro terá a função de desligar corretamente o servidor e receberá o nome de **shutdown.bat**.

Copie e cole o seguinte texto.

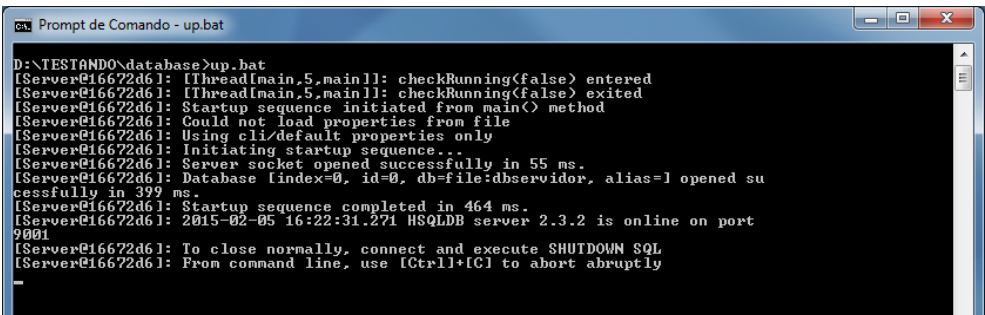
```
@ECHO OFF
REM este arquivo faz o shutdown do servidor hsqldb e encerra acesso ao BD dbserveridor.
REM adapte as linhas abaixo conforme a sua necessidade
REM javapath é a localização do seu executável Java.exe
REM jarpath é a localização do seu arquivo hsqldb.jar
REM hsqldb é a versão do hsqldb
REM a versão 2, utiliza sqltool para fazer shutdown.
set hsqldb=2
```

continua...



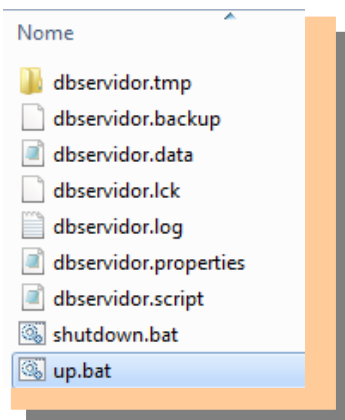
```
set javapath=Java\jre6\bin\java.exe
set jarpath=D:\hsqldb\lib\sqltool.jar
IF EXIST "%PROGRAMFILES(X86)%" GOTO Win64
:Win32
set javapath=%PROGRAMFILES%\%javapath%
GOTO FIM
:Win64
set javapath=%PROGRAMFILES(X86)%\%javapath%
:FIM
REM SHUTDOWN DA VERSÃO 1.8
REM SE USAR ESTA VERSÃO, COMENTE AS LINHAS FINAIS DO ARQUIVO
REM E DESCOMENTE A SEGUINTE
REM ::: Servidor local com user=SA, sem password (APENAS HSQLDB 1.8)
REM "%javapath%" -cp "%jarpath%" org.hsqldb.util.ShutdownServer
"%javapath%" -jar "%jarpath%" --sql "shutdown;" --inlineRC
url=jdbc:hsqldb:hsq://localhost/,User=SA,Password=
```

Abra uma janela de comando e execute o up.bat. Você deve ter uma mensagem, como a que aparece na figura abaixo.





Com o servidor hsqldb em execução são criados outros arquivos e uma pasta tmp.



Abra uma segunda janela de prompt de comando e execute o shutdown.bat. Você verá que o servidor parou corretamente e os arquivos temporários foram apagados.

```
ca: Prompt de Comando
D:\TESTANDO\database>up.bat
[Server@16672d6]: [Thread[main,5,main]]: checkRunning(false) entered
[Server@16672d6]: [Thread[main,5,main]]: checkRunning(false) exited
[Server@16672d6]: Startup sequence initiated from main() method
[Server@16672d6]: Could not load properties from file
[Server@16672d6]: Using cli/default properties only
[Server@16672d6]: Initiating startup sequence...
[Server@16672d6]: Server socket opened successfully in 55 ms.
[Server@16672d6]: Database [index=0, id=0, db=file:dbserveridor, alias=1] opened successfully in 399 ms.
[Server@16672d6]: Startup sequence completed in 464 ms.
[Server@16672d6]: 2015-02-05 16:22:31.271 HSQLDB server 2.3.2 is online on port 9001
[Server@16672d6]: To close normally, connect and execute SHUTDOWN SQL
[Server@16672d6]: From command line, use [Ctrl]+[C] to abort abruptly
[Server@16672d6]: Initiating shutdown sequence...
[Server@16672d6]: Shutdown sequence completed in 100 ms.
[Server@16672d6]: 2015-02-05 16:35:34.808 SHUTDOWN : System.exit() is called next
D:\TESTANDO\database>
```

Configurando o acesso pelos clientes

A configuração seguinte exige que você já tenha o hsqldb rodando em modo servidor no computador **ServidorDB** e que cada computador cliente use o mesmo arquivo **hsqldb.jar**.



O melhor é que cada cliente use o mesmo arquivo **estoque.odb**, pois terão acesso aos mesmos formulários, consultas, relatórios, etc.

Note que sempre que o arquivo **estoque.odb** for alterado no **ServidorDB** ele deverá ser novamente copiado e configurado nos clientes.

As alterações no front-end de um cliente não afeta a visão dos outros.

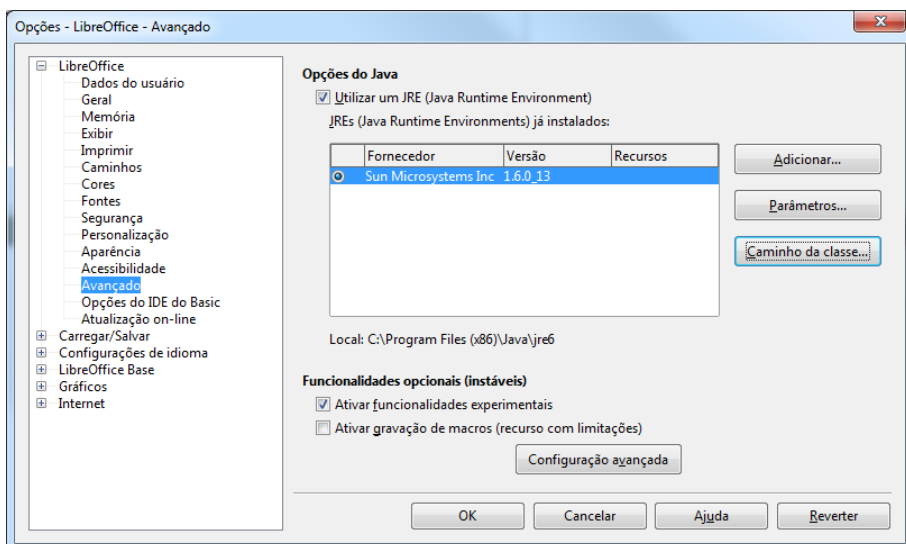
Assim, se Maria resolver criar um novo formulário no seu **estoque.odb** instalado no seu computador apenas ela terá acesso.

No computador da Maria e do João copie apenas o arquivo **estoque.odb** e cole em uma pasta qualquer.

Abra o arquivo e antes de qualquer coisa, habilite o **hsqldb.jar** e altere as propriedades de conexão.

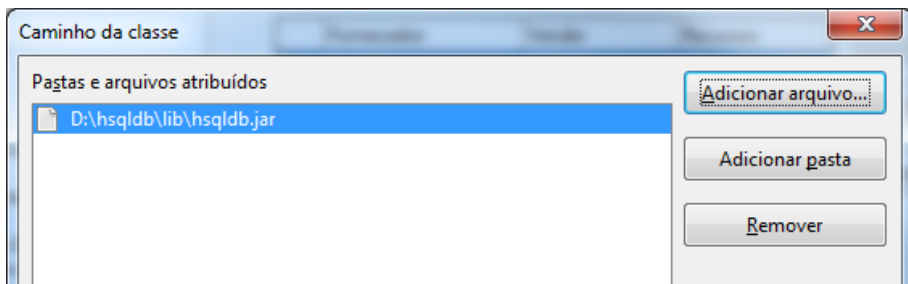
Para habilitar o **hsqldb.jar**, no menu do LibreOffice vá em:

- **Ferramentas > Opções > LibreOffice > Avançado.**
- Clique em **Caminho da Classe...**



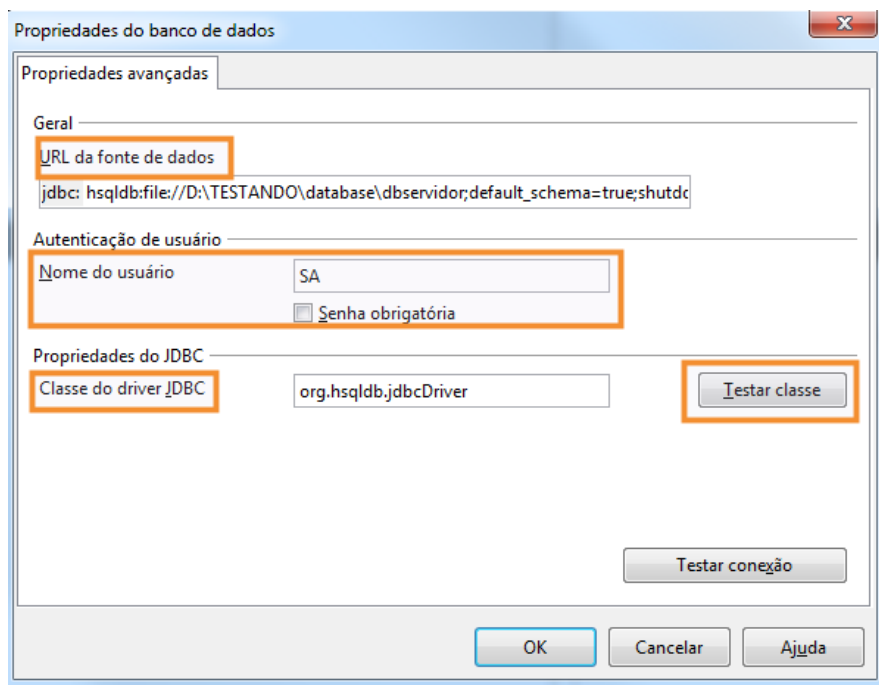


Adicione o seu arquivo **hsqldb.jar**



Em seguida, altere a conexão JDBC.

No menu do LibreOffice Base vá em **Editar > Banco de dados > Propriedades**.





- Em **nome do usuário** deixei o padrão da instalação do hsqldb - **SA** e **sem senha**.
Altere se for necessário.

- Altere a URL da fonte de dados para:

```
hsqldb:hsql://endereçoIPdoServidorDB;default_schema;get_column_name=false
```

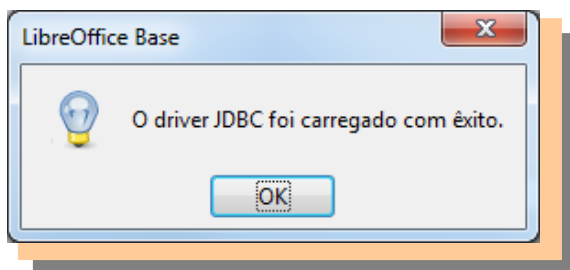
Como o endereço do meu servidor **ServidorDB** é 10.68.100.1, fica:

```
hsqldb:hsql://10.68.100.1.;default_schema;get_column_name=false
```

- Em Classe do driver JDBC digite **org.hsqldb.jdbcDriver**

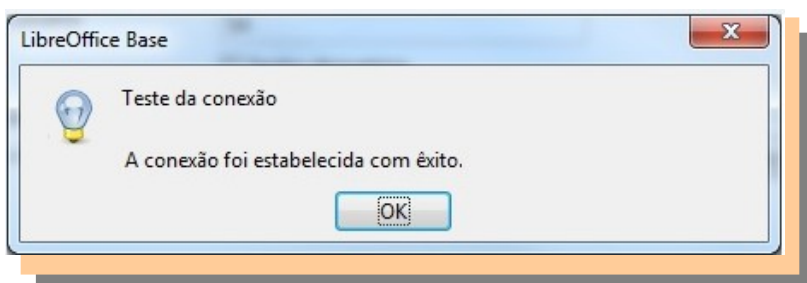
- Pressione o botão **Testar Classe**

Se tudo correr bem, você verá a seguinte mensagem.



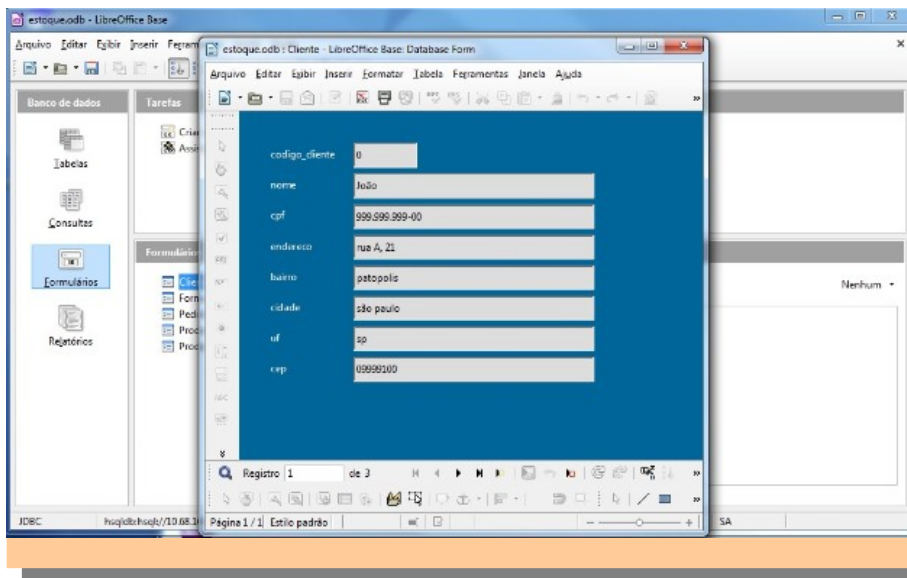
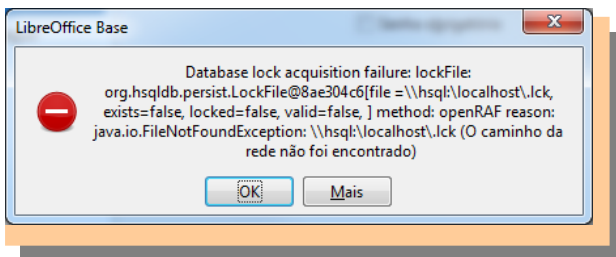
- Pressione o botão **Testar Conexão**

Se tudo correr bem, você verá a tela de mensagem abaixo.





Caso o seu servidor não esteja rodando ou configurado corretamente, você poderá ver algo como mostra a figura abaixo.



Aguarde um pouco e você verá as tabelas, formulários, etc.

- Salve o seu documento.



Para testar crie um formulário e use ao mesmo tempo nos clientes Maria e João.

Atualize a visão do Formulário, clicando no botão de navegação correspondente para acompanhar as mudanças efetuadas.

Bom trabalho! Sucesso!



Johnny Furusato - Advogado e Gerente de TI com ampla experiência na área de programação, segurança e tecnologia da informação. Sempre que pode compartilha o pouco que sabe. Acredita que podemos transformar o Mundo com educação de qualidade e a vivência das virtudes.