

TESTE DE SELEÇÃO: C.R.U.D. MySQL e JDBC

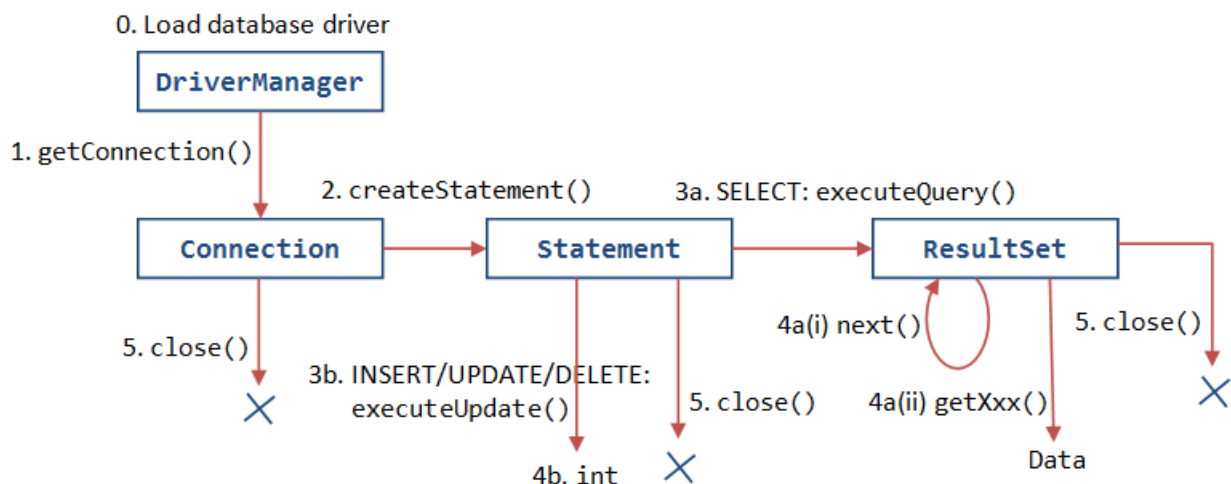
Recentemente, recebi uma informação de uma vaga de emprego onde o candidato deveria realizar um CRUD simples, somente com acesso ao banco de dados, de escolha do próprio, sem acesso a internet. O candidato teria a sua disposição a documentação do Banco de Dados Utilizado MySQL ou Postgresql e do Java utilizado. Nesta vaga, não houve aprovações. Alguns comentaram, negativamente e positivamente o fato. Como Desenvolvedor Candidato, você conseguiria se sair bem neste exame?

O JDBC, **Java Database Connectivity**, reúne classes e interfaces escritos em Java para se conectar através de um driver, programa também em Java para a conexão do banco de dados escolhido à linguagem.

Agora com o driver específico, e a IDE favorita precisamos, importar o driver conector ao programa e testar conexão, acessar o Banco de Dados escolhido e utilizar as classes e realizar as operações pedidas.

Num momento de teste, estas informações seriam bem mais complicada sem o acesso aos documentos. Os bancos de dados mais utilizados, possuem exemplos e detalhes escritos que facilitam a nossa vida de desenvolvedor. A documentação da linguagem também favorece a feitura do projeto.

Acredito que o candidato, ao realizar tal exame, já esteja familiarizado com desenvolvimento de APIs com o Spring JPA ou JBDC e tudo mais, onde a feitura do CRUD é rápida e o mesmo possui funções pré-definidas e queries personalizáveis. No caso do exame, devemos nos atentar para criar os arquivos necessários como o ConnectionFactory, o DAO , a Entity, e o Main.



Acima, a figura nos mostra o caminho de como utilizar as classes, interfaces específicas do Java para fazer a ligação entre o banco de dados e a própria linguagem. Criei Diretórios específicos para a conexão, para as operações CRUD e para a Entidade utilizada, finalizando as operações no Main. Arquivos utilizados estarão dispostos no meu GitHub.

O arquivo ConnectionFactory foi criado com o objetivo de realizar a parte 1 desta figura. Ele faz a criação, verifica e pega os dados se usuário, senha, banco de dados, porta utilizada para tal.

As operações que na figura foram realizadas com a classe Statement, no meu CRUD foram realizadas pela classe análoga prepareStatement, com objetivo de evitar problemas de SQL injection. No arquivo ContatoDAO, as operações do CRUD foram separadas em funções específicas para que sejam chamadas no arquivo principal, Main, utilizando a Entidade, modelo de classe referente à tabela Contato de uma agenda.

Realizando esta separação, comum nos ensinamentos deste assunto nas salas de aula, o candidato poderá realizar a tarefa com mais calma, estando atento a cada parte da tarefa sem comprometimento das outras. Acredito também que o candidato mais habituado a lidar com banco de dados, poderá realizar o exame com um pouco mais de calma também.

Bom, aproveito para finalizar desejando bons estudos e boas práticas para todos. Forte Abraço.