



Uso e funcionamento de um driver JDBC



**Certified
Developer**
The Ultimate Tech Degree

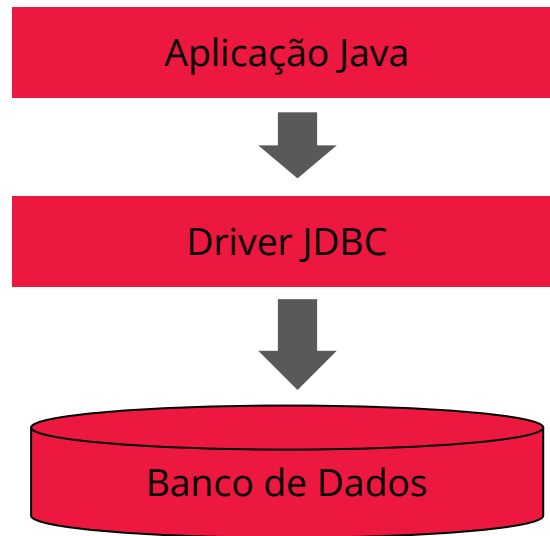
DigitalHouse >
Coding School



Drivers

Embora qualquer um possa implementar as interfaces que o JDBC fornece, isso pode ser um trabalho árduo e extenso, pois exige um grande conhecimento sobre os motores (engines) de banco de dados.

Cada implementação para um motor (engine) de banco de dados consiste em um pacote de classes, normalmente em arquivos .jar, que podem ser baixados dos sites oficiais do motor (engine) e são conhecidos como driver JDBC. Em outras palavras, temos um driver JDBC para Oracle, outros para MySQL e etc.





Drivers

Uso e funcionamento

Cada driver é um pacote de classes que devemos incorporá-lo como uma biblioteca em nosso programa Java. Portanto, as classes que implementam as interfaces do driver precisam estar disponíveis no CLASSPATH. Embora seja possível fazer isso manualmente, normalmente fazemos isso por meio da IDE.





Drivers

Uso e funcionamento

Um vez que o driver esteja disponível no CLASSPATH, podemos começar a fazer uso dele. De uma forma bem simples, cada vez em que quisermos interagir com o banco de dados, precisamos realizar algumas etapas. Vamos conhecê-las a seguir:





Drivers

Etapas





Vamos ver, no código, como essas etapas são implementadas. Suponha o uso de uma classe, com um método principal contendo o seguinte código:

```
Connection c = null;
try {
    Class.forName("org.h2.Driver");
} catch (ClassNotFoundException e) {
    e.printStackTrace();
    System.exit(0);
}
```



```
try {  
    String url =  
        "jdbc:h2:tcp://localhost//D:/banco-de-dados/exemplo";  
    c = DriverManager.getConnection(url, "sa", "sa");  
    c.setAutoCommit(false);  
} catch (SQLException e) {  
    e.printStackTrace();  
    System.exit(0);  
}
```



```
try {  
    Statement s = c.createStatement();  
    s.execute("SUA INSTRUÇÃO SQL");  
} catch (Exception e) {  
    e.printStackTrace();  
} finally {  
    try {  
        c.close();  
    } catch (SQLException e) {  
        e.printStackTrace();  
    }  
}
```


DigitalHouse>
Coding School