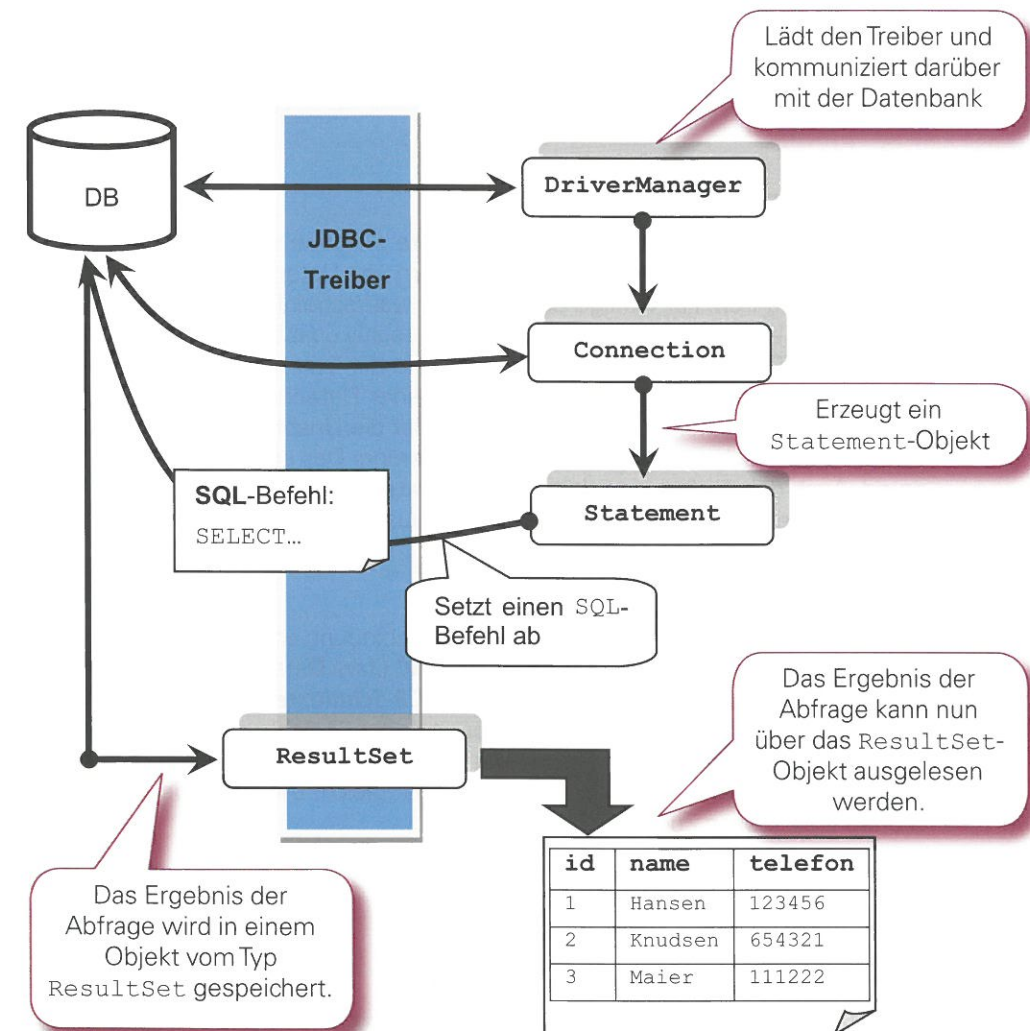


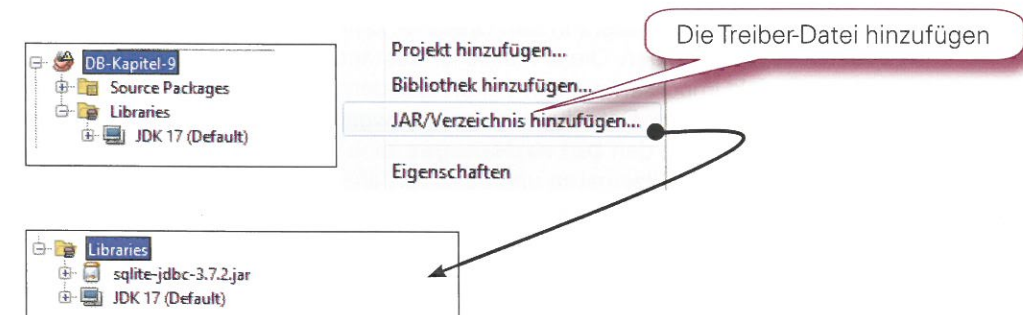
Die folgende Abbildung soll den Zusammenhang noch einmal darstellen:



9.1.3 Zugriff auf eine SQLite-Datenbank

Im Folgenden wird der Zugriff auf eine SQLite-Datenbank vorgestellt. Das Prinzip ist aber übertragbar auf andere relationale Datenbanken wie beispielsweise MySQL/Maria- oder Oracle-Datenbanken. Dabei muss das Paket `java.sql` importiert werden. In diesem Paket sind alle relevanten Klassen, um auf eine Datenbank zuzugreifen.

In den folgenden Beispielen wird die Entwicklungsumgebung **NetBeans** verwendet. Das Prinzip ist aber auf andere Umgebungen (wie **Eclipse**) übertragbar. Nach dem Download des gewünschten Treibers (beispielsweise `sqlite-jdbc-XXX.jar`) wird die Datei vom Typ Java-Archive in das Projekt integriert:



Nach dem erfolgreichen Hinzufügen des Treibers kann die Verbindung mit der Klasse **Class** erzeugt werden:

```
String datenbank = "jdbc:sqlite:/Pfad/Datenbank";
```

Den Pfad und die Datenbankdatei angeben.

```
Class.forName("org.sqlite.JDBC");
```

Den Treiber laden.

Die Verbindung herstellen

```
Connection verbindung;
```

```
verbindung = DriverManager.getConnection(datenbank, "", "");
```

Tipp:

ID	Name
Filter	Filter
1	Maier
2	Knudsen
3	Kaiser
4	Franzen
5	Knobloch
6	Laufer

Die Datenbank SQLite ist eine kostenfreie und portable Datenbank, die eine komplette Datenbanklogik und die Daten selbst in einer Datei kapselt. Damit ist die Weitergabe von Java-Programmen mit einer eigenen Datenbank möglich. Für kleine Projekte mit relativ wenig Datenvolumen ist es eine hervorragende Alternative zu den großen Datenbanken wie Oracle oder auch MySQL / MariaDB. Mit kostenfreien Tools wie dem „DB-Browser für SQLite“ können die Datenbanken einfach administriert werden.

Die zugrunde liegende Datenbank *Kunden.sqlite* liegt für das folgende Beispiel in dem Ordner `C:\temp`. Sie verfügt über eine Beispieltabelle *Kunden* mit den Attributen *ID* (Typ *Zahl*) und *Name* (Typ *VARCHAR*):

```
package db_zugriff_java;
import java.sql.*;
```

```
public class DBZugriff {
    public static void main(String[] args) {
        try {
```

Den Verbindungsstring mit Treiberangabe und der Datenquelle festlegen.

```
String datenbank = "jdbc:sqlite:/c:/temp/kunden.sqlite";
```

Den Treiber laden.

```
Class.forName("org.sqlite.JDBC");
```

Ein Verbindungsobjekt mithilfe der statischen Methode `getConnection` anfordern.

```
Connection verbindung =
    DriverManager.getConnection(datenbank, "", "");
```

Über das Verbindungsobjekt wird ein Objekt für den SQL-Befehl erstellt.

```
Statement sqlBefehl = verbindung.createStatement();
```