

```
import java.nio.file.*;
import console.*;

/*
 * Beispielprogramm für die Klassen DatabaseConnector und QueryResult
 * (Name, Datum)
 */
public class DBBeispiel
{
    private DatabaseConnector dbCon;
    private String dateiname = "terra.db";

    public DBBeispiel()    // dbCon wird in main() erzeugt
    { }

    /*
     * Öffnet eine Verbindung mit der SQLite-Datei
     * und ruft die Methoden für die Datenbankabfragen auf.
     */
    public void main()
    {
        // Verbindung aufbauen (bei Fehler: Abbruch)
        if (!Files.exists(Paths.get(dateiname)))
        {
            Console.println("Datei " + dateiname + " existiert nicht.");
            return;
        }
        dbCon = new DatabaseConnector("", 0, dateiname, "", "");
        String fehler = dbCon.getErrorMessage();
        if (fehler != null)
        {
            Console.println(fehler);
            return;
        }

        // Abfragen an die Datenbank
        Console.clear();
        europa();
        Console.println();
        englisch();
    }

    /*
     * Ermittelt die Länder Europas und gibt sie auf der Konsole aus
     */
    public void europa()
    {
        QueryResult result;

        // Sucht Namen und Fläche der Kontinente
        dbCon.executeStatement(
            "SELECT Land.Name, Land.Einwohner, Land.Flaeche, Ort.Name "
            + "FROM Land, Kontinent, Ort "
            + "WHERE Land.KNR = Kontinent.KNR AND Land.HauptONR = Ort.ONR "
            + "AND Kontinent.Name = 'Europa';");
    }
}
```

```
result = dbCon.getCurrentQueryResult();
Console.println("Länder Europas");

// Fehler ausschließen
if (dbCon.getErrorMessage() != null)
{
    Console.println(dbCon.getErrorMessage());
    return;
}
if (result == null)
{
    Console.println("Kein Abfrageergebnis");
    return;
}

int zeilen          = result.getRowCount();
int spalten         = result.getColumnCount();
String[] spaltenNamen = result.getColumnNames();
String[] datentypen  = result.getColumnTypes();
String[][] daten     = result.getData();
int s, z;

// Spaltennamen ausgeben
for (s = 0; s < spalten; s++)
{
    Console.print(spaltenNamen[s] + " ");
}
Console.println();
// Datentypen ausgeben
for (s = 0; s < spalten; s++)
{
    Console.print(datentypen[s] + " ");
}
Console.println();
// Daten ausgeben
for (z = 0; z < zeilen; z++)
{
    for (s = 0; s < spalten; s++)
    {
        Console.print(daten[z][s] + " ");
    }
    Console.println();
}
}

/*
 * Ermittelt die Anzahl Menschen weltweit, die Englisch sprechen
 */
public void englisch()
{
    dbCon.executeStatement(
        "SELECT Land.Einwohner, gesprochen.Anteil "
        + "FROM Sprache, gesprochen, Land "
        + "WHERE Sprache.SNR = gesprochen.SNR "
        + "AND gesprochen.LNR = Land.LNR "
        + "AND Sprache.Name = 'Englisch' "
        + "AND gesprochen.Anteil IS NOT NULL;");
}
```

```
QueryResult result = dbCon.getCurrentQueryResult();

// Fehler ausschließen
if (dbCon.getErrorMessage() != null)
{
    Console.println(dbCon.getErrorMessage());
    return;
}
if (result == null)
{
    Console.println("Kein Abfrageergebnis");
    return;
}

int zeilen      = result.getRowCount();
String[][] daten = result.getData();
int z, anzahl;

// Durchschnittliche Einwohnerzahl ausrechnen
anzahl = 0;
for (z = 0; z < zeilen; z++)
{
    anzahl += Double.parseDouble(daten[z][0]) *
               Integer.parseInt(daten[z][1]) / 100;
}
Console.println(anzahl + " Mio. Menschen sprechen Englisch
als Muttersprache.");
}
```