

LE04 – Einführung Bindings

Mit Binding versteht man die Übergabe von Variablenwerten in SQL-Statements. Damit können User-Eingaben in SQL-Statements übernommen werden.

Erstes einfaches Beispiel: Die Usereingabe des Genres, listet alle dazugehörenden Songs aus. Als Genre sind gültig: Latin, Metal, Rock, Jazz, Bossa Nova, Blues, etc.

binding1.py

```
1  import os
2  import sqlite3
3
4  # Datenbank öffnen und verbinden
5  conn = sqlite3.connect(os.path.join('C:/Users/tom/WORKLOCAL/PythonProj/UE04/', 'chinook.db'))
6
7  # Cursor kreieren. Dieser wird im Zusammenhang mit einem Query verwendet
8  cur = conn.cursor()
9
10 bind_value = input("Genre: ")
11 print(bind_value)
12
13 query = "SELECT * FROM songs WHERE genre = ?;"
14 # Query mit Cursor ausführen
15 cur.execute(query, (bind_value,)) # (1)
16 # alternative Schreibweise für single element tuple: cur.execute(query, [bind_value])
17
18
19 # Bearbeite jede Zeile des Result-Sets
20 for row in cur.fetchall():
21     print(row)
22
23 # Schliesse Cursor und Verbindung
24 cur.close()
25 conn.close()
```

1. Als Eingabe wird ein Single-Element-Tupel erwartet! Python requires a comma to create a single element tuple
! Alternativ: `cur.execute(query, [bind_value])`

zum Erinnern!

Vom Befehl `cursor.execute()` werden TUPEL erwartet! Ansonsten funktioniert das Binding nicht. Wenn man also nur einen Wert als Binding hat, muss man ein `Single Element Tuple` kreieren!

Single Element Tupel schreibt man immer mit Komma!

Ohne Komma erhält man einen String.

BEACHTEN:

```
>>> type(('eins',))
<class 'tuple'>
>>> type('eins')
<class 'str'>
>>>
```