

4.1.5 Arbeiten mit Musik und Soundeffekten

4.2 Rückkehr eines Klassikers: Pong mit Pygame

4.2.1 Noch ein wenig Objektorientierung

4.2.2 Sound kommt hinzu

4.2.3 Code für pong.py

5 GUI-Programmierung mit Tkinter

5.1 Tkinter einbinden

5.2 Die Bausteine einer GUI

5.2.1 Das Basisfenster Tk()

5.2.2 Die wichtigsten GUI-Elemente

5.3 Eine Oberfläche für den Vokabeltrainer

5.3.1 Das Layout des Vokabeltrainers

5.3.2 Die Callback-Methoden implementieren

5.4 Noch mehr Tkinter und ein Taschenrechner für den Raspberry Pi

5.4.1 Die Layout-Manager

5.4.2 Eine Klasse für Taschenrechner

6 Willkommen in der Cloud: Web Apps mit dem Raspberry Pi

6.1 Erste Schritte mit Flask

6.1.1 Routing

6.1.2 HTML und HTML-Templates

6.1.3 GET und POST

6.1.4 Ein Crashkurs in SQL und
Datenbanken

6.2 Eine To-do-App für den Raspberry Pi

6.2.1 Bestandteile der Web App

6.2.2 Das To-do-Listen-HTML-Template

6.2.3 Der Python-Code für die To-do-Liste

6.3 Eine Web App automatisch starten

6.3.1 Python-Programme starten ohne IDE

6.3.2 Programme beim Systemstart automatisch starten

7 Ein Webradio mit dem Raspberry Pi

7.1 Vorbereitungen für den Raspberry Pi

7.2 Den Music Player installieren

7.3 Die Stationsliste erstellen

7.4 Teil 1: Ein Webradio mit Flask

7.4.1 Das Webradio-HTML-Template

7.4.2 Der Python-Code für das Radio-Webinterface