Inhalt

1 Einführung 1

2 Den Raspberry Pi und Python zum Laufen bringen 5

2.1 Was brauchen wir alles? 5

2.2 Ein Blick auf die Hardware 6

2.3 Raspbian – das Mini-Linux für deinen Raspberry Pi 8

2.4 Die SD-Karte vorbereiten 10

2.4.1 NOOBS oder Raspbian? 10

2.4.2 Eine fertige SD-Karte bestellen 11

2.4.3 Ein Betriebssystem-Image auf die SD-Karte schreiben 11

2.5 Python 3 installieren 12

2.6 Der Python Package Index 14

2.7 Die Entwicklungsumgebung 15

2.7.1 IDLE – die Standardumgebung 16

2.7.2 Geany – der Allrounder 18

2.7.3 Python lernen mit Raspberry Pi und Thonny 19

2.8 Den Raspberry Pi fernsteuern 21

2.8.1 SSH-Verbindung im Textmodus herstellen 22

2.8.2 Remotedesktop-Verbindung für Windows 25

2.8.3 X11-Forwarding für Linux und Windows 26

3 Einstieg in die Python-Programmierung 33

3.1 Interpreter und Compiler 33

3.2 Die interaktive Shell 34

3.2.1 Es geht immer los mit „Hallo Welt!“ 35

3.2.2 Die Shell als Taschenrechner nutzen 35

3.2.3 Integer- und Fließkommazahlen 36

3.2.4 Werte in Variablen abspeichern ... 37
3.2.5 Strings und Typumwandlungen ... 38
3.2.6 Wahrheiten und Unwahrheiten ... 39
3.3 Ein Zahlenratespiel – das erste richtige Programm ... 40
3.4 Kontroll- und Datenstrukturen in Python ... 49
3.4.1 Daten kommen oft als Listen daher ... 49
3.4.2 Eine besondere Listenform ... 51
3.5 Wörterbücher in Aktion – ein Vokabeltrainer ... 54
3.5.1 Die Textdatei in Thonny anlegen ... 55
3.5.2 Der Vokabeltrainer – Schritt für Schritt erklärt ... 57
3.5.3 Der Vokabeltrainer wird mehrsprachig ... 60
3.6 Python bei der Arbeit zusehen – der Debugger ... 64
3.6.1 Ein kurzer Ausflug in die Theorie ... 64
3.6.2 Der Debugger in Aktion ... 68
3.6.3 Rekursive Funktionen und noch mehr Debugger-Aktion ... 71
3.7 Mit Objekten zum Vokabellernassistenten ... 73

4 Grafik und Sound mit Pygame ... 81
4.1 Ein Überblick der Möglichkeiten von Pygame ... 82
4.1.1 Bring Bewegung ins Spiel: eine Laufschrift ... 84
4.1.2 Auf Ereignisse reagieren ... 86
4.1.3 Mit pygame.draw geometrische Formen zeichnen ... 88
4.1.4 Eine Analoguhr mit Pygame ... 92
4.1.5 Arbeiten mit Musik und Soundeffekten ... 96
4.2 Rückkehr eines Klassikers: Pong mit Pygame ... 99
4.2.1 Noch ein wenig Objektorientierung ... 100
4.2.2 Sound kommt hinzu ... 105
4.2.3 Code für pong.py ... 106

5 GUI-Programmierung mit Tkinter ... 111
5.1 Tkinter einbinden ... 112
5.2 Die Bausteine einer GUI ... 113
5.2.1 Das Basisfenster Tk() ... 113
5.2.2 Die wichtigsten GUI-Elemente ... 114

5.3 Eine Oberfläche für den Vokabeltrainer ... 118
5.3.1 Das Layout des Vokabeltrainers ... 120
5.3.2 Die Callback-Methoden implementieren ... 126
5.4 Noch mehr Tkinter und ein Taschenrechner für den Raspberry Pi ... 131
5.4.1 Die Layout-Manager ... 131
5.4.2 Eine Klasse für Taschenrechner ... 132

6 Willkommen in der Cloud: Web Apps mit dem Raspberry Pi ... 139
6.1 Erste Schritte mit Flask ... 140
6.1.1 Routing ... 142
6.1.2 HTML und HTML-Templates ... 144
6.1.3 GET und POST ... 149
6.1.4 Ein Crashkurs in SQL und Datenbanken ... 153
6.2 Eine To-do-App für den Raspberry Pi ... 159
6.2.1 Bestandteile der Web App ... 160
6.2.2 Das To-do-Listen-HTML-Template ... 162
6.2.3 Der Python-Code für die To-do-Liste ... 166
6.3 Eine Web App automatisch starten ... 171
6.3.1 Python-Programme starten ohne IDE ... 172
6.3.2 Programme beim Systemstart automatisch starten ... 175

7 Ein Webradio mit dem Raspberry Pi ... 177
7.1 Vorbereitungen für den Raspberry Pi ... 177
7.2 Den Music Player installieren ... 178
7.3 Die Stationsliste erstellen ... 179
7.4 Teil 1: Ein Webradio mit Flask ... 185
7.4.1 Das Webradio-HTML-Template ... 186
7.4.2 Der Python-Code für das Radio-Webinterface ... 190
7.5 Teil 2: Ein Webradio mit Tkinter ... 196
7.5.1 Eine Autoupdate-Funktion hinzufügen ... 197
7.5.2 Die benötigten GUI-Elemente für das Webradio ... 199
7.5.3 Der Python-Code für das Webradio ... 201

8 Datengewinnung aus dem Internet ... 207
8.1 Web Scraping ... 208
8.1.1 Pythons Zugang zum Web – das Modul requests ... 208
8.1.2 JSON- oder XML-Format ... 212
8.1.3 Die wundervolle Suppe aus HTML ... 214
8.1.4 Ein reales Beispiel: Fußballergebnisse scrapen ... 225
8.1.5 Die rechtliche Seite ... 226
8.1.6 Analyse des HTML-Codes ... 226
8.2 Wetterdaten abfragen mit OpenWeatherMap ... 231
8.2.1 Den API-Key erhalten ... 232
8.2.2 API-Abfragen: Die Wetter-App für die Konsole ... 233
8.2.3 Die Wetter-Web App: Flask meets OpenWeatherMap ... 238
8.2.4 APIs für jeden Zweck ... 240

Index ... 243