

Stichwortverzeichnis

Symbole

- - Eigenschaften 365
 - Private Attribute/Methoden 326
- - Befehlszeilenoptionen 32
 - Negation (unärer Operator) 179
- - Befehlszeilenoptionen 32
 - Dekrement 178
- , 172
- ;
- ? 37
- - Attribute 143
 - git add 242
 - Konfigurationsdateien 38
 - Pfade 27, 39
- .. 27, 38
- / 31
- ()
 - Arithmetische Klammern 172
 - Aufrufoperator 139
 - Tupel 172
- []
 - Indexoperator 134
 - Slices 113
- { }
 - Einrückungen 104
 - Stringformatierung 112
- @ 50
- *
 - Argumente für innere Funktionen 197
 - Jokerzeichen 36, 242
 - Positionsargumente 191
 - Variadische Funktionen 192
- **
 - Argumente für innere Funktionen 197
 - Potenzierung 274
 - Schlüsselwortargumente 191
 - Variadische Funktionen 195
- /
 - Befehlszeilenoptionen 32
 - Pfade 25
- /? 32
- \ 113
- \\
 - Maskierung 111
 - Pfade 25
- # 210
- #! 51, 217
- %* 50
- +
 - Stringverkettung 112
 - Unärer Operator 179
- ++ 178
- += 381
- != 172
- *= 381
- = 130
- == 110, 130
- > 221
- __add__() 375
- .bashrc 49
- c 33
- __class__ 369
- @classmethod 351
- __doc__ 218
- __future__ 105
- .gitignore 243
- help 32
- __init__() 324
- /mnt 24
- __mul__() 376
- ((n) 273
- __name__ 300
- @property 365
- .py 34
- .pyc 149
- __qualname__ 329
- __repr__() 372
- %s 112
- @staticmethod 354
- __str__() 372
- __version__ 20
- /Volumes 24

A

- Abelson, Harold 209
- Ablaufverfolgung Siehe Traceback, Fehlermeldungen
- Aktivitätsanzeige 28
- Aktuelles Arbeitsverzeichnis 26, 37
- Algorithmen
 - Binärer Suchalgorithmus 269, 280
 - Komplexitätsanalyse 267
 - Sortieralgorithmen 270, 280
 - Timsort 169
 - XOR-Algorithmus 105, 262
- all() 179
- Amdahls Gesetz 267
- Anführungszeichen 32
- Anweisungen 141
- Anwendungsprogrammierschnittstellen 151
- Any (Typhinweis) 226
- APIs 151
- append() 295
- Arbeitsspeicher
 - Belegung messen 157
 - Deterministische Funktionen 200
 - Garbage Collection 125
 - Optimierung für Integerobjekte 176
 - String-Interning 177
 - Verweise 157
- args 193
- Argumente
 - args 193
 - Befehlszeilenargumente 32
 - Boolesche Argumente 94
 - Flags 94
 - Funktionen als Argumente 200, 201
 - Innere Funktionen 197
 - Iterierbar 193
 - Konstruktorargumente 325
 - kwargs 192, 196
 - Mehrere Elemente als Argumente übergeben 190
 - Positionsargumente 191
 - Schlüsselwortargumente 191
 - Standardargumente 164, 190
 - Unterschied zu Parametern 148
 - Variable Anzahl von
 - Positionsargumenten 192
 - Variable Anzahl von
 - Schlüsselwortargumenten 195
- ASCII-Grafik 289, 352
- ASCII-Sortierung 168
- assert 307
- Attribute
 - Anfangswerte festlegen 324
 - Bearbeiten 325
 - Datenvalidierung 367
 - Definition 143, 321, 325
 - Get-Methoden 149
 - Grundvariable 366
 - In Eigenschaften umwandeln 364
 - __init__() 324
 - Klassenattribute 354, 355
 - Komposition 346
 - Methoden 144
 - Objekte als Attribute 347
 - Privat 326
 - Set-Methoden 149
 - Stringdarstellung 374
 - Unterschied zu Eigenschaften 149
- AttributeError 342
- Aufrufbare Objekte 139
- Aufrufoperator 139
- Ausdrücke 141
- Ausführungsgeschwindigkeit
 - Amdahls Gesetz 267
 - Deterministische Funktionen 200
 - Funktionen 265
 - Messen 106, 263
 - Stringverkettung 167
 - timeit 106, 263
 - Zeitkomplexität 267
- Auskommentieren 84
- Ausnahmen
 - AttributeError 342
 - Dunder-Methoden 376
 - Eigene Ausnahmen auslösen 368
 - Fehlermeldungen 8
 - KeyError 114
 - Leerer except-Block 91
 - ModuleNotFoundError 323
 - NameError 84
 - 'NoneType' object has no attribute 205
 - NotImplemented 376
 - RecursionError 367
 - Reihenfolge von Bedingungen 298
 - StopIteration 145
 - SyntaxError 146
 - TabError 56
 - Traceback 8
 - try 93
 - ValueError 194
 - Vorteil gegenüber Fehlercodes 206

B

Bambrick, Leon 69
Bash 29
Batchelder, Ned 267
Batchskript 50
Bedingte Ausdrücke 118
Befehlszeile
 Argumente 32
 Aus Python-Programm heraus ausführen 35
 Befehlsverlauf 36
 -c 33
 Dateien verwalten 40
 Einführung 29
 Eingabeaufforderung 30
 Flags 32
 Hilfe 32
 Jokerzeichen 36
 Öffnen 30
 Optionen 32
 Ordner auflisten 38
 Ordner verwalten 40
 PATH 45
 Programme ausführen 31
 py 34
 Python-Code ausführen 33
 Python-Programme ausführen 34
 python/python3 34
 Schalter 32
 Schalter kombinieren 38
 Speicherort von Programmen finden 43
 Tabulatorvervollständigung 35
 Terminalfenster leeren 44
 Umgebungsvariablen anzeigen 44
 Unterordner auflisten 39
 Verzeichniswechsel 37
 Vorteile 29
Benutzereingaben 296
Benutzerordner 25, 37, 45
Benutzervariablen 47
Bezeichner Siehe auch Namenskonvention
 Buchstaben weglassen 73
 Funktionen 186
 Integrierte Namen 76
 Länge 72
 Leerzeichen 70
 Leicht zu findende Namen 75
 Nicht lateinische Schriftzeichen 71
 Präfixe 73
 Schlüsselwörter 126
 Schreibweisen 70
 Suffixe 74, 313

Bibliotheken 150
Binärdateien 149
Binärer Suchalgorithmus 269, 280
Binnenmajuskel 70
Biquadratische Zeitkomplexität 271
Black
 Ausführen 64
 Einstellungen 64
 Installieren 63
 Vorschau 66
 Zeilenweise ausschalten 67
Boolesche Werte 180
Bourne-Shell 29
Browser 183
Bytecode 149

C

camelCase 70
Casting 148
cd 31, 37
chdir() 26
chr() 168
class 322, 340, 361
clear 44
close() 109
cls
 Befehlszeile 44
 Parameter 352, 353
Codeblöcke 104, 142
Codepunkte 168
Code Smells
 Angebliche Code Smells 92
 Auskommentierter Code 84
 Duplizierter Code 80
 Einführung 79
 Hartcodierung 297
 Leerer except-Block 91
 Magische Zahlen 82, 297
 Nichtssagende Fehlermeldungen 92
 Numerische Suffixe 87, 313
 Print-Debugging 86
 Toter Code 85
 Überflüssige Klassen 88
 Verschachtelte Listennotation 89
Codetags 216
collections 116
Commit
 Beschreibung 244
 Commit und Bereitstellung kombinieren 244
 Durchführen 242

- Einführung 232
- Einzelne Dateien 244
- Hash 252
- Häufigkeit 248
- Lokale Commits auf Remoterepository
übertragen 259
- Protokoll 252
- Zurücknehmen 254
- Container 137
- Cookiecutter 233
- copy 40
- copy() 114, 162
- cp 40
- cProfile 265
- CPython 124
- cwd() 26

D

Dateien

- Als Befehlszeilenoptionen 32
- Auflisten 38
- Ausführbare Dateien 32
- Binärdateien 149
- Bytecode 149
- close() 109
- Erweiterung 32
- Konfigurationsdateien 38
- Kopieren 40
- Lesen/Schreiben 109
- Löschen 41
- Namen 77
- Nicht geschlossene Dateien 110
- open() 109
- Pfade 24
- Pfad für temporäre Dateien 44
- Projektdateien anlegen 232
- Protokolldateien 87
- .py 34
- .pyc 149
- Python-Programme 34
- Umbenennen 41
- Verschieben 40
- with 109
- Datensystem 24
- Datenträger 24
- Datentypen
 - Container 137
 - Dynamische Typisierung 220
 - Ermitteln 328
 - Explizite Typumwandlung 148
 - Graduelle Typisierung 221

- Implizite Typumwandlung 148
- Klassen 221
- Klassenobjekte 328
- Kommentare 212
- Konstruktor 221
- Literale 126
- Objekte auf bestimmten Typ prüfen 350
- Rückgabewerte 203
- Statische Typisierung 220
- Typalias 227
- Typhinweise 220
- Typinferenz 222
- Typobjekte 328
- Typwarnungen ignorieren 224
- Umwandeln 148
- Ungarische Notation 74
- Unveränderbar 131
- Veränderbar 131
- Debugging
 - Debugger 86
 - Globale Variablen 96
 - Print-Debugging 86
 - Privater Zugriff 328
 - Protokollierung 87
 - Typobjekte 329
- decimal 170
- Deduplizierung 80
- deepcopy() 163
- def 140
- defaultdict() 116
- Dekrementoperator 178
- del 41
- Delete-Methoden 365, 366
- Dictionarys
 - defaultdict() 116
 - Dictionarynotation 90
 - Elemente als Argumente übergeben 190
 - Ersatz für switch 117
 - Geordnet 138
 - get() 115
 - Indexoperator 134
 - Konfigurationseinstellungen 192
 - Kopieren 161
 - Reihenfolge der Elemente 137
 - Schlüssel 135
 - Schlüsselwortargumente 192, 196
 - setdefault() 116
 - Standarddictionary 116
 - Standardwert bei nicht vorhandenem
Schlüssel 115
 - Vorhandensein eines Schlüssels prüfen 114

Diff-Format 66
Diff-Programme 247
dir 37, 38
Distributivgesetz 274
Division durch null 10
Docstrings 217, 294
Dokumentationswerkzeuge 219
Doppelte Freigaben 125
Dunder-Methoden
 Direkt 381
 Einführung 138
 Erweiterte Zuweisungsoperatoren 381
 Kommutativ 379
 Numerisch 375, 378
 other 375
 Reflektiert 379, 381
 self 377
 Stringdarstellung von Objekten 372
 Vergleiche 383, 388
 Zweck 371
Duplizierter Code 80, 342
Dynamische Typisierung 220

E

Easter Egg
 import antigravity 183
 import braces 105
echo 45
Eigenschaften
 Attribute in Eigenschaften umwandeln 364
 Datenvalidierung 367
 Delete-Methoden 365, 366
 Get-Methoden 365, 366
 Grundvariable 366
 Ohne Grundvariable 370
 Schreibgeschützt 369
 Set-Methoden 365, 366, 368
 Sinnvolle Verwendung 371
 Unterschied zu Attributen 149
 Zweck 364
Einchecken 232
Eingabeaufforderung (Windows) 29
Einrückungen 55, 104, 142
Elternordner 38
Engines 151
enumerate() 108
env 44
Erweiterte Zuweisungsoperatoren 178, 381
Etikette 14
except 91

exit() 297
Explizite Typumwandlung 148
Exponentielle Zeitkomplexität 271

F

Faktorielle Zeitkomplexität 272
Fakultät 272
Falsehoods Programmers Believe 336
Fehlercodes 206
Fehlermeldungen
 Ablaufverfolgung 8
 Frame-Objekte 9
 Ignorieren 8
 Klassenname 369
 Nichtssagende Fehlermeldungen 92
 None 204
 Recherchieren 11, 223
 Stacktrace 8
 Tracebacks 8
 Typprüfung 223
 Typwarnungen ignorieren 224
 Zeile mit Ursache finden 11
filter() 203
find 39
Flag-Argumente 94
Flags (Befehlszeile) 32
Fließkommazahlen 169
fmt\
 off/on 67
Folgen 137
for Siehe Schleifen
Ford, Henry 63
format() 112
Formatierung
 Anführungszeichen 65
 Black 63
 Codeblöcke 104
 Einrückungen 55, 104
 Formatspezifizierer 112
 Kommentare 211
 Leerzeichen 57
 Leerzeilen 60
 PEP 8 55
 Rohstrings 111
 Stilrichtlinien 54
 Strings 111
 Übermittlung per E-Mail 18
 Weißraum 55, 105
 Wichtigkeit 53
 Zeilenlänge 64

- Formatspezifizierer 112
- Frame-Objekte 9
- Frameworks 151
- F-Strings 112
- Funktionale Programmierung 198
- Funktionen
 - Aliase 140
 - An andere Funktionen übergeben 140, 200, 201
 - Anonym 201
 - Anzahl der Parameter 189
 - Ausführungsgeschwindigkeit 265
 - Beschränkung auf eine Aufgabe 93
 - Boolesche Argumente 94
 - Deduplizierung 80
 - def-Anweisung 140
 - Definition 143
 - Deterministisch 199
 - Docstrings 294
 - Erstellen 140
 - Fehlercodes 206
 - Fehlermeldungen 9
 - Frame-Objekte 9
 - Globale Variablen 95
 - Höhere Ordnung 200
 - Konstruktor 324
 - Lambda-Funktionen 201, 202
 - Leistung 265
 - Mehrere Elemente als Argumente übergeben 190
 - Methoden 143, 324
 - Namen 71, 186
 - Nebenwirkungen 198
 - Objekte erster Klasse 140, 201
 - Ordnung bestimmen 275
 - Ordnung gängiger Funktionen 281
 - Parallelverarbeitung 200
 - Profiling 265
 - Reine Funktionen 200
 - return-Anweisungen 93
 - Rückgabewerte 203
 - Standardargumente 164, 190
 - Stubs 85
 - try 93
 - Umbenennen 140
 - Umfang 186
 - Variablen zuweisen 140
 - Variadisch 192, 195
 - Von anderen Funktionen zurückgeben 200
 - Wrapper 197
- G**
- Garbage Collection 125, 152, 263
- get() 115
- getcwd() 27
- Get-Methoden 365, 366
- getsizeof() 157
- Git
 - Änderungen bestätigen 244
 - Änderungen einsehen 245, 247
 - Anmeldeinformationen eingeben 236
 - Bereitstellung 239, 242
 - Bereitstellung aufheben 254
 - Bereitstellung und Commit kombinieren 244
 - Commitbeschreibung 244
 - Commit durchführen 242, 244
 - Commithashes 252
 - Commitprotokoll 252
 - Commits auf Remoterepository übertragen 259
 - Commits zurücknehmen 254
 - Dateien hinzufügen 241
 - Dateien ignorieren 243
 - Dateien löschen 249
 - Dateien umbenennen 250
 - Dateien verschieben 251
 - Dateistatus 237
 - Diff-Programme 247
 - Einzelne Dateien bestätigen 244
 - Einzelne Dateien zurücksetzen 255
 - Frühere Versionen wiederherstellen 253
 - git add 242, 244
 - git checkout 254, 256
 - git clone 259
 - git commit 242, 244
 - git config 236
 - git diff 245
 - git help 237
 - GitHub 257
 - GitHub-Repository klonen 259
 - .gitignore 243
 - git init 240
 - git log 241, 252
 - git mv 250
 - git push 259
 - git remote add 258
 - git restore 253, 254
 - git revert 254
 - git rm 249
 - git show 253
 - git status 238, 240

- GUI-Werkzeuge 236, 247
- Häufigkeit von Commits 248
- Hilfedateien 237
- Installieren 235
- Laufende Statusanzeige 241
- Lokale Speicherung 240
- Module importieren 293
- Repository auf GitHub übertragen 258
- Repository erstellen 240
- Sensible Informationen 243, 256
- staged 239
- TortoiseGit 236
- Vim 244
- Gleichheitsoperator 110, 130
- Glossar 152
- Graduelle Typisierung 221
- Groß- und Kleinschreibung
 - Befehlszeilenoptionen 32
 - Benutzereingaben 297
 - Bezeichner 70
 - Klassen 322
 - Pfade 24
 - Programm- und Befehlsnamen 31
- Grundvariable 366

H

- Hartcodierung 297
- hash() 136
- Hash 135, 252
- help() 219
- Hettinger, Raymond 158, 160
- hex() 204
- Hoare, Tony 205, 262
- home() 25

I

- id() 127
- Identität 127
- Identitätsoperator 110, 130
- IDLE 14
- IEEE 754 169
- if-Anweisungen 117
- Implizite Typumwandlung 148
- import
 - from ... import 25, 225
 - import antigravity 182
 - import braces 105
 - import this 100
 - Mehrere Module auf einmal importieren 62
- in 121

- Indexoperator 134
- Initialisierer 322
- Inkrementoperator 178
- input() 296
- Instanzen
 - Klassen 324
 - Programminstanzen 28
- Integer
 - Boolesche Werte 180
 - Genauigkeit 170
 - Vertauschen 105, 262
 - Vorab zugewiesene Integer 176
- Interpreter 124
- is 110, 130
- isinstance() 350
- issubclass() 351
- iter() 144
- Iteratorobjekte 145
- Iterierbare Objekte 144

J

- Java 104, 125, 149
- join() 167
- Jokerzeichen 36
- Just-in-time-Compiler 125
- Jython 124

K

- Kapselung 356
- Karlton, Phil 69
- Kelly-Bootle, Stan 135
- KeyError 114
- Klassen
 - Alternativer Konstruktor 352
 - Auf Kindbeziehung zu anderer Klasse
 - prüfen 351
 - Aufrufbare Objekte 139
 - cls 352, 353
 - Datentypen 221
 - Einführung 317
 - Erstellen 321
 - Gründe für die Verwendung von Klassen 333
 - Importieren 323
 - __init__() 324
 - Initialisierer 322
 - Instanzen 324
 - Kapselung 356
 - Kindklassen erstellen 340
 - Klassenattribute 354, 355
 - Klassenmethoden 351, 355

- Klassenname in Fehlermeldungen 369
- Klassenobjekte 328
- Komposition 346
- Konstruktor 324
- Konstruktorargumente 325
- Mathematische Operatoren nutzen 378
- Mehrere Elternklassen 358
- Methoden 143, 324
- Methoden der Elternklasse aufrufen 344
- Mixins 358
- Namen 71, 322
- Oberklassen 342
- Objekte 318
- Objekte erstellen 321, 322
- Qualifizierter Klassenname 369
- Reale Objekte modellieren 335
- Reihenfolge der Methodenauflösung 361
- Stringdarstellung 373
- Stubs 85
- Typobjekte 328
- Überflüssige Klassen 88
- Unterklassen 341
- Vererbung 341
- Klauseln 142
- Knuth, Donald 105, 262
- Koeffizienten 274, 277
- Kommandozeile Siehe Befehlszeile
- Kommentare
 - \ 214
 - # 210
 - Blockkommentare 211
 - Code auskommentieren 84
 - Codetags 216
 - Datentypen 212
 - Docstrings 217
 - Einzeilig 210
 - Formatierung 211
 - Formulierung 215
 - Gliederung des Codes 213
 - Informativ 213
 - Inline-Kommentare 211, 212
 - Leerzeichen 59, 211
 - Leerzeilen 211
 - Links 212
 - Magische Kommentare 217
 - Mehrzeilig 210
 - Notwendigkeit 96
 - Nutzlose Kommentare 213
 - Rechtliche Hinweise 215
 - TODO 216
- Typhinweise 227
- Typhinweise ignorieren 224
- Kompare 247
- Komplexitätsanalyse
 - Binärer Suchalgorithmus 280
 - Biquadratische Zeitkomplexität 271
 - Einführung 267
 - Exponentielle Zeitkomplexität 271
 - Faktorielle Zeitkomplexität 272
 - Faustregeln 283
 - Kleines n 284
 - Koeffizienten 277
 - Konstante Zeitkomplexität 269
 - Kubische Zeitkomplexität 271
 - Landau-Notation 268
 - Lineare Zeitkomplexität 270
 - Logarithmische Zeitkomplexität 269
 - n-log-n-Zeitkomplexität 270
 - O(1) 269
 - O(2n) 271
 - O(log n) 269
 - Omega-Notation 273
 - O(n!) 272
 - O(n) 270
 - O(n²) 270
 - O(n log n) 270
 - O-Notation 268
 - Ordnungen 268
 - Ordnung gängiger Funktionen 281
 - Ordnung von Code bestimmen 275
 - Polynomielle Zeitkomplexität 270
 - Quadratische Zeitkomplexität 271
 - Schleifen 275, 278
 - Terme niedriger Ordnung 276
 - Theta-Notation 273
 - Verhalten der verschiedenen
 - Ordnungen 277
 - Verschachtelte Schleifen 279
- Komposition 346
- Konstanten
 - Ändern 83
 - Global 96
 - Magische Zahlen 82
 - Namen 71
- Konstante Zeitkomplexität 269
- Konstrukturen
 - Alternativer Konstruktor 352
 - Argumente 325
 - Datentypen 221
 - Definition 324
 - __init__() 324

Kubische Zeitkomplexität 271
kwargs 192, 196

L

lambda 201
Lambda-Funktionen 201, 202
Landau-Notation Siehe Komplexitätsanalyse
Laufwerke 24, 37
Laufzeitfehler 146

Leerzeichen

Abstände innerhalb einer Zeile 57
Benutzereingaben 297
Einrückungen 55
Kommentare 211
Namen 70

Leerzeilen 60, 211

Leistung

Amdahls Gesetz 267
Deterministische Funktionen 200
Funktionen 265
Komplexitätsanalyse 267
Messen 106, 263
Ordnung gängiger Funktionen 281
Ordnung von Code bestimmen 275
Profiling 106, 265
Schleifen 278
Stringverkettung 166
Verschachtelte Schleifen 279
Vorzeitige Optimierung 105, 262

len() 137

Lineare Zeitkomplexität 270

Linten 13

Listen

Direkte Bearbeitung 132
Elemente als Argumente übergeben 190
Elemente in einer Schleife ändern 158
Elemente in einer Schleife hinzufügen/
entfernen 154
Elemente nach dem Rückgabewert einer
Funktion sortieren 202
enumerate() 108
Filtern 203
Indexoperator 134
Kopieren 113, 161
Listennotation 89, 157, 203
Rückwärts durchlaufen 159
Slices 113
Stacks 294
Verketteten 132
Verschachtelte Listen kopieren 163
Zuordnen 202

List (Typhinweis) 226

Literale 126

Lochkarten 64

log() 274

Logarithmische Zeitkomplexität 269

Logarithmus 269, 274

logging 87

lower() 168

ls 37, 38

M

Magische Kommentare 217

Magische Methoden Siehe Dunder-Methoden

Magische Zahlen 82, 297

Mailinglisten 15

map() 202

Maps 137

Mars Climate Orbiter 74

Maschinencode 149

Maskierungszeichen 111

max() 193

MCR 17

md 43

Mehrere Anweisungen in einer Zeile 62

Mehrfachvererbung 358

Mehrfachzuweisung 107, 263

Meld 247

Mengennotation 90

Metasyntaktische Variablen 70

Methoden

cls 352, 353

Definition 143, 324

Delete-Methoden 365, 366

Get-Methoden 149, 365,

Klassenmethoden 351, 355

Mehrfachvererbung 358

Methoden aus Methoden heraus
aufrufen 335

Methoden der Elternklasse aufrufen 344

Namen 71, 186

None 349

Privat 326

Reihenfolge der Methodenauflösung 360

self 324

Set-Methoden 149, 365, 366

Statisch 354, 355

Überschreiben 342

Vererbung 341

min() 193

Mixins 358

- mkdir 43
- Module
 - collections 116
 - cookiecutter 233
 - copy 114
 - cProfile 265
 - decimal 170
 - Definition 139
 - __future__ 105
 - Importieren 25, 293
 - logging 87
 - Mehrere Module auf einmal importieren 62
 - Namen 71
 - operator 383
 - Pakete 139
 - pathlib 25
 - Standardbibliothek 139
 - subprocess 35
 - timeit 106, 263
 - typing 225, 227, 228
 - webbrowser 183
- ModuleNotFoundError 323
- Monty Python 70
- move 40
- mro() 360
- mv 40, 41
- Mypy 223

- N**
- Nachkommastellen 171
- NameError 84
- Namenskonvention
 - Buchstaben weglassen 73
 - Funktionen 186
 - is/has 74
 - Klassen 322
 - Kulturelle Anspielungen 75
 - Länge 72
 - Leicht zu findende Namen 75
 - Maßeinheiten 74
 - PEP 8 71
 - Präfixe 73
 - Projektordner und -dateien 232
 - Suffixe 74
 - Ungarische Notation 74
 - Wortspiele 75
- Nebenwirkungen 198
- next() 145
- n-log-n-Zeitkomplexität 270

- None
 - Methoden 349
 - Rückgabewert 204
 - Standardargument 165
 - Typhinweise 225
 - Vergleiche 110
- no-op 85
- NotImplemented 376
- NULL-Verweis 205

- O**
- O(1) 269
- O(2n) 271
- object 360
- Objekte
 - Als Attribute 347
 - Attribute 143, 321, 325
 - Auf bestimmten Typ prüfen 350
 - Aufrufbar 139
 - Datentyp ermitteln 328
 - Definition 127
 - Erstellen 321, 322
 - Hash 136
 - Identität 127
 - Iteratorobjekte 145
 - Iterierbar 144
 - Klassen 318
 - Klassenobjekte 328
 - Konstruktor 324
 - Methoden 324
 - Stringdarstellung 372
 - Typobjekte 328
 - Veränderbarkeit 131
- Objektorientierte Programmierung Siehe OOP
- O(log n) 269
- Omega-Notation 273
- O(n!) 272
- O(n) 270
- O(n²) 270
- Onlineforen 15
- O(n log n) 270
- O-Notation Siehe Komplexitätsanalyse
- OOP
 - Attribute 325
 - Einführung 317
 - Kapselung 356
 - Klassen 318
 - Komposition 346
 - Mehrfachvererbung 358
 - Methoden 324

- Methoden der Elternklasse aufrufen 344
- Nachteile 317
- Objekte erstellen 320
- Polymorphismus 356
- Privater Zugriff 326
- Reale Objekte modellieren 335
- Überladen von Operatoren 375
- Vererbung 340
- Vorteile 335
- Wiederverwendung von Code 343
- open()
 - Browser 183
 - Dateien 109
- Operatoren
 - 178
 - () 139
 - [] 134
 - + 112
 - ++ 178
 - != 172
 - = 130
 - == 110, 130
 - Aufrufoperator 139
 - Dekrementoperator 178
 - Dunder-Methoden für mathematische Operatoren 377, 380
 - Dunder-Methoden für Vergleichsoperatoren 388
 - Erweiterte Zuweisungsoperatoren 178, 381
 - Gleichheitsoperator 110, 130
 - Identitätsoperator 110, 130
 - in 121
 - Indexoperator 134
 - Inkrementoperator 178
 - is 110, 130
 - Kommutativ 379
 - Kurzschließen 119
 - Mathematische Operatoren 375
 - Stringverkettung 112
 - Ternär 118
 - Überladen 375
 - Unär 179
 - Ungleichheitsoperator 172
 - Vergleichsoperatoren 120
 - Verkettung 120, 172
 - Zuweisungsoperator 120, 130
- operator (Modul) 383
- Optional (Typhinweis) 225
- ord() 168
- Ordner
 - Aktuelles Arbeitsverzeichnis 26, 37
 - Als Befehlszeilenoptionen 32
 - Benutzerordner 25, 37, 45
 - Einführung 24
 - Elternordner 38
 - Erstellen 43
 - Inhalt auflisten 38
 - Kopieren 40
 - Löschen 41, 43
 - Stammordner 24
 - Umbenennen 41
 - . und .. 27
 - Unterdner auflisten 39
 - Verschieben 40
 - Verzeichniswechsel 37
 - Wurzelordner 24
- other 375
- P**
- Pakete 139
- Parallelverarbeitung 200
- Parameter
 - args 193
 - cls 352, 353
 - Komplexität 189
 - kwargs 192, 196
 - Numerische Suffixe 87
 - other 375
 - self 324
 - Standardargumente 190
 - Unterschied zu Argumenten 148
- Parametrisierung 81
- Parsingfehler 146
- PascalCase 70
- pass 85, 91
- path 46
- Path() 25
- pathlib 25
- Path-Objekte 25
- PATH (Umgebungsvariable) 31, 45
- P-Code 149
- PEP 8 55, 71
- PEP 257 220
- Permutationen 272
- Peters, Tim 100

Pfade

- Absolut 27
 - Aktuelles Arbeitsverzeichnis 26, 37
 - Backslashes maskieren 111
 - Benutzerordner 37
 - Elternordner 38
 - pathlib 25
 - Path-Objekte 25
 - Relativ 27
 - Rohstrings 111
 - Umgebungsvariable PATH 45
 - Umwandeln 25
 - . und .. 27
 - Unterschiede zwischen Betriebssystemen 25
- Pike, Rob 284
- pip
- pip list 19
 - pip/pip3 14
- Polymorphismus 356
- Polynomielle Zeitkomplexität 270
- pop() 295
- Portierbarer Code 149
- Positionsargumente 191, 192
- POSIX 25
- Potenzierung 274
- Präfixe 73
- print() 86, 141, 191
- Print-Debugging 86
- Problem des Handelsreisenden 272
- Profiling 106, 265
- Protokollierung 87
- Prozesse 28, 45
- py.exe 34
- Pyflakes 14
- PyPy 125
- Python-Programme
- Ausführen an der Befehlszeile 35
 - Batchskript 50
 - Beenden 297
 - Bytecode 149
 - Dateiendung 34
 - Garbage Collection 125
 - Linux 51
 - macOS 51
 - Module 139
 - Projektdateien anlegen 232
 - .pyc-Dateien 149
 - .py-Dateien 34
 - Speicherverwaltung 125
 - Windows 49
- python/python3 14, 34

Q

- Quadratische Zeitkomplexität 271

R

- Ramalho, Luciano 357
- range() 108
- Rapid Environment Editor 48
- rd 43
- RecursionError 367
- Reddit 18
- ren 41
- Repository

 - Dateien hinzufügen 241
 - Definition 232
 - Erstellen 240
 - Informationen entfernen 256
 - Lokales Repository 240
 - Lokales Repository auf GitHub übertragen 258
 - Remoterepository 240
 - Remoterepository erstellen 257
 - Remoterepository lokal klonen 259

- repr() 372
- return 93
- rm 41
- rmdir 43
- Rohstrings 111
- Rollback 253
- run() 35
- Rundungsfehler 169

S

- Schalter (Befehlszeile) 32
- Schleifen

 - Deduplizierung 81
 - Elemente einer Liste ändern 158
 - Elemente in einer Liste hinzufügen/entfernen 154
 - Endlosschleife 154
 - enumerate() 108
 - Iteratorobjekte 145
 - Iterierbare Objekte 144
 - Listennotation 90, 157
 - Listen rückwärts durchlaufen 159
 - Verschachtelt 279
 - Zeitkomplexität 278, 279

- Schlüsselwortargumente 191, 195
- Schlüsselwörter 126
- Schwerelosigkeit 182
- SDKs 151

- self 324
 - Semantische Fehler 147
 - Sensible Informationen
 - Git 243, 256
 - repr-Strings 374
 - Sequenzen 137
 - set 44
 - setdefault() 116
 - Set-Methoden 365, 366, 368
 - setx 48
 - Shebang-Zeile 51, 217
 - Shell Siehe Befehlszeile
 - Shellskript 51
 - Skripte 150
 - Slices 113
 - snake_case 70
 - Snapshots Siehe Git, Commit
 - Software Development Kits 151
 - sort() 168
 - sorted() 202
 - Sortierung
 - ASCII-Sortierung 168
 - Dunder-Vergleichsmethoden 387
 - Folgentypen 386
 - Lambda-Funktionen 202
 - Objekte eigener Klassen sortieren 383
 - Ordnung von Sortieralgorithmen 270, 280
 - Speicherlecks 125
 - Speicherverwaltung 125
 - Stack Overflow 17
 - Stacks 294
 - Stammordner 24
 - Standardargumente 164, 190
 - Standardbibliothek 139
 - Statische Analyse
 - Im Hintergrund 223
 - Linten 13
 - Mypy 223
 - Typhinweise 221
 - Typwarnungen ignorieren 224
 - Statische Typisierung 220
 - Stilrichtlinien 54
 - StopIteration 145
 - str() 372
 - Strings
 - Ändern 166
 - Anführungszeichen 65
 - format() 112
 - Formatieren 111
 - Formatspezifizierer 112
 - F-Strings 112
 - Interpolation 112
 - Mehrzeiliger String als Kommentar 210
 - Rohstrings 111
 - Slices 113
 - Sortieren 168
 - Stringdarstellung von Objekten 372
 - String-Interning 177
 - Unveränderbarkeit 133
 - Verketten 133, 166
 - strip() 296
 - Stubs 85
 - subprocess 35
 - Suffixe 74, 313
 - sum() 193
 - super() 344, 360, 361
 - Sussman, Gerald und Julie 209
 - switch 117
 - SyntaxError 146
 - Syntaxfehler 146
 - Systemvariablen 47
- ## T
- TabError 56
 - Tabulatoren 56
 - Tabulatorvervollständigung 35
 - Taschenrechnerprogramm 31
 - Task-Manager 28
 - Tastatureingaben 296
 - Teile und herrsche 269
 - TEMP 44
 - Terminal Siehe Befehlszeile
 - Ternärer Operator 118
 - Theta-Notation 273
 - Threadsicherheit 200
 - Tic-Tac-Toe 329
 - timeit 106, 263
 - Timsort 169
 - tkdiff 247
 - TODO 216
 - TortoiseGit 236
 - Toter Code 85
 - Traceback
 - Frame-Übersicht 9
 - Untersuchen 8
 - Zeile mit Ursache finden 11
 - try 93
 - Tupel
 - Einelementiges Tupel 172
 - Elemente als Argumente übergeben 190

- Ohne Klammer 298
- Unveränderbarkeit 133
- Turm von Hanoi 288
- Typaliase 227
- type() 328
- Typhinweise
 - Any 226
 - Containertypen 226
 - Einfügen 221
 - Ignorieren 224
 - Mehrere Typen 225
 - None 225
 - Rückportierung 227
- Typinferenz 222
- typing 225, 227, 228
- Typobjekte 328

U

- Umgebungsvariablen
 - Ändern 46
 - Anzeigen 44
 - Benutzerordner 45
 - Benutzervariable 47
 - Dauerhaft ändern 47
 - Einführung 44
 - HOME/HOMEPath 45
 - Kindprozesse 45
 - PATH 31, 45
 - Systemvariable 47
 - TEMP 44
- Ungarische Notation 74
- Ungleichheitsoperator 172
- Unicode 168
- Union (Typhinweis) 225
- Unveränderbare Objekte 131
- upper() 296
- U Thant 335

V

- Validierung 367
- ValueError 194
- Van Rossum, Guido 100, 119
- Variablen
 - Attribute 325
 - Definition 143
 - Direkte Änderung veränderbarer Objekte 132
 - Etikettenmetapher 129
 - Global 95
 - Grundvariable 366

- Integerwerte vertauschen 105, 262
- Kistenmetapher 129
- Klassenpräfix 73
- Konstanten 82
- Kopieren 161
- Metasyntaktisch 70
- Namen 71
- Numerische Suffixe 87
- Umgebungsvariable 44
- Unveränderbarkeit 131
- Verweise 129
- Werte prüfen 120
- Veränderbare Objekte
 - Als Standardargumente 164
 - Datentypen 131
 - Kopieren 161
- Vererbung
 - Beziehungen von realen Objekten 342
 - Einführung 340
 - Mehrere Elternklassen 358
 - Mehrfachvererbung 358
 - Methoden überschreiben 342
 - Mixins 358
 - Nachteile 348
 - Reihenfolge der Methodenauflösung 360
 - Unangemessene Vererbung 346, 357
 - Vorteile 343
 - Wiederverwendung von Code 343
- Vergleichsoperatoren 120
- Versionssteuerung Siehe auch Git
 - Diff-Programme 247
 - Frühere Versionen wiederherstellen 253
 - Git 235
 - GitHub 257
 - Häufigkeit von Commits 248
 - Module importieren 293
 - Online 257
 - Repository 232, 240
 - Rollback 253
 - TortoiseGit 236
 - Zweck 231
- Verzeichnisse Siehe Ordner
- Vier gewinnt 301
- Vim 244
- Vorzeitige Optimierung 105, 262

W

watch 241
webbrowser 183
Weißraum 55, 105
where 43
which 43
while Siehe Schleifen
WinMerge 247
with 109
Wurzelordner 24

X

XKCD 17, 123, 183, 259
XOR-Algorithmus 105, 262

Z

Zeichencodierung 217
Zeilenlänge 64
Zeitkomplexität Siehe Komplexitätsanalyse
Zen of Python 100
Z-Shell 29
Zuweisungsoperator 120, 130