

Die in jeder Python-Version hinzugefügten Erweiterungen bzw. entfernten Eigenschaften ("What's new in Python x.y")

Doku --> <http://docs.python.org/3/whatsnew/index.html>

Ver	Jahr	Beschreibung	PEP
3.11	2022	...	604
3.10	2021	Allow writing union types as X Y Parameter specification variables Precise line numbers for debugging and other tools Add optional length-checking to zip Parenthesized context managers officially allowed Explicit type aliases Structural pattern matching: Specification Structural pattern matching: Motivation and rationale Structural pattern matching: Tutorial Add optional EncodingWarning	604 612 626 618 613 634 635 636 597
3.9	2020	Union operator for dict Type hinting generics in standard collections Relaxed grammar restrictions on decorators String methods to remove prefixes and suffixes Flexible function and variable annotations CPython now uses a new parser based on PEG range/tuple/set/frozenset/list/dict faster vectorcall Garbage collection does not block on resurrected objects IANA time zone database in new module "zoneinfo" Topological graph sort offered in new module "graphlib" CPython adopts an annual release cycle	584 585 614 616 593 617 590 615 602
3.8	2019	Type annotation/Hinting Assignment Expression := ("Walrus" Operator) Positional only parameters (a, b, /, c, d, *, e) PYTHONCACHEPREFIX (Option -X) F"{EXPR}" --> "EXPR=VALUE-OF-EXPR"	572
3.7	2018	PYTHONCOERCELOCALE Data Class @dataclass Neue C-API für Thread-lokalen Speicher in CPython Module "time" kennt Nanosekunden Neuer UTF-8 Modus "surrogateescape" (für POSIX locale) (-X utf8, PYTHONUTF8) Neuer "Development mode" (-X dev, PYTHONDEVMODE) Hash-basierte "*.pyc"-Checks (statt Zeitstempel-basiert) Funktionsdef/Aufruf mit mehr als 255 Parametern/Argumenten Viele Standard-Module erweitert dict bewahrt Schlüssel-Reihenfolge (jede Implementierung) Schlüsselworte "async" und "await"	557
3.6	2016	Literal "f" für formatierte Strings Syntax für Annotation von Variablen (Funktionen) Asynchrone Generatoren und Comprehensions CPython-dict-Implementierung bewahrt Schlüssel-Reihenfolge	
3.5	2015	Operator "@" für Matrizen-Multiplikation Koroutinen mit "async" und "await" Type hints (Modul "typing")	484
3.4	2014	Umfassende PIP-Integration Modul "enum" (Aufzählungs-Datentyp) "asyncio"-Bibliothek	
3.3	2012	Syntax "yield from" (Generator Delegation) Syntax u"STRING" wieder erlaubt (Unicode explizit) Virtual Environments "venv" Implicit Namespace Packages Module: faulhandler ipaddress lzma unittest.mock venv	405 420
3.2	2011	"*.pyc"-Kompilate in Unterverz. "__pycache__" sammeln Viele Standard-Module erweitert	

3.1	2009	Datentyp "OrderedDict" im Modul "collections" Methode "bit_length" des Datentyps "int" Autom. Numerierung von Platzhaltern in String-Formatierung Neuer Algorithmus zur internen Repr. von "float"-Werten atexit
3.0	2008	Änderungen die nicht auch in Version 2.6 enthalten sind: Viele Funktionen/Methoden geben Iterator statt Liste zurück Datentyp "long" entfernt Datentyp "bytes" für Binärdaten Datentyp "str" speichert ausschließlich Unicode-Strings Dictionary-Comprehension Reine Schlüsselwort-Parameter Anweisung "nonlocal" Literale für Datentyp "set" Viele Standard-Module entfernt, umbenannt, verändert Annotationen
2.7	2010	Letzte Version von Python 2 (enthält Neuerungen von 3.1/2 Dictionary und Set-Comprehension Modul "argparse")
2.6	2008	Zeitgleich zu 3.0 entwickelt (enthält Neuerungen, die die Abwärtskompatibilität nicht beeinträchtigen) Kontext-Manager "with" Paket "multiprocessing" Neue Syntax zur String-Formatierung ("...".format(...)) Funktion "print" (statt Anweisung) Oktale und binäre Zahlen-Literale
2.5	2006	Conditional Expression Relative "import"-Anweisung Vereinheitlichung "try-except-finally"-Anweisung (vorher "except" und "finally" nicht gemeinsam verwendbar)
2.4	2004	Generator Expressions Dekoratoren für Funktionen und Klassen Modul "decimal"
2.3	2003	Datentyp "set" (erst im Modul "sets", ab 2.4 eingebaut) Generatorfunktionen Datentyp "bool"
2.2	2001	Verschachtelte Namensräume New-Style Klasse erlaubt Erben von built-in Datentypen Mehrfachvererbung Properties Iterator-Konzept Statische Methoden + Klassenmethoden Automatische Konvertierung Datentypen "int" und "long"
2.1	2001	Modul "__future__" Vergleichsoperationen einzeln überladbar "import" auch unter Windows/macOS case-sensitiv
2.0	2000	Datentyp "unicode" für Unicode-Strings List Comprehension Erweiterte Zuweisung (augmented assignment +=, ...) Garbage Collection für zyklische Referenzen
1.0	1994	Veröffentlichung 1. Version