

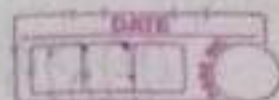
PEP - Index of Python Enhancement Proposals (PEPs)

• Introduction

This PEP contains the index of all Python Enhancement Proposals, known as PEPs. PEP numbers are signed by the PEP editors and once assigned and never changed. The version control history of the PEP texts represent their historical record.

→ Index by category Meta-PEPs (PEPs about PEPs or Process)

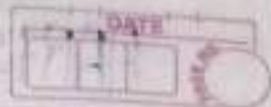
PEP	PEP title	PEP Author(s)
1	PEP purpose and Guidelines	Goodger, Hyton
4	Deprecation of standard models	Cannon, Löwis
5	Guidelines for language Evolution	Prescod
6	By fix Releases	Aahz, Baxter
7	Style Guide for C code	GVR, Warsaw
8	Style Guide for python code	Warsaw
10	Voting Guidelines	Warsaw
11	Removing support for some little used platform forms	Von Louis, Cannon
12	Sample restructured Text PEP Template	Goodger, Cannon
87	Backwards Compatibility policy	Peter Sorn
609	PyPA Governance	Ingram, Ciedam, Hanikane Swara



other informational PEPs

PEP	PEP title	PEP Author(s)
1 13	Python Language Governance	community
1 20	The Zen of Python	Peters.
1 19	Doing python Releases 101	Warsaw, GVR
IF 247	API for cryptographic Hash functions	Kuchling
IF 248	Python Database API v1.0	Lernburg
IF 249	Python Database API v2.0	Lernburg
IF 257	Docstring conventions	Croodgers, GVR
IF 272	API for Block Encryption Algorithms v1.0	Kuchling
1 287	restricted Text Dowstring format	Croodgers
1 290	code migration and modernization	Hettinger
IF 291	Backward compatibility for the python 2 standard	Norwitz
IF 333	Python Web Server Gateway Interface v1.0	Fby.
1 394	The "Python" command on Unix-like systems	Staley, Coghlan
IF 404	Python 2.8 on-release schedule	Warsaw
IF 299	Pure python /c Accelerator module compatibility	cannon
1 411	Provisional patches in the Python standard library	Coghlan, Bender sky
IF 430	Migrating to Python 3 as the default online	Coghlan
1 434	IDLE Enhancement Exception for All Branches	Rovibo, Reedy
1 440	version identification and Dependency.	Coghlan, stufit

AARIS MALIK FYIT-122



IF 452	API for cryptographic Hash function V2.0	Kuchling, Haimen
IF 457	Notation for positional-only parameters	Hastings.
I 478	Python 3.5 Release Schedule	Hastings.
IF 482	Python Literature Overview for Type Hints	GUR, levkivskyi
IA 503	Simple Repository API	Stefft
I 508	Dependency specification for python Software Packages.	Collins.
I 514	Python registration is the windows registry	Dawers
I 537	Python 3.7 Release Schedule	Odly
I 569	Refactoring C Functions and methods	Demeyer et
I 596	Python 3.9 Release Schedule	langa
I 599	Python 3.8 Release Schedule	langa
IA 600	Future 'many linux' platform Tags for portable linux	Smith, Kluyver
IA 602	Annual Release cycle for Python	Lang
IF 607	Reducing CPython's Feature Delivery Latency	Langa, Dower.
IF 627	Recording installed projects	Viktorin
I 630	Isolating Extension modules	Viktorin
IF 635	Structural Pattern matching Motivation and Rationale.	Kohn, GUR
IF 636	Structural Pattern matching Tutorial	moisset.

IA 656	Platform Tag for Linux Distributions Using MUSL	chung
I 672	Unicode-related Security considerations for Python	Viktorin
IF 3333	Python webserver Gateway Interface v1.0.1	Eby
I 8000	Python language Governance proposal overview	Warsaw
I 8002	Open source Governance proposal overview 2	Smit Stuftt
IA 8016	Tastuning concil model	Smith, Stuftt
I 8100	January 2019 steering election	Smith, III
I 8101	2020 Term steering concil election	Jodlowsha, III
I 8102	2021 Term steering concil election	Jodlowsha, III, Carley
I 8103	2022 Term steering concil election	Jodlowsha, III, Carrey

Provisional PEPs

PEP	PEP title	PEP Author's
SP 484	Type Hints	GIWR, Lentosalo, Langa

Accepted PEPs

SA 458	Secure PyPI downloads with signed repository metadata	Kuppusamy, Didz, moore, puchringa, Lock, DeLong, Cappsos
SA 544	Protocols: Structural Subtyping (static duck typing)	Leukivskyi, Lehtosalo, Langa
SA 560	Core support for Typing module and generic Types	Leukivskyi

SA 561	Distributing and packaging Type Information	Smith
SA 563	Postponed Evaluation of Annotations	Langa
SA 570	Python Positional-only Parameters	Hastings, Gratindo
SA 572	Assignment Expressions	Angelico, peters, CVR
SA 578	Python Runtime Audit Hooks	Dower
SA 585	Type Hinting Generics in Standard Collections	Langa
SA 586	Type Hinting Literal Types	Lee, Lehtasalo
SA 589	Typed Dict: Type Hints for Dictionaries with a fixed	Lehtasalo
SA 590	Vectorcall: a fast calling protocol for C Python	Shannon, Demeyer
SA 591	Adding a final qualifier to typing	Sullivan, Levkivskyi
SA 593	Flexible Function and variable annotations	Varoquaux, kashin
SA 604	Allow writing union types as X Y	Prados, Moss
SA 612	Parameters Specification variables	Mendoza zhu
SA 613	Explicit Type Aliases	
SA 615	Support for the IANA Time Zone Database in the	Gransble
SA 616	String methods to remove prefixes and suffixes	sweeney.

AARIS MALIK FYIT-122

SA 617	New PEG parser for CPython	GVR, Grinido, Nikolaou
SA 623	Removes tr from unicode	naoki
SA 631	Dependancy specification in Py project tom 1 based on	Leu
SA 632	Deprecate distutils module	Dowser
SA 634	Structural Pattern matching Specification	Bucher, GVR
SA 646	Varadic Generics	mendoza
SA 647	User-Defined Type words	Traut
SA 654	Exception groups and except	Katriel, GVR
SA 658	serve Distribution metadata in the simple	chung
SA 660	Editable installs for py project tom 1 based builds	Hoith, Bidoul
SA 673	Self type	Srinivasan, Hilton, - Balfe
SA 3121	Extension module initialization and finalization	Von Louis

Open PEPs (Under consideration)

PEP	PEP Title	PEP Author(s)
S 467	minor API improvements for binary	Coghlan, Furman
S 480	Surviving compromise of PyPI: End to end Signing	Kuppasamy, Dia Cappos, Moore
S 554	Multiple interpreters in the stdlib	Snow
S 558	Defined semantics for locals()	Coghlan
S 582	Python local packages directory	Das, Dowers.
S 588	Gitlab Issues migration plan	Mariatta
S 594	Removing dead batteries from the standard library	Heimes, Cannon
S 603	Adding a frozen map type to collections	Selivanov
I 619	Python 3.10 Release Schedule	Salgado
S 620	Hide implementation details from the C API	Stimmer
S 625	File name of a source distribution	Chung, Moore
S 638	Syntactic as macros	Shannon
S 638	Improving License Clarity with Better Package metadata	Ombredanne, Gierlach
S 642	Explicit pattern syntax for structural pattern matching	Coghlan
S 645	Allow writing optional types as x?	Moss
S 649	Deferred Evaluation of Annotation Using Descriptions.	Hastings
S 655	Making individual Typed Dict items as required or...	Faster
I 659	Specializing Adaptive Interpreter	Shannon
S 661	Sentinel values	Ginat
I 664	Python 3.11 Release Schedule	Salgado
S 667	Consistent views of name spaces	Shannon
I 668	Graceful operation between external and python	Thomas, Klose
S 669	Low impacting monitoring of python	Shannon

S	670	Convert macros to functions in the Python CAPI	Aasland, stinner
S	671	Syntax for late-bound function argument defaults	Angelico
S	674	Disallow using macros as L-values	stinner
S	675	Arbitrary literal string Type	Srinivasan, Belaney
S	676	PEP infrastructure Process	Turner
S	678	franching Exceptions with notes	Hatfield - Dodds
S	679	Allow parentheses in assert statements.	Salgado
S	680	tomllib: Support for parsing TomL in the standard	Hukkinen
S	681	Data Class Transforms	Bonte, Traut
S	682	Format specific for signed	Belmonte
S	683	Immortal zero	Belmonte
I	801	Reserved	Warsaw

AARIS MALIK PYIT - 122

Finished PEPs.

PEP	Title	Authors
100	Python Unicode integration	Lembury
201	Lockstep Iteration	Warsaw
202	List comprehensions	Warsaw
203	Augmented Assignments	Wouters
205	Weak References	Drake
207	Rich comparisons	GVR, Asher
208	Reworking the coercion model	Schmenauer
214	Extended print statement	Lembury
217	Display Hook for Interactive Use	Warsaw
218	Adding a Built-Inset object Type	Zadka
221	Import As	Wilson, Heltinger
223	Change the meaning of /x Escapes	Wouters
227	statically Nested scopes	Peters
229	Using Distutils to Build Python	Hyllon
230	Warning Framework	Kochling
232	Function Framework Attributes	GVR
234	Function iterators	Warsaw
235	Import on case-insensitive platforms	Yee, GVR
236	Back to the — Future —	Peters
237	unifying long Integers and Integer	Peters
238	changing the Division operator	Zadka, GVR
241	Metadata for Python software	Zadka, GVR
250	Using site-packages on windows	Kuchling
252	Making Types look more like classes	moore
253	Subtyping Built-in Types	GVR
255	Simple Generators	GVR
260	Simplify xrange()	Schmenauer
		GVR

PEPs	Title	Author
261	Support for "wide" unicode characters	Prescod
263	Defining python source code	Lemburg, von lavis
264	Future statements in simulated shells	Hudson
273	Import modules from zip Archives	Ahlstrom
274	Dict comprehensions	Warsaw
277	Unicode file name support for window NT	Hodgson
278	Universal newline support	Jansen
279	The enumerate() built-in function	Hettiger
282	The logging system	Sajip
285	Adding a bool type	GVR
289	Generator Expressions	Hettiger
292	Simpler string substitutions	Warsaw
293	Codec Error Handling callbacks	norwald
301	Package Index and metadata for Distutils	Jones
302	New import Hooks	JVR, Moore
305	CSV File API	Attis, Cole, Moma
307	Extensions to the pickle protocol	GVR, Hettiger
308	Conditional Expressions	Harris
309	Partial function Applications	Hammond
311	Simplified Global Interpreter	Kuchling, Jones
314	Metadata for python software	Smith
318	Decorators for python and methods	Hettiger
322	Reverse Iteration	Astrand
324	Subprocess	Bahista
327	Decimal Data Type	Aahiz
328	Imports: multi-line and Absolute	Reis
331	Locale - Independent Float/String	GVR, Peters

AARIS MALIK FYIT - 122

338	Executing modules as scripts	Coghlan
341	Unifying try - except and try - finally	Brandi
342	The "with" statements	GVR, Coghlan
343	Metadata for python software packages 1.2	Jones
345	Meta for python	GVR
352	Required superclass for Exception	Cannon, GVR
353	Using size_t as the index type	Von Louis
357	Allowing Any object to be used for slicing	Oliphant
358	The "bytes" object	GVR
362	Function signature object	Cannon,
366	Per user site-packages dictionary	Heimes
371	Addition of the multiprocessing	Noller, Oudkerk
376	Database of installed python Distributions	Ziad
378	Format Specific for Thousands	Hettinger
380	Syntax for Delegating to a subgenerator	Fwing
383	Non-decodable Bytes in system character interfaces	V. Louis
384	Defining a stable ABI	V. Louis
389	algpaux - New command	Bethard
391	Dictionary - Configuration	Sqjip

Historical Meta-PEPs and PEPs

2 PEPs	Title	Author(s)
82	Procedure for Adding New modules	Fasssen
9/60	Python 1.6 Release Schedule	Drake
4220	Python 2.0 Release Schedule	Hylton
10226	Python 2.1 Release Schedule	Hylton
251	Python 2.4 Release Schedule	Warsaw, CNR
320	Python 2.3 Release Schedule	Warsaw, CNR
347	migrating the python cvs to Subversion	Van Louis
356	Python 2.5 Release Schedule	Norwitz, CNR
360	Externally maintain packages	Cannon
361	Python 2.6 and 3.0 Release Schedule	Norwitz, Warsaw
373	Python 2.7 Release Schedule	Peterson
374	Choosing a distributed VCS for the Python Project	Cannon, Turnbull
375	Python 3.1 Release Schedule	Peterson
385	migrating from Subversion to mercurial	Ochtman, Pitou,
392	Python 3.2 Release	Brand L
398	Python 3.8 Release	Brand L
429	Python 3.3 Release	Brand L
428	Python 3.4 Release	Hasbings
438	Transitioning to release-file hosting on PyPI	Krekel, Meyer
449	Removal of the PyPL Mirror Auto	Stuft
464	Removal of the PyPL Mirror API	Stuft
470	Removing External Hosting support on PyPI	Stuft
494	Python 3.6 Release	Deily
512	Migrating from hg.python.org to Github	Cannon

AARIS MALIK FYIT - 122

527	Removing Under Used File on PyPI	StuFFT
541	Package Index Name Retention	Langa
545	Python documentation Translations	Palard
581	using Github Issues for Cpython	Maniatta
53000	Python 3000	GVR
3002	procedure for Backwards - Incompatible changes	Bethard
3003	procedure for Backwards -	GVR Noller.
3009	Things that will not change	
9	in Python 3000	Brand L
5100	Miscellaneous python 3.0 plans	Cannon
8001	Python Governance voting process	Cannon
	Deferred PEPs	
213	Attribute Access Handlers	Prescod
219	Stackless Python	McMillan
222	Web library Enhancements	Kuchling
232	Python online Help	Prescod
262	Optimized Access to module Namespace	Hylton
267	A Database of Installed Python	Kuchling
269	Pgen module for python	Riehl
280	optimizing access to globals	GVR
286	Enhanced Argument Tuples	Von Louis
312	Simple Implicit Lambda	Suzi
316	Programming by python	Way
337	Logging usage	Dubner
368	Standard image protocol and class	mastromarino
400	Deprecate coders.	Stinner
403	General purpose decorator clause	Loughlan
407	New release cycle and introducing	Pitrou

419	Protecting Clean up statements from interruptions	Colomietz
423	Naming conventions and recipes	Bryon
444	Python web 3 interface	CM Donough
447	Add --getdescriptor -- Method to metaclass	Oussoren
491	The Wheel Binary 1.9	Holth
499	Python - m Foo should bind	Simpson
501	General purpose string interpolation	Coghlan
505	None - aware operators	Haase, Dower
532	A circuit breaking protocol	Coghlan, Haase
533	Deterministic Clean up for iterators	Smith
535	Rich comparison chaining	Coghlan
534	Improved Errors	Orsava
536	Final Grammar for String	Angeven
547	Running extension modules	pick
556	Threaded garbage collection	Pitrou
568	Generator - Sensitivity for context variables	Smith
5688	overloading, interfaces	Eby
3124	standard daemon process library	Finnéy
3145	statement local namespace	Coghlan

Abandoned, withdrawn, Rejected

PEP	Title	Author
3	Guidelines for Handling Reports	Hylton
9	Sample plain text PEP	Warsaw
42	Feature Requests	Hylton
102	Doing python micro Releases	GVR
103	Collecting information	Bray thonan
204	Range literals	Wouters
206	Python Advanced	Kuchling
209	multi-dimensional Arrays	Barrett,
210	Decoupling Loop	Ascher
211	Adding a New Outer	Wilson
212	Loop Counter Iteration	Yee
215	String Interpolation	Kamp
216	Docstring format	Zadka
220	Coroutines, Generators	Mc millan
224	Attribute Docstrings.	Lemburg
225	Elementwise operators.	Zhu,
228	Reworking Python	Zadka
231	-- Final attr -- ()	Warsaw
239	Adding a Rational Type	Zadka
240	Adding Literal Python	Zadka
242	Numeric kinds	Dubois
243	module Repository	Reif shneider
248	The directive	Von Louis
248	Python Interface Syntax	Pelletier
248	Object Adaption	Martelli
2584	Docstring Processing	Goodger
256	Docstring Design	GVR
265	Building Dictionaries	Griffin

270	uniq method list objects	Petrone
271	Prefixing sys.path	Giacomitti
275	Switching on multiple	Lemburg
276	Simple Iterator for ints	Althoff
281	Loop counter range & xrange	Hettland
284	Integer for - loops	Ewing
288	Generators Attributes	Hettinger
294	Type name in the type modules	Tirosh
295	interpretation of multiline	Koltsov
296	Adding a bytes object Types	Gilbert
297	Support for system upgrades	Lemburg
298	To Locker Buffer Interface	Helter
299	Special -- main -- ()	Epler
303	Extend distmod ()	Bellman
304	controlling Generation	Montanaro
306	How to change Grammar	Peterjon
310	Reliable Acquisition / Release pairs	Hudson, moore
313	Adding Roman Numerical	Meyer
315	Enhanced while loop	Carroll
317	Eliminate Implicit	Taschuk
319	Python Block	Pelletier
321	Date / Time passing	Kuchling
325	Resource Release	Pedroni
326	A case for To paid Bottom values	Reedy
329	Treating Builtins as constants in the standard	Hettinger

AARIZ MALIK PYIT - 12.2

Numerical Index.

PEP	PEP Title	Authors
1	PEP purpose and Guidelines	Goodger
2	procedure for adding New modules	Fassen
3	Guidelines for Handling Bug reports	Hylton
4	Depreciation of standard modules	Cannon.
5	Guidelines for language	Prescod
6	Bug Fix Releases	Aahz, Baxter
7	Style Guide for C Code	CVR
8	Style Guide for python code	CVR.
9	Sample plain text PEP Template	Warsaw
10	Voting Guidelines	Warsaw
11	Removing Support for limited used platforms	Cannon.
12	Sample restructured Text PEP Template	Goodger
13	Python languages	community
20	The Zen of Python	Peters
42	Features Requests	Hylton
100	Python unicode	Lemburg
101	Doing python Releases 101	Warsaw, CVR
102	Doing python micro Releases	Baxter.

103	collecting information about gift	Broymman Drake
160	Python 1.6 Release	Hylton
200	Python 2.0 Release	Warsaw
201	lockstep iteration	Warsaw
202	List comprehensions	Wouters
2003	Augmented Assignments	Wouters
204	Range literals	Drake
205	Weak Reference	Kuchling
206	Python: Advanced Library	GR, Ascha
207	Rich Comparisons	Lemberg
208	Reworking the coercion	Oliphant
209	Multi-dimensional Array	Ascherr
210	Decompiling loop	Wilson
211	Adding a new operator	Kamp
212	loop counter	Prescod
213	Attribute Access	Warsaw
214	Extended print statement	Yee
215	String Interpolation	Zedka
216	Docstring format	Wilson.
217	Display Hook for use	Heitinger
218	Adding a Built-In Type	McMillan
219	stackless python	McMillan
220	Generators	Wouters
221	Import As	Peters.
222	Web Library Enhancements	B

AARTS MALIK FYIT - 122

223	change the meaning	Peters
224	Array Doubling.	Lemburg
225	Elementwise / Objectwise	
	Opeters.	Hylton
226	Python 2.1 release	zhu.
227	Statically Nested	Hylton
228	Reworking Python's	
	Numerica model	Zadka
229	Using Distutils	Kuchling
230	Warning framework	GVR
231	-- Final attr -- ()	Warsaw
232	function Attributes	Warsaw
233	Python Help outline	Prescod
234	Iterators	Yee, GVR
235	Import on case- insensitive platforms	Peters.
236	Back to the -- future	Peters.
237	Unifying long integers	Zadka, GVR
238	changing the Division Operators.	Zadka, GVR
239	Adding a Rational	Craig
240	Adding a Rational literal	Zadka
241	Metadata	Kuchling
242	Numeric kinds	Dubois
243	Module Repository	
	Upload	Reifschneider
244	The directive statements	Pelletier.

Reserved Pep Numbers

PEPs	Title	Authors
801	RESERVED	Warsaw

PEP Types key

① I - Informational PEP

② P - Process PEP

③ Standard Track PEP - S

PEP status key

A - Accepted (Standard Track only) or
Active proposal

D - Deferral proposal

F - Find proposal

P - Provisional proposal

R - Rejected proposal

S - superseded proposal

W - withdrawn proposal

AARZ MALIK PYIT - 122

Authors / Owners

Name	Email Address
Aahz	aahzat.pythoncraft.com
Aasland, Erlend Egeberg	@protonmail.com
Ahlstorm, James C	@Interest.com
Althoff, Kevin	@i2.com
Community, The PythonCore Team	@semi-retired.com
Angelico, Chris Philipp	@gmail.com
Ascherr, David	phil.angerer@gmail.com
Ascherr, Ammar	david@activestate.com
Astrand, Peter	ammar@ammarasharr.com
Banks, Carl	python dev@.com
Barker, Christopher	chris.barker@noaa.gov
Darrett, Paul	barret at, stsci.edu
Bechel, Paul Stefan	Stefan - m@.de
Bellman, Thomas	bellman + ped - divmod@.se
John Alexander	Jonhanatneggie.net
Bendersky, Eli	alexander@gmail.com
Benfield, Cory	eliben@gmail.com
Bethard, Steven	cory@lukasa.co.uk
Cory	Steven.Bethard@gmail.com
Steven	cory@lukasa.co.uk
Altis, Kevin	Steven.Bethard@gmail.com

AARIS MALIK FYIT-122

Bellaman, stefan

Stefan - ml@

behnel.de

Bellman, Thomas

liu.se

Belmonte, John

John at neggle.net

Belopolsky,

alexander.belopolsky

Alexander

@ gmail.com

Bendersky, Eli

alexander @ gmail.com

Benfield, cory

eliben @ gmail.com

Bethard, steven

cory at lukasa.com uk

steven.bethard @

gmail.com

Bidoul, stephane

Stephane @ gmail.com

Bleaney, Graham

gbleaney at gmail.com

Bonte, Erik De

erikd at microsoft.com

Borini, stefano

Brandl, Georg

georg @ python.org

Bray, Erik M

Britton, Gerald

gerald.britton at gmail

.com

Broylman, oleg

Bryon, Benoit

phd at phdru.name

benoit at marmelune

.net

Buchner, Brandt

brandt at python.org

Cannon, Brett

brad at python.org

Cappos, Justin

Carey, Joe

j.cappas at nyu.edu

Carroll, Josiab

Joe @ python.org

Chisholm, malt

jcarlson at uci.edu

Bendersky, Alexander

malt - python at

@ .org

AARIS MALIK FYIT - 122

chung, Tzu-ping
chung, Nick
cole, Dave
collins, Robert
colomiets, paul
marino

uranusjr@gmail.com
uranusjr@gmail.com
n.waghlan@gmail.com
djic@objectgmail.com
rhtcollins@hp.com
paul at colomiets name
marinoj.89@gmail.com

Craig, Christopher A.

Aprano, Steven

python - pepat
ccraig.org
steve at pearwood
info
mail@khrushaldas.
in

Dasg, Kushal

Deliy, ned
Delaney, Tim
DeLong, Lois Anne
Demeyers, Jerroen

nad@gmail.org
timothy@gmail.org
Lad 278 at nyu.edu
j.demeyer@rtrgent.
be
Jack died@gmail.
com

Diederich, Jack

Dorwald, Walter

Walter at living
logic.com
fred at fdrake.net
Jubnem at mindless
com

Drake, Fred, L. Jr

Dubner, Michael P.

Dubois, Paul F

paul at pfDubois.com

AARIS MALIK FYIT - 122

Eby, Phillip J.

pje at telecommunity.
com

Eby, P. J.

pje at telecommunity.
com

Elinat, Jal

tal@gmail.com

Elizondo Eddie

eduardo.elizond

orucdat@gmail.com

Elliot, Micah

mde@tracos.org

Epler, Jeff

Jepler@.net

Eppstein, David

eppstein at .io

Eustace, Sebastien

Sebastien at
eustace.io

Evans, Clark E.

Cce at clark.evans.
com

Ewing, Greg

greg-ewing at
cantelary.ac.nz

Fassen, Martijn

Fassen at nfrac.com

Finney, Ben

ben.python@.edu

Food, Michael

michael at python.org

Furman, Ethan

ethan at stoneleaf
.us

Gabor, Bernat

gobar@gmail.com

Galindo, Pablo

pablo.sal. at python.org

Ganssle, Paul

alex.gajnor at
gmail.com

Grady, Prady M

pradyunsg at
gmail.com

Gerlach, C.A.M

cam.gerlach at
gerlach.com