

Python from Scratch

**Programming for absolute beginners
with Python**

Nilo Ney Coutinho Menezes

Authorized English translation of the Portuguese edition of *Introdução à Programação com Python 4ª Edição*, ISBN 97875228869 © 2024 Novatec Editora Ltda. This translation is published and sold by permission of Novatec Editora Ltda, the owner of all rights to publish and sell the same. All rights reserved.

This edition is published by LOGIKRAFT SRL under exclusive license from the author, Nilo Ney Coutinho Menezes, with the express authorization of Novatec Editora Ltda, the original publisher and rights holder of the Portuguese edition.

© 2025 Nilo Ney Coutinho Menezes – LOGIKRAFT SRL – Novatec Editora Ltda. All rights reserved.

No part of this publication may be reproduced, stored or transmitted, in any form or by means electronic, mechanical, photocopying, or otherwise, without prior written permission of the publisher.

Publisher: Rubens Prates

Translator: Nilo Ney Coutinho Menezes

Proofreader: Ariane L. Smith and Kelsey Yurek

Cover designer: Olinart

Cover illustration: Olinart

Paperback: 978-85-7522-949-1

Hardcover: 978-85-7522-950-7

Ebook: 978-85-7522-951-4

Novatec Editora Ltda.

Rua Luís Antônio dos Santos 110

02460-000 – São Paulo, SP – Brasil

Tel.: +55 11 2959-6529

Email: novatec@novatec.com.br

Site: <https://novatec.com.br>

Twitter: twitter.com/novateceditora

Facebook: facebook.com/novatec

LinkedIn: <https://linkedin.com/company/novatec-editora/>

LogiKraft SRL

Rue de la Grande Campagne, 40

7340 – Colfontaine

Belgium

Email: contact@logikraft.be

Website: <https://logikraft.be>

DISCLAIMER

Every effort has been made to ensure that the information provided in this book is as accurate and complete as possible. However, such information is provided “as is” and without warranty of any kind, either expressed or implied. The author, publisher, distributors, and any entity directly or indirectly involved in its dealings assume no liability whatsoever for any loss or damage, direct or indirect, arising from the information contained in this book.

Table of Contents

Acknowledgments	11
Preface	12
Introduction	14
 Chapter 1: Motivation.....	 17
1.1 Do you want to learn how to program?	18
1.2 What is your patience level like?	18
1.3 How long do you intend to study?	19
1.4 To program for what?	20
1.4.1 Writing web pages	20
1.4.2 Set your clock.....	20
1.4.3 Learn to use maps.....	20
1.4.4 Show your friends that you know how to program	21
1.4.5 Seem weird	21
1.4.6 Better understand how your computer works.....	21
1.4.7 Cook.....	21
1.4.8 Save the world	22
1.4.9 Free software	22
1.4.10 Get rich	23
1.5 What do you do if you don't know how to solve something?	23
1.6 How to ask for help	24
1.6.1 Common mistakes when trying to learn to program.....	24
1.6.2 When will you have finished learning?.....	25
1.6.3 What is really important?	25
1.6.4 What to avoid?	26
1.7 Why Python?	26
 Chapter 2: Preparing the environment.....	 29
2.1 Python installation.....	29
2.1.1 Installation on Windows	30

2.1.2 Installation on Linux	34
2.1.3 Installation on macOS	35
2.2 Using the Python interpreter	35
2.3 Editing files	36
2.4 Visual Studio Code.....	40
2.5 Be careful when typing your programs	50
2.6 First programs.....	50
2.7 Concepts of variable and assignment	53
2.8 A quick review of mathematical concepts	56
2.8.1 Numbers	56
2.8.2 Properties of addition and subtraction.....	57
2.8.3 Division properties.....	57
2.8.4 Multiplication properties.....	59
2.8.5 Properties of the remainder	60
2.8.6 Rule of three and proportion	62
2.8.7 Percentage.....	62
2.8.8 Arithmetic mean.....	63
2.8.9 Weighted arithmetic mean.....	64
Chapter 3: Variables and data entry	66
3.1 Variable names.....	66
3.2 Numeric variables	67
3.2.1 Representation of numeric values	68
3.3 Boolean variables	70
3.3.1 Relational operators	70
3.3.2 Boolean operators.....	72
3.3.2.1 not operator.....	73
3.3.2.2 and operator	73
3.3.2.3 or operator	73
3.3.3 Boolean expressions.....	74
3.4 String variables	76
3.4.1 String operations	77
3.4.1.1 Concatenation	77
3.4.1.2 Composition.....	78
3.4.1.3 String slicing.....	81
3.5 Sequences and time.....	82
3.6 Tracing.....	83
3.7 Data entry (inputs).....	84
3.7.1 Data entry conversion.....	85
3.7.2 Common Mistakes	86
Chapter 4: Conditions	89
4.1 if.....	89
4.1.1 Income tax calculation example	92
4.1.2 Example of calculating a cell phone bill	93
4.2 else.....	95
4.3 Nested structures.....	96
4.4 elif.....	98

4.5 Inversion of conditions.....	99
4.6 Note of caution when comparing values.....	101
Chapter 5: Loops.....	103
5.1 Counters	106
5.2 Accumulators.....	108
5.2.1 Augmented assignments.....	109
5.3 Interrupting the repetition.....	110
5.4 Nested loops.....	112
5.5 F-Strings	114
Chapter 6: Lists, dictionaries, tuples, and sets	116
6.1 Working with indexes.....	118
6.2 Copying and slicing lists.....	118
6.3 List size	120
6.4 Adding elements.....	121
6.5 Removing elements from the list	123
6.6 Search.....	123
6.7 Using for	124
6.8 Range	125
6.9 Enumerate	126
6.10 Operations with lists.....	127
6.11 Applications	127
6.12 Lists with strings.....	129
6.13 Lists within lists.....	129
6.14 Sorting	131
6.15 Using lists as queues.....	134
6.16 Using lists as stacks	136
6.17 Dictionaries.....	137
6.18 Dictionaries with lists.....	140
6.19 Default value dictionaries	142
6.20 Tuples.....	142
6.21 Sets (set)	146
6.21.1 Union	147
6.21.2 Intersection	147
6.21.3 Difference	148
6.21.4 Symmetric difference	148
6.21.5 Other operations.....	149
6.21.6 When to use sets.....	150
6.22 What data structure to use?	151
Chapter 7: Working with Strings	152
7.1 Partial string verification.....	153
7.2 Counting.....	154
7.3 String search.....	154
7.4 Positioning Strings	157
7.5 Breaking or separating strings.....	158
7.6 String replacement	158

7.7 Removing whitespace characters	159
7.8 Validation by content type	159
7.9 String formatting.....	161
7.9.1 Formatting numbers	163
7.10 Hangman.....	166

Chapter 8: Functions.....170

8.1 Local and global variables	175
8.2 Recursive functions.....	176
8.3 Validation	178
8.4 Optional parameters	179
8.5 Naming parameters.....	181
8.6 Functions as a parameter	183
8.7 Packing and unpacking parameters.....	184
8.8 Unpacking parameters.....	184
8.9 Lambda functions	185
8.10 Exceptions	186
8.11 Modules.....	190
8.12 Random numbers.....	192
8.13 Type function.....	195
8.14 List comprehensions	197
8.15 Generators.....	200
8.16 Generator comprehensions.....	203
8.17 Dict comprehensions.....	204
8.18 Set comprehensions	204
8.19 Map and zip	204
8.20 Reduce	206
8.21 Filter.....	208
8.22 Partial application of functions	209
8.23 Mathematical functions.....	210
8.23.1 Walrus operator	212

Chapter 9: Files215

9.1 Opening files.....	215
9.2 The command line	218
9.3 Basic commands in Windows	220
9.4 Basic commands in Linux/macOS.....	222
9.5 Command line parameters	224
9.6 Generating files.....	224
9.7 Reading and writing.....	225
9.8 Processing a file	226
9.9 HTML generation	230
9.10 Files and directories.....	233
9.11 A bit about time.....	237
9.12 Use of paths.....	240
9.13 Pathlib.....	241
9.14 Visiting all subdirectories recursively.....	242
9.15 Date and time	243

9.15.1 Time zones	245
9.16 JSON files	246
9.17 Binary files.....	248
Chapter 10: Classes and objects.....	257
10.1 Before objects.....	257
10.2 Objects as a representation of the real world	259
10.3 Passing parameters.....	262
10.4 A bank example	266
10.5 Inheritance	270
10.6 Developing a class to control lists	272
10.7 Class attributes and methods.....	283
10.8 Revisiting the phonebook	285
10.9 Creating exceptions.....	289
Chapter 11: Database.....	291
11.1 Basic concepts.....	291
11.2 SQL.....	293
11.3 Python & SQLite	293
11.4 Querying records	298
11.5 Updating records.....	300
11.6 Deleting records	302
11.7 Simplifying access without cursors.....	302
11.8 Accessing fields as in a dictionary.....	303
11.9 Generating a primary key	303
11.10 Changing the table.....	305
11.11 Grouping data.....	305
11.12 Working with dates	308
11.13 Keys and relations	311
11.14 Converting the phonebook to use a database	313
Chapter 12: Patterns.....	322
12.1 Pattern recognition	322
12.2 Regular expressions	329
12.2.1 Finding sequences	333
12.2.2 Greedy capture.....	335
12.2.3 Compiling regular expressions	336
12.2.4 Capturing multiple groups and subgroups	337
12.2.5 When to use regular expressions.....	338
12.3 Structural pattern-matching.....	339
Chapter 13: Graphical interface.....	344
13.1 A first program	345
13.2 Counting clicks.....	347
13.3 Using classes	348
13.4 Adding counters	350
13.5 Entering data.....	351

13.6 Drawing.....	354
13.6.1 Drawing a cursor with lines.....	356
13.6.2 Drawing lines.....	358
13.6.3 More tools.....	361
13.6.4 Cleaning and undoing.....	362
13.6.5 Colors.....	362
13.7 A website database.....	368
13.7.1 Site class.....	369
13.7.2 The home screen.....	371
13.7.3 The details screen.....	374
13.7.5 Integrating with the program.....	379
13.7.6 Completing the app.....	381
Chapter 14: Next steps.....	387
14.1 Functional programming.....	387
14.2 Algorithms.....	388
14.3 Games.....	388
14.4 Object orientation.....	389
14.5 Database.....	389
14.6 Web systems.....	389
14.7 Data science and artificial intelligence.....	390
14.8 Other Python libraries.....	390
14.9 Mailing lists.....	391
Appendix A: Error messages.....	392
A.1 SyntaxError.....	392
A.2 IndentationError.....	393
A.3 KeyError.....	393
A.4 NameError.....	394
A.5 ValueError.....	394
A.6 TypeError.....	395
A.7 IndexError.....	395
A.8 TabError.....	396
Appendix B: Bitwise operators.....	397
B.1 Displacements.....	397
B.2 Bitwise operators.....	399
References.....	402
Index.....	403

Symbols

%
 remainder of division 52
 - 51, 52
 "" 115
 " 27, 50
 """" 115
 () 50
 [:] 81
 * 51, 52
 ** 51, 52
 / 51, 52
 // 51, 52
 & 399
 # 53
 % 52
 + 52
 += 110
 < 71
 << 397, 400
 <= 71
 -= 110
 := 212, 213, 214
 != 71
 **= 110
 *= 110
 // = 110
 /= 110
 = 53
 ≠ 71
 == 71
 > 71
 >= 71
 >> 398
 >>> 51
 | 400
 ~ 400
 ≤ 71
 ≥ 71
 @classmethod 284
 %d 78
 __eq__ 277, 278, 285
 __exit__ 297
 \f 78, 333
 __ge__ 277
 __getitem__ 274
 __gt__ 277
 __init__ 260, 261, 271
 __iter__ 274
 __le__ 277
 __len__ 274
 __lt__ 277, 278, 285

__main__ 251
 \n 115, 161
 __name__ 251
 __neq__ 277
 __new__ 261
 \r 161, 333
 __repr__ 277, 371
 %s 78
 __setitem__ 274
 __str__ 276, 285, 370
 \t 161, 333
 \v 333

A

abs 211
 abstraction 263
 accumulators 108
 addition 57
 and 72
 inverted 99
 and operator 73
 arithmetic mean 63, 109
 assert 256
 assignment 53
 attributes 259, 266

B

base 2 70
 conversion 69
 base 8 70
 base 10
 conversion 68
 base 16 70
 batched 251
 batteries included 26
 big endian 255
 bin 398
 binary 70
 conversion 69
 binary operators 72
 bindings 26
 bitwise operators 397
 BMP 252
 book web site 15
 boolean expressions 74
 boolean operators 72
 boolean variables 70
 break 110, 125, 171
 Bubble Sort 131
 bytes 248

C

Carriage Return 161, 217, 250, 333

case 340

if 343

case sensitive 50

cat 222, 223

cd 220, 222

ChatGPT 26

chdir 220

chr 251

class 259

class attribute 283, 284

classes 257

clear 222

close 216

closing 297

cls 220, 284

command line 218

comments 53, 71

commit. *See* databases:commit

concatenation 77

construtor 260, 261

contacts 16

book web site 16

email 16

Telegram 16

contextlib 297

continue 168

copy 220

cosine 211

counter 106

cp 222, 223

CPython 27

CR. *See* Carriage Return

CRC 252

CRUD 291

cursor 294, 317

cyclic redundancy check. *See* CRC

D

database management programs 291

databases 291, 294

close 295

commit 301, 317

connection 294

creating 293

execute 295

executemany 296, 302

executescript 315

fetchall 296

inserting 294

lastrowid 317

query 295

rollback 301

rowcount 301

row_factory 303, 310, 316

transactions 295

datetime 243

date 244, 310

datetime 243

time 243

timedelta 245

datetime.date

strftime 310

datetime.datetime

fromisoformat 244

isoformat 244

isoweekday 244

time zones 245

datetime.timedelta 310

DBMS 291

decorators 277

def 170

del 220

dictionaries 139

with lists 123

dict comprehensions 204

dictionaries 137

default value 142

del 139

get 142

in 138

items 141

key 137

key order 138

keys 139

values 139

dir 220, 221

dividend 57

division 57

divisor 57

divmod 211

docstring 256

E

elif 98

else 95, 190

nested 96

try 190

email for questions 15

encapsulation 262

enumerate 126

errata 16

except 186, 289

exception 187

Exception 289
 exceptions 186, 188
 creating new types of 289
 except 186
 finally 187, 190
 raise 188
 try 186
 exclusive-or 401
 exponentiation 52

F

factorial 173
 False 70
 fetchone 295. *See* databases:fetchone
 Fibonacci sequence 177
 fields 292
 FIFO 134
 files
 close 217
 end of line 217
 open 215, 216, 231
 open modes 215
 path 219, 240
 readlines 217, 225
 write 216, 225
 filter 208
 finally 187, 189
 floating point 69
 for 124
 else 125
 in 124
 list comprehensions 197
 format 80
 from import 191
 f-string 80, 163
 [^] 114
 < 114
 > 114
 debug 214
 functions 170
 functools 278
 partial 328

G

generator comprehensions 203
 generators 200
 getters 282, 283
 global 176, 349
 global variables 175
 Graphical User Interfaces 344
 groups 16
 GUI. *See* Graphical User Interfaces

H

hash 140
 help 256
 hexadecimal 70
 HTML 230

I

IndentationError 393
 IDLE 35
 if 89
 immediate 167
 list comprehensions 198
 nested 96
 import 190, 267
 in
 for 124
 list comprehensions 198
 lists 154
 index 116
 IndexError 77, 186, 188, 395
 indexes 118
 infinite loop 105
 inheritance 270, 272, 275, 289
 input 84
 common errors 86
 float 85
 int 85
 value conversion 85
 Installation on Linux 34
 Installation on MacOS 35
 Installation on Windows 30
 instance 259
 interface 266
 interpreter 27
 is 185, 285
 isinstance 195
 ISO 8601 309
 ISO 8859-1 249
 iter 200
 itertools 251
 batched 251

J

json
 dump 247
 dumps 248
 load 246
 loads 248
 JSON 246

K

key 311
 KeyError 138, 393

L

lambda 205
 lambda functions 185
 len 76
 LF 217, 250. *See* Line Feed
 LIFO 136
 Line Feed 161, 217
 list 123, 152
 list comprehensions 197
 for 197
 if 198
 in 198
 walrus (:=) 214
 lists 116
 [$\backslash$:] 119
 append 121
 copies 118
 empty 116
 extend 122
 in 154
 len 120
 of strings 129
 pop 135, 136
 searching 124
 slices 119
 sort 134
 strings 129
 with strings 129
 little endian 255
 local variable 175
 loops 103
 ls 222, 223

M

map 204, 207
 match 339
 *_ 341
 | 340
 classes and objects 343
 dictionaries 342
 if 343
 lists and tuples 341
 order 342
 packing 342
 type 341
 unpacking 342
 math 210
 ceil 210

 cos 211
 degrees 211
 floor 210
 modf 211
 pi 211
 radians 212
 sin 211
 tan 211
 trunc 211
 max 174
 md 220
 method 262, 263
 min 174
 mkdir 222
 move 220, 222
 multiplication 59
 mv 222, 223

N

NameError 394
 name mangling 280, 281, 283
 next 200, 201
 None 142, 172, 179, 185, 276, 285
 not 72
 inverted 99
 not operator 72, 73
 numbers
 even 60
 integer 56
 irrational 56
 natural 56
 odd 60
 primes 113
 rational 56
 real 56
 numerical variables 67

O

object 257, 259
 Object Relational Mapping 313
 octal 70
 open. *See* files:open
 OperationalError 295
 operators 52
 optional parameters 180
 or 72
 inverted 99
 ord 251
 ORM. *See* Object Relational Mapping
 or operator 73
 os
 chdir 234

- getcwd 233
- makedirs 235
- mkdir 235
- remove 235
- rename 235
- rmdir 235
- walk 242

- os.path
 - abspath 240
 - basename 240
 - dirname 240
 - exists 236
 - getatime 237
 - getctime 237
 - getmtime 237
 - getsize 237
 - isdir 236
 - isfile 236, 315
 - join 240
 - splitdrive 240

P

- partial 209, 210, 328
- pass 289
- Path. *See* pathlib.Path
- pathlib 241
 - Path 241
- pathlib.Path
 - parts 241
 - suffix 242
 - suffixes 242
 - walk 243
- paths. *See* files:path
- pattern matching 322
- percentages 62
- pi 211
- pickle 288
- PNG 252
- pound 53
- primary key 303, 311
- print 27, 36, 225
 - end 225
- priorities
 - boolean operators 74
 - mathematical operations 52
- property 280, 281, 282. *See* property
- proportion 62
- pseudolanguage 28
- Python installation 29
- Python interpreter 35

Q

- queue 134
- quotient 57

R

- raise 188, 189
- random 192
 - randint 192
 - random 193
 - sample 193
 - shuffle 193
 - uniform 193
- random numbers. *See* random
- range 123, 125, 126
- raw strings 331
- rd 220
- readlines 217
- record 292
- recursive function 176
- redirecting output 223
- reduce 206, 208
- Regex. *See* regular expressions
- regular expressions 322, 329
 - ?? 330, 331, 333, 334, 335, 336
 - \1 337
 - character classes 330
 - compile 336
 - \d 331
 - \D 332
 - findall 330
 - finditer 336
 - greedy match/capture 335
 - groups 333, 337
 - IGNORECASE 335
 - match 330, 334, 337
 - Match 330
 - {m,n} 336
 - naming groups 337
 - negating a class with ^ 332
 - non capturing group 334
 - non-greedy capture/match 336
 - ?P 337
 - raw strings 331
 - RE.IGNORECASE 336
 - \s 332, 334
 - \S 333
 - search 330
 - \w 332
 - \W 332
- relational operators 70
- remainder of division 52
- reminder 57, 60

- ren 221
- repr 277
- return 171, 172, 185
- RGB 255, 363
- rmdir 220, 222
- rollback. *See* databases:rollback
- round 210
- rule of three 62
- running your program 38

S

- saving a file 38
- scientific notation 165
- scope 175
- self 260, 261, 279
- set 146
 - != 149, 150
 - contains 149
 - difference 148
 - empty 150
 - in 146
 - intersection 147
 - len 147
 - number of elements 147
 - symmetric difference 148
 - union 147
- set comprehensions 204
- sets 146
- setter 280
- setters 282, 283
- short-circuit 378
- sine 211
- singleton 185
- sorted 134
- sorting 131
- SQL 293
 - alter table 305
 - and 299
 - as 307
 - autoincrement 303, 313
 - avg 306
 - count 306, 316
 - create table 294
 - delete 302
 - desc 304
 - from 312
 - group by 305, 307
 - having 308
 - insert 294
 - max 306
 - min 306
 - not 299
 - or 299
 - order by 304, 307
 - primary key 303, 313
 - select 295
 - set 300
 - sum 306
 - update 300
 - where 298, 300
- SQLInjection 299
- SQLite 293, 303
- stack 136, 190
- state 257, 258, 259, 324
- staticmethod 279, 284
- StopIteration 200, 201
- string
 - center 157
 - count 154
 - endswith 153
 - find 154
 - format 161
 - f-strings 114
 - index 155
 - isalnum 160
 - isalpha 160
 - isdigit 160
 - islower 161
 - isprintable 161, 250
 - isspace 161
 - isupper 161
 - join 152
 - ljust 157
 - lower 153
 - lstrip 159
 - replace 158
 - rfind 155
 - rindex 155
 - rjust 157
 - rstrip 159
 - split 158
 - splitlines 158
 - startswith 153
 - strip 159
 - upper 153
- string composition 78
- string operations 77
- string slicing 81
- string variables 76
- Structured Query Language. *See* SQL
- subclass 270, 272
- subtraction 57
- sum 174
- super 271, 349
- superclass 270, 272
- SVG 368

SyntaxError 181, 212, 392
 sys
 argv 224

T

TAB 161
 TabError 396
 tables 292
 tangent 211
 Tcl 346
 Telegram 15, 24
 time
 ctime 237, 238
 gmtime 237
 localtime 238
 strftime 238
 time 238
 TimSort 134
 tk
 BOTH 378
 CENTER 373
 END 379
 Menu 382
 N 355
 NSEW 355
 S 355
 StringVar 375
 Text 379
 VERTICAL 373
 W 378
 X 356
 Y 356
 tkinter 344
 <Motion> 356
 askopenfilename 383
 askquestion 382
 asksaveasfilename 383
 Button
 command 346
 Button-1 359
 ButtonRelease-1 359
 Canvas 356
 bind 356
 create_line 358
 delete 362
 tags 359
 colorchooser 362
 Combobox 376
 Double-Button-1 380
 expand 356
 fill 364

grid 355
 column 355
 row 355
 ipadx 353
 ipady 353
 outline 364
 pack 346, 353, 355
 fill 356
 padding 352
 padx 353
 pady 353
 showinfo 383
 sticky 355
 StringVar 375
 total_ordering 278
 traceback 190
 triple apostrophes 115
 triple quotes 115
 True 70, 104
 try 186, 188, 289
 else 190
 ttk 344
 Button 346
 Frame 346, 352, 354
 Label 346
 Scrollbar 373
 Treeview 373
 tuples 142
 empty 144
 packing 145
 swapping values 143
 unpacking 145
 with a single element 144
 type 195, 220
 FunctionType 196
 TypeError 77, 143, 182, 189, 273, 395

U

unary operator 72
 Unicode 67
 universal newline support 217
 UserList 274
 UTF-8 67, 231, 249
 uuid 370

V

ValueError 87, 88, 186, 188, 189, 276, 394
 variable names
 naming rules 66
 variables 53
 Visual Studio Code 36, 40

W

- walrus 212
- Walrus operator 213
- weighted average 64
- while 103, 125
 - else 125
- with 217, 295, 297
- write. *See* files, write

X

- xor 401

Y

- yield 201, 316

Z

- zip 205