INDEX

A

aggregating data, 110–114
angle brackets (<>), 5
anonymous functions, 168
APIs (application programming interfaces), 28–30
- API keys, 31
- JSON (JavaScript Object Notation) format, 30–37
- rate limit, 60
- research questions and, 37–41
- URL-based API calls, 29, 48–49

append() function, 73
apply() function, 168
archives, 64–65
arguments, 20
artificial intelligence, 179–180
ascending order, 116
assignment operator (=), 18–19
Atom, 43
attributes, 7

B

backend languages, 14–25
Baumgartner, Jason, 145
beautifulsoup4 library, 47, 68–70
Bird, Steven, 180
Boolean values, 156–157
bots, 103–104
browsers, 10

C

calling functions, 20–21
causation, 117
cd command, 45
cells, 140–142
chaining, 48, 157
change directory command, 45
charts, 124–130
classes, 8
clients, 28
closing tags, 5
code and coding. *See also* Python
- code-sharing platforms, 31
- coding resources, 178
- encode, 60
- libraries, 46–48
- minified, 87
- pseudocoding, 46
- resources, 178–180
- unicode, 60
- unminified, 87

color property, 7
color scale formatting, 132–133
column charts, 125–126
command line interfaces (CLIs), 14–15, 44, 137–138
Command Prompt, 14
comma-separated values, 47
comments, 46
comparison charts, 125–126
composition charts, 127
concatenation, 17
conditional formatting, 131–133
conditionals, 23–25, 56, 116
Cooper, Katherine, 78–79
correlation, 117
Cox, Amanda, 102
crawlers, 80–81
credentials, 31
CSS (Cascading Style Sheets), 6–12
csv library, 47, 49–50, 68

D

data analysis, xviii, 101
- aggregating data, 110–114
- bots, 103–104
- Google Sheets, 104–106, 121–122
- Jupyter Notebook, 136–142
- merging data sets, 117–121
- modifying and formatting data, 106–109
- pandas library, 142–149
- process, 102–103
- resources, 179
- sorting and filtering data, 114–117

data dictionary, 154
data frames, 144
data scraping. *See* web scraping
data types, 16
datetime library, 47

declaring functions, 21
declaring loops, 22–23
defining functions, 21
descending order, 116
describe() function, 160
developer tools, 10–11
dictionaries, 51–53, 73
DictWriter() function, 74
Digital Forensic Research Lab, 103, 176
directories, 44–45
distribution charts, 126
div elements, 5–6
<div> tags, 67, 70–71
documentation, 28
donut charts, 127
Downey, Allen B., 178
dropna() function, 155

E

elements, 5
encode, 60
end tags, 5
engagement metrics, 152
error messages, 30–31
ethics, 80
expressions, 16

F

Facebook, 64–79
filepaths, 44
filtering data, 114–117
find() function, 71–72, 90–91
floats, 16, 96
for loops, 22–23
formatting data, 106–109
formulas, 112–114
frontend languages
    CSS (Cascading Style Sheets), 6–12
    HTML (HyperText Markup Language), 4–6
    JavaScript, 12–13
functions, 20–22
    append(), 73
    apply(), 168
    describe(), 160
    DictWriter(), 74
    dropna(), 155
    find(), 71–72, 90–91
    get_text(), 71–72
    head(), 146
    json.load(), 51
    lambda, 168–169
    len(), 20, 148
    loads(), 49
    make_csv(), 59–60
    mean(), 160
    open(), 49–50
    print(), 20, 148
    reusable, 58–61
    set_index(), 172
    sleep(), 96
    sort_values(), 158
    tail(), 147
    writeheader(), 74
    writer(), 50
    writerow(), 50

G

General Data Protection Regulation (GDPR), 64
get_text() function, 71–72
Google
    Chrome, 10
    Sheets, 104–106, 121–122, 128–133

H

head() function, 146
Heisler, Sofia, 178
hexadecimal colors, 7
home pages, 4
HTML (HyperText Markup Language), 4–6

I

IDs, 8
if clauses, 23–25
=iferror() formula, 120
iloc[] method, 149
indentation, 5–6
inheritance of styles, 7
inline CSS, 7
integer-location-based indexing, 149
integers, 16, 96
internal style sheets, 8
Internet Archive, 145
IPython Notebooks, 136
iteration, 22–23

J

JavaScript, 12–13
joining data sets, 117–121
JSON (JavaScript Object Notation) format, 30–37

json library, 47, 49
JSON objects, 34
json.load() function, 51
Jupyter Notebook, 136–142

K

keys, 34
key-value pairs, 34
Klein, Ewan, 180

L

lambda functions, 168–169
len()function, 20, 148
libraries, 46–48
  beautifulsoup4 library, 47, 68–70
  csv library, 47, 49–50, 68
  datetime library, 47
  importing, 68
  json library, 47, 49
  matplotlib library, 175–176
  pandas library, 47, 142–149, 165
  pip library, 47–48
  requests library, 47, 49
  scikit-learn library, 180
  third-party, 46
Linder, Lindsey, 78–79
lists, 19–20
loads() function, 49
logical operators, 24–25
loops, 22–23
Loper, Edward, 180
Lytvynenko, Jane, 38

M

machine learning, 179–180
macOS, xxi
make_csv() function, 59–60
matplotlib library, 175–176
McKinney, Wes, 142
mean, 152
mean() function, 160
measures of central tendency, 152–153
median, 152
merging data sets, 117–121
minified code, 87
modifying and formatting data, 106–109

N

Naked Statistics (Wheelan), 179
NaN values, 155–156
natural language processing (NLP), 179
Natural Language Processing with Python (Bird, Klein, and Loper), 180
nested elements, 5
nextPageToken key, 55–57
NLTK (Natural Language Toolkit), 179
null values, 154–156
numbers, 16

O

one-dimensional data sets, 143–144
open() function, 49–50
opening tags, 5
operators, 16, 24–25
overloading a server, 82

P

pagination, 55–57
pandas library, 47, 142–149, 165
panel data, 142
parameters, 29–30, 41
parsing, 69
part parameter, 30
paste special, 115
pie charts, 127
pip command, 68
pip library, 47–48
pivot tables, 110–111
placeholders, 154
plotting data, 175–176
population data, 153
print statements, 15
print()function, 20, 148
prompts, 15
properties, 7
pseudocoding, 46
PyPI (Python Package Index), 47
Python. *See also* code and coding
  dictionaries, 51–53
  downloading and installing, xx–xxi
  resources, 178
  scripts, 43
  standard library, 46–48
  use of, 14

Q

Quealy, Kevin, 102

R

rate limit, 60
raw data, 102
Reddit, 145–149, 152–162

Reinhart, Alex, 179
rendering, 4
requests library, 47, 49
resampling data, 172–175
research questions, 37–41
resources on writing code and data analysis, 178–180
robots exclusion protocol, 80–81
Rocha, Roberto, 80

S

scatterplots, 127–128
scikit-learn library, 180
scraping data. *See* web scraping
scripts, 28–29, 44–46
search parameter, 29
series, 143–144
servers, 4, 82
set_index() function, 172
Silverman, Craig, 38
Singer-Vine, Jeremy, 38, 178
single-color formatting, 131–132
sleep() function, 96
sort_values() function, 158
sorting and filtering data, 114–117, 158–160
spaCy, 179–180
spiders, 80–81
spreadsheets, 49–50, 72–74
start tags, 5
statistical analysis, 179
Statistics Done Wrong (Reinhart), 179
stems, 156
string concatenation operator (+), 17
strings, 16, 17
style attribute, 7–8
style sheets, 9
summarizing data, 157–162
summary data, 102
syntax highlighting, 43

T

tags, 5
tail() function, 147
templates, 57–61
Terminal, 14
terms of service, 82
Think Python, 2nd Edition (Downey), 178
third-party libraries, 46
time periods, 128
time series, 128, 170–172
timestamps, 23
transposing data, 147–148
troubleshooting, xxii–xxiv
Twitter, 163–176
types, 16

U

unicode, 60
unminified code, 87
Unminify, 87
unordered lists, 90
URL-based API calls, 29, 48–49
URLs (uniform resource locators), 4
user-agent, 81

V

values, 19–20, 34
variables, 17–19, 49–50, 57–58, 88–92
virtual environments, 135–138
visualizations, 123
  charts, 124–130
  conditional formatting, 131–133
=vlookup() formula, 117–118
void elements, 5

W

Web Inspector, 10–11
web robots, 80–81
web scraping
  best practices, 80, 94–98
  Facebook, 64–70
  robots exclusion protocol, 80–81
  template, 92–94
  terms of service, 82
  variables, 88–92
  Wikipedia, 83–87
websites, HTML and, 4–6
Wheelan, Charles, 179
Wickham, Hadley, 179
Wikipedia, 83–87
Windows, xxi
writeheader() function, 74
writer() function, 50
writerow() function, 50

Y

YouTube, 31–41