Extracted from:

Python Brain Teasers

Exercise Your Mind

This PDF file contains pages extracted from *Python Brain Teasers*, published by the Pragmatic Bookshelf. For more information or to purchase a paperback or PDF copy, please visit http://www.pragprog.com.

Note: This extract contains some colored text (particularly in code listing). This is available only in online versions of the books. The printed versions are black and white. Pagination might vary between the online and printed versions; the content is otherwise identical.

The Pragmatic Bookshelf
Raleigh, North Carolina

The Pragmatic Programmers

Python Brain Teasers

Exercise Your Mind

Miki Tebeka

edited by Margaret Eldridge

Python Brain Teasers

Exercise Your Mind

Miki Tebeka

The Pragmatic Bookshelf
Raleigh, North Carolina

Pragmatic Bookshelf

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and The Pragmatic Programmers, LLC was aware of a trademark claim, the designations have been printed in initial capital letters or in all capitals. The Pragmatic Starter Kit, The Pragmatic Programmer, Pragmatic Programming, Pragmatic Bookshelf, PragProg and the linking *g* device are trademarks of The Pragmatic Programmers, LLC.

Every precaution was taken in the preparation of this book. However, the publisher assumes no responsibility for errors or omissions, or for damages that may result from the use of information (including program listings) contained herein.

For our complete catalog of hands-on, practical, and Pragmatic content for software developers, please visit *https://pragprog.com*.

The team that produced this book includes:

CEO: Dave Rankin
COO: Janet Furlow
Managing Editor: Tammy Coron
Development Editor: Margaret Eldridge
Copy Editor: Jennifer Whipple
Indexing: Potomac Indexing, LLC
Layout: Gilson Graphics
Founders: Andy Hunt and Dave Thomas

For sales, volume licensing, and support, please contact *support@pragprog.com*.

For international rights, please contact *rights@pragprog.com*.

ISBN-13: 978-1-68050-900-7
Encoded using the finest acid-free high-entropy binary digits.
Book version: P1.0—September 2021

Puzzle 2

A Slice of π

pi.py

```python
π = 355 / 113
print(π)
```

Guess the Output

Try to guess what the output is before moving to the next page.

This code will print: `3.1415929203539825`

There are two surprising things here: one is that `π` is a valid identifier, and the second is that `355 / 113` computes to a `float`.

Let's start with `π` (the Greek letter pi). Python 3 changed the default encoding for source files to UTF-8 and allows Unicode identifiers.

This can be fun to write, but in practice it'll make your coworkers' lives harder. I can easily type `π` in the Vim editor that I use; however, most editors and IDEs will require more effort.

One area where I've found that Unicode identifiers are helpful is when translating mathematical formulas to code. Apart from that, stick to plain old ASCII.

Now for `355 / 113`. Python 3 does the right mathematical division. If you try this code in Python 2, you'll get `3` since Python 2 shows more of its C origins. If you want integer division to return an `int` in Python 3, use the `//` operator (e.g., `355 // 113`). This is handy when calculating indices, which must be whole numbers.

Further Reading

Identifiers and Keywords in the Python Reference
docs.python.org/3/reference/lexical_analysis.html#identifiers

PEP 3120: Using UTF-8 as the Default Source Encoding
python.org/dev/peps/pep-3120/

PEP 263: Defining Python Source Code Encodings
python.org/dev/peps/pep-0263/

Vim Editor
vim.org