

Constants!

in Python

Magic Numbers and Magic Strings

- Programs often use of *literal values* in code such as:
 - Numbers: 3.14159, 110
 - Strings: "aeiouy", "3.8.5"
- When *you* write these literals, *you* know what they mean at the time.
 - When someone else reads them, though, they often feel **magically confusing!**
 - "Why did they choose this **magic number** here?"
 - That *someone else* may be you two months from now after *you've* forgotten!
- There's also a high likelihood these literals are scattered throughout your program. To change it, you need to change it in many places!

Introducing: Named Constants

- Constants are a special kind of variable we declare and initialize with the expectation of *never reassigning its value once the program begins*.
- Constants are named with **ALL_UPPERCASE_IDENTIFIERS**, e.g.:
`VERSION: str = "3.8.5"`
`PI: float = 3.14159`
- Constants are conventionally defined beneath **import** statements, above functions and other constructs
- Good Style: Anywhere you have a magic number in your program, you should replace it with a named constant instead!

Follow-along Lesson: Random Word Generation

1. Create a new Python module in the `lessons` directory named `ls09_constants.py`
2. We'll use the `random` package's `choice` function for choosing one item at random from a sequence, where our sequence is a string.

```
from random import choice
```

3. Then, let's define a couple constants which represent magic strings we'll use in our program:

```
VOWELS: str = "aeiouy"
```

```
CONSONANTS: str = "bcdfghjklmnpqrstvwxyz"
```

4. Finally, let's implement the functions `random_letter` and `random_word` together in the video tutorial!