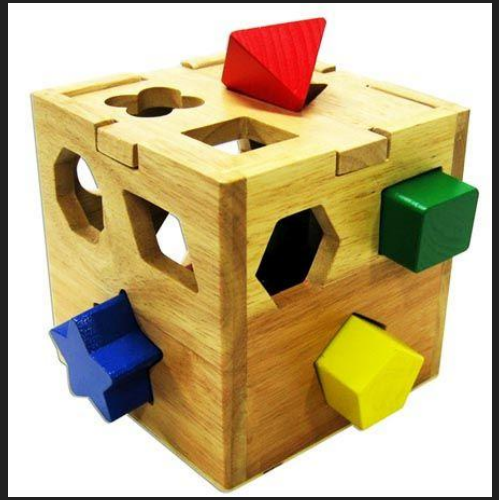
Code for Teachers

A practical approach to programming

Episode 3:
Conditionals, Input, and Numbers

Basic Concepts

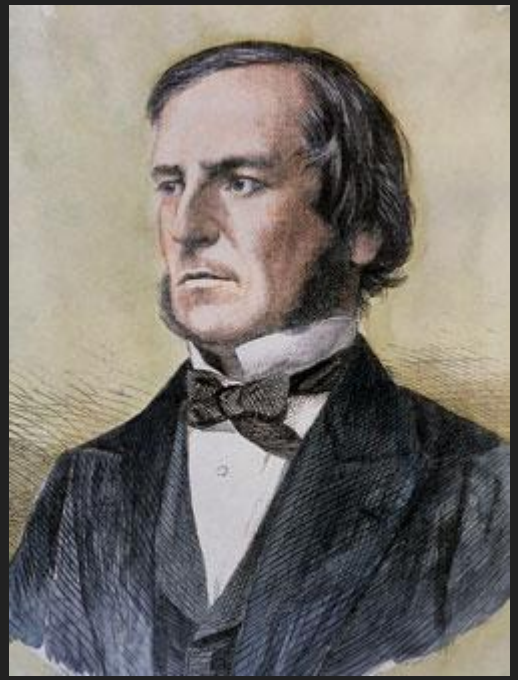
- Information is classified in *data types*
  - We've already seen the `str` data type
- Thinking about what data type we're using is important
  - Enormous relevance outside programming
    - Units in measurement
    - Grammar
  - It's pretty much just that toy from preschool



Basic Concepts, Cont'd

- We can control the flow of our program using *conditional statements*
  - Inside conditionals, lines of code only get executed after a test is passed
  - Tests evaluate to `True` or `False`
  - This is just basic Boolean logic
  - Sounds advanced, but isn't!
  - $P \rightarrow Q; \neg P \rightarrow \neg Q$
  - We can discuss Boolean logic using *truth tables*

George Boole



Boolean Truth Tables

“If it’s sunny, then I’ll go to the beach”

Sunny?	Beach?
`True`	`True`
`False`	`False`

Boolean Truth Tables

“If it’s sunny, then I’ll go to the beach. Otherwise, I’ll go to the movies.”

Sunny?	Beach?	Movies?
`True`	`True`	`False`
`False`	`False`	`True`

Even/Odd Checker

- Let’s make a program that checks whether a user’s number is even or odd
- Things to figure out:
  - How to get input from the user?
  - How do we determine if a number is even or odd?
- Concept > Code
  - Make sure you can articulate the problem and the steps to solve it in English before writing any Python

Section Review

- Information is classified in data types
- We can control the flow of a program with *conditional statements*
- Conditionals use Boolean (`True`/`False`) tests to determine flow
- We can get string input from the user, but might need to change data type
- The `int` data type lets us work with integer numbers

How to keep learning between episodes

- Play with it
  - Nothing surpasses experimentation for raw learning
- Use the documentation
  - docs.python.org
- Ask for help
- Believe in yourself!

theforeverstudent.com

@mttaggart

mttaggart