Lecture 2 - converting values

Video of This

Overview

I have a pre-record video of this is lecture if you want to go back to it. We are going to cover a bunch of stuff - that is in a lot of chapters in the book, 1, 2, 3, some 4, some 5, some 6 and some 7 all in one set of examples.

Also the lecture notes are online in the lect-02 github. https://github.com/Univ-Wyo-Education/F21-1010/class/lect/lect-02

There is a ./conv directory that has a series of steps where you can go back to this and see the code as I develop it.

Demo - of this in browser.

A lot of what happens when you program seems so simple - until you have to learn a non-human language. Programs are formal languages. English is an informal language. For example I can make a sentence that most of you will not understand, at first, but with some explanation I can show that it is using proper English crammer.

"The old man the boat."

In this context the old is a type of person. "man" is to get on board the boat and operate it. It is a verb.

So… The sentence is roughly equivalent to "The old people get on the boat and operate it."

Python is a formal language. It uses a rigorous syntax. As humans we are not used to this.

Topics Covered

1. Files and Directories
2. Editing
3. Operators, `*` is multiply.
4. Other operators like `+`, `-`, `/`, `%`, and unary `-`. There are more.
5. `def` code reusability
6. Float, int and string data types
7. Basic testing
8. Functions - parameters - return values
9. if

10. if / else
11. ‘:’ starts a block
12. Indentation
13. a = a + 1 - not algebra
14. Files
15. Import of files
16. Input
17. Output
18. Formatting of output
19. Patterns in code
20. Fast and Slow Learning

Step 1

Convert from miles to kilometers.

Conversion generally is ((X + k1) * C) + k2

In our case k1 and k2 are 0. So we just get X * C

Demo - lookup conversion from miles to kilometers

```
m4_inclue(conv/step-1.py)
```

Demo - of this as a visualization

Step 2 - Input with error

```
14_inclue(conv/step-2.py)
```

Step 3 - Fixed error / Types

```
m4_inclue(conv/step-3.py)
```

Step 4 - Make a function

```
m4_inclue(conv/step-4.py)
```

Step 5 - Make Reusable Code

step-5.py:

```
m4_inclue(conv/step-5.py)
```

conv/mi_to_km.py:

```
m4_inclue(conv/mi_to_km.py)
```

Copyright