CS1301-Intro to Computing

Day 4

Review – Order of precedence (PENMDASRL)

Precedence	Operator	Explanation
1	()	Parentheses
2	**	Exponentiation
3	-x, +x	Negative, positive
4	*	Multiplication
5	/, //, %	Division, floor division, modulus
6	+	Addition
7	-	Subtraction
8	<, <=, >, >=, ==, !=	Relational
9	not, and, or	Logical

Review

- A function is a named sequence of statements that belong together.
  - Buit-in: from the standard library or as import from an external library
  - User defined: `def function_name(PARAMETERS):`
    `STATEMENTS`
  - Void function: returns the value `None`
  - Value-returning function: returns a value using the statement
    `return expression`
    - Use the function header to call "execute" the function:
      `function_name(PARAMETERS)`

Review

- Function `main()`
  - called when the program starts
  - Calls other listed functions
  - Defines the mainline logic of the program

```
def main():
    pass

if __name__ == '__main__':
    main()
```

Miniquiz

Standard library and others with `import`

- Standard library: library of pre-written functions that comes with Python
  - Example: `print(), input(), range()`

- External libraries: to access a module you need to write an import statement
  - Written at the top of the program
  - Format: `import module_name`
  - To access a function in the module, use dot notation

    ```
    module_name.function_name()
    ```

Standard library and others with `import`

- External modules:
  - `import random`: library functions for working with random numbers.
  - `import math`: library functions that are useful for performing mathematical calculations
  - `import sys`: library functions that allow access to system-specific parameters and functions.
  - `import datetime`: library functions that supply classes for manipulating dates and times.
  - `import turtle`: library functions that feature a drawing board, to command a 'turtle' to draw on it.

Scope of variables

- Scope: the part of a program in which a variable may be accessed
  - Local variable:
    - Accessed by the function in which created.
    - Only statements inside that function can access it, error will occur if another function tries to access the variable.
    - Different functions may have local variables with the same name.
  - Parameter variable:
    - Accessed by the function in which the parameter (argument) is used
    - Assigned the value of an argument when a function is called
    - General format: `def function_name(parameter):`

Scope of variables

- Global variable:
  - Created by assignment statement written outside all the functions
  - Can be accessed by any statement in the program file, including from within a function
  - General format: `global` *`variable_name`*
  - ***avoid using global variables :***
    - They making debugging difficult as functions that use global variables are usually dependent on those variables, which makes them hard to transfer to another program

Parameter variables

- **Argument:** piece of data that is sent into a function
  - When calling the function, the argument is placed in parentheses following the function name
  - Arguments are passed by position to corresponding parameters
    - First parameter receives value of first argument, second parameter receives value of second argument, etc.
- **Parameter**: variable that is assigned the value of an argument when the function is called
  - The parameter and the argument reference the same value
  - Changes made to a parameter value within the function do not affect the argument (pass by value)

Documenting functions

- Use the IPO Chart method to help document your code
  - Typically laid out in columns
  - Provides brief descriptions of input, processing, and output, without going into details.

IPO Chart for the `get_regular_price` Function		
Input	Processing	Output
None	Prompts the user to enter an item's regular price	The item's regular price

After Class

1. Review lecture notes and code from today
2. Begin working on Homework 1
3. Read textbook Chapter 5 and Chapter 6
4. Practice Python

Time to program

1. Launch Canvas
   - Select this course KEARSE- CS1301-A/C
   - Click Files - Lecture notes - **04notes.py, 04code.py**
   - **download the files to your CS1301 folder**
2. Launch IDLE
   - Click on the File - Open
   - Navigate to your CS1301 folder
   - Click on the downloaded file for today
   - Click the Open button