Built-in Python Functions (class slides)

CSC 110 Python Functions

Write a function

Write a Python function that does the following:

1. Its name is `greeting`
2. It takes two arguments, `first_name` and `last_name`
3. It returns a string with a greeting using `first_name` and `last_name`

```
print( greeting("Adriana", "Picoral") ) # Hello, Adriana Picoral!
```

Reading the documentation

Access the Python 3.11 documentation and read the definitions for `print()`, `round()`, `input()`, `len()`, `int()`, and `str()`.

With your table members, write a short definition for each of the built-in functions above on a white board.

`len()` function

- The `len()` function can be used with many types – we will be using it with `string` for now
- It returns the number of characters in a string

```
character_count = len("Adriana")
print(character_count)
```

7

Write a function

Write a Python function that does the following:

1. Its name is `count_characters`
2. It takes a string argument, `name`
3. It returns the total number of characters in `name`

```
print( count_characters("Adriana") ) # 7
print( count_characters("") ) # 0
print( count_characters(" ") ) # 1
print( count_characters("10") ) # 2
```

`input()` function

- The `input()` function prompts the user to input text in the standard output
- Whatever is inside the parentheses in `input()` will be written to the standard output (without a trailing newline, which you can add using `\n`).
- The function then reads a line from input, converts it to a string (stripping a trailing newline), and returns that
- `input()` always returns a string

```
input("What's your name?\n")
```

Write main()

Now you should have two functions in your `.py` script: `greeting` and `count_characters`.

Write a `main()` function.

Write a function

In the same script, write `main()`:

1. Prompt user to enter their first name and last name using `input()`
2. Call your `greeting` function and save the string it returns to a variable
3. Print the variable
4. Call your `count_characters` function with the user's full name and save the integer value returned to a variable
5. Print a message to the user: `Your full name has X letters.` – replacing X with the right character count

Announcements

- Midterm 1 on February 14 (this room)
- Modules 1-5 (practice problems on the website)
- Review session TBA
- BRING PHOTO ID TO THE EXAM

`input()` function

Use `input()` to get user input (it always returns a string)

`int()` function

- The `int()` function can be used to convert a string to an integer type
- It only works if the string only contains digits

```
age = '35'
age_int = int(age)
print(type(age), type(age_int))
```

```
<class 'str'> <class 'int'>
```

`float()` function

- The `float()` function can be used to convert a string to a float type
- It only works if the string only contains digits and optionally a decimal point

```
age = '35'
age_float = float(age)
print(type(age), type(age_float))
```

```
<class 'str'> <class 'float'>
```

Write a function

Write a Python function that does the following:

1. Its name is `calculate_year_born`, with no parameters
2. It prompts user to enter their age `input()`
3. It converts user's age to integer (since `input()` always returns a string)
4. It calculates (imperfectly) the year a person of `age` was born by subtracting `age` from `2023`
5. It returns an integer representing the approximate year person of `age` was born

```
print( calculate_year_born() ) # user enters 42, function returns 1981
```

Organizing your code

- Split your functions and your `main()` in different files
- Use `from script_name import *`
- For gradescope, submit only your script with all of your functions