Module 1: Hello, world!

JHU EP 605.206 - Introduction to Programming Using Python

Introduction

In this assignment you will setup your Python 3 development environment and write some basic code to help get you started. We'll look at the proverbial "Hello, world" example and use variables to build a formatted output string.

Skills: running a Python program in an IDE, working with variables, using the 'print' and 'input' functions, and string concatenation

Programming Assignment

For this assignment you will write the standard “Hello, world!” program:

1. Using the IDE of your choice, create a Python program in a file named ***hello.py*** that does the following:
    a. Prompts the user for their name (please use “Joe” as your input) and stores it in a variable
    b. Prints the string “Hello, <name>!” to the screen.
        i. Sample output: *Hello, Joe!*
    c. Take a screenshot of the result.

2. For the second part of the assignment you will create variables of different types and print the output using a formatted string:
    1. Create a program called ***due_date.py***
    2. Prompt the user for the day, month, year, time (minutes), and time (hours) this assignment is due and store each of those 5 values in a variable. Please use the actual assignment due date from Blackboard as your inputs.
    3. Use the ‘print’ statement to concatenate the variables into a string (converting integers to strings as needed) and output the due date of the assignment in the following format:
        a. “Module 1 Assignment is due on MM/DD/YYYY at HH:MM EST.”
            i. Fill in MM, DD, YYYY, HH, and MM using the corresponding variables
            ii. Your output should not include the quotation marks
    4. Take a screenshot of the result.

Deliverables

readme.txt

So-called “read me” files are a common way for developers to leave high-level notes about their applications. Here’s an example of a README file for the Apache Spark project. They usually contain details about required software versions, installation instructions, contact information, etc. For our purposes, your readme.txt file will be a way for you to describe the approach you took to complete the assignment so that, in the event you may not quite get your solution working correctly, we can still award credit based on what you were trying to do. Think of it as the verbalization of what your code does (or is supposed to do). Your readme.txt file should contain the following:

1. **Name**: Your name and JHED ID
2. **Module Info**: The Module name/number along with the title of the assignment and its due date
3. **Approach**: a detailed description of the approach you implemented to solving the assignment. Be as specific as possible. If you are sorting a list of 2D points in a plane, describe the class you used to represent a point, the data structures you used to store them, and the algorithm you used to sort them, for example. The more descriptive you are, the more credit we can award in the event your solution doesn’t fully work.
4. **Known Bugs**: describe the areas, if any, where your code has any known bugs. If you’re asked to write a function to do a computation but you know your function returns an incorrect result, this should be noted here. Please also state how you would go about fixing the bug. If your code produces results correctly you do not have to include this section.

Please submit your ***hello.py*** and ***due_date.py*** source code files along with a PDF file containing screenshots of your outputs in a PDF called JHEDID_mod1.pdf (ex: jkovba1_mod1.pdf). Please do not ZIP your files together.

Recap:

1. readme.txt
2. hello.py
3. due_date.py
4. Screenshots of your outputs

Please let us know if you have any questions via Teams or email!