FCS

# ***Assignment1:***

Part A:

* By now you should all have a GitHub account. Make sure to create a repository called

“foundations-cs-python” where you will be uploading all your FCS work.

* Design your GitHub profile, as we have seen and discussed in class.

Make sure to have:

* A professional picture
* Your fullname as your username

- A simple yet professional bio

- The programming languages and tools you know

- How to connect with you (your email, linkedin, instagram etc...)

Answer:

<https://github.com/zahraahusseini>

PART B

 Question 1:

What is the result of each of the following expressions (first solve by hand):

Answer:

a. 10\*(90+2)-5 = 915

b. 10\*90+2-5 = 897

c. 10\*90+(2-5) = 897

d. 10.0\*(90+2)-5 = 915.0

e. 120/(20+40)-(6-2)/4 = 1

f. 5.0/2 = 2.5

g. 5/2 = 2.5

h. 5.0/2.0 = 2.5

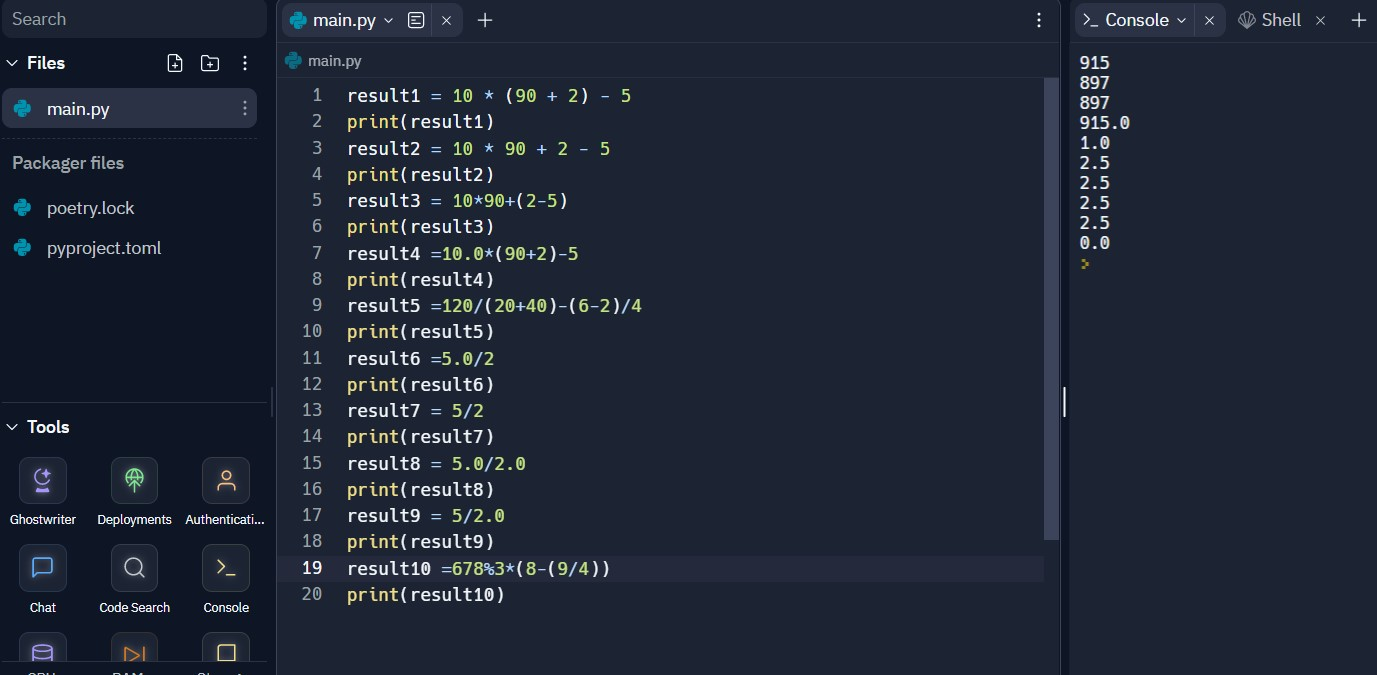
I . 5/2.0 = 2.5

j. 678%3\*(8-(9/4)) = 0

After you finished solving by hand, write a program that outputs each of the expressions

in Question 1 along with its result. Verify if what you wrote above is correct.

Answer:



 Question 2:

Read input from user (ID, name, date of birth, address), process the strings and output the

values as follows:

1. ID should always has the following format 0Nb (we need to add a 0 in the beginning).

Ex: User writes 1, output 01

User writes 3, output 03

2. Name should be in all uppercase.

3. Date of Birth should be in the following format/MM/YYYY Ex: User might input

DD-MM-YYYY not DD/MM/YYYY

4. Address should be in all lowercase and does not contain any leading/trailing spaces

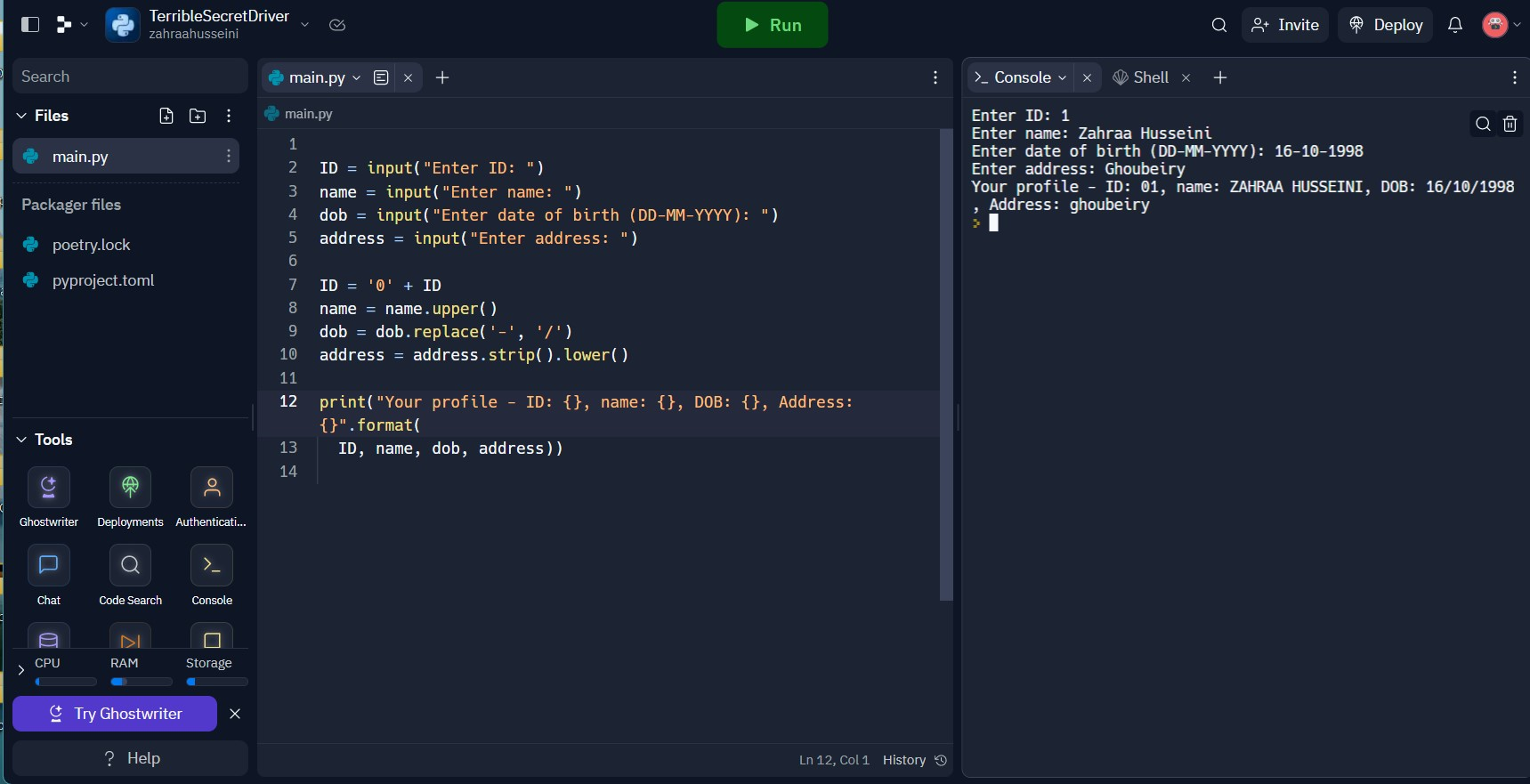
5. All fields should not have any extra space

Output Format:

Your profile - ID: [ID], name: [name], DOB: [Date of Birth], Address: [Address]

Hint: use the input() function in Python to receive data from user.

Answer:



 Question 3:

Create a program that prompts (asks) the user for a number and returns the number of

digits the number contains.

EX: if the number is 3725, the program displays “3725 has 4 digits”.

Answer:

number = input("Enter a number: ")

number = number.strip()

num\_of\_digits = len(number)

print(number, "has", num\_of\_digits, "digits.")

 Question 4 (Solve it without seeing the solution that we did in class for practice):

Compute the letter grade from the numeric one:

Generate a program that asks the user for a

numeric grade and outputs the equivalent letter

grade. (BONUS: Further split the grades into

A+, A, A-, B+, B etc...)

EX: if the user inputs 75, the program will output

“75 is equivalent to a C.”

Numeric Grade Letter Grade

<60 F

60<=grade<70 D

70<=grade<80 C

80<=grade<90 B

>=90 A

Answer:

numeric\_grade = float(input("Enter the numeric grade: "))

if numeric\_grade >= 90:

if numeric\_grade >= 98:

letter\_grade = "A+"

elif numeric\_grade >= 93:

letter\_grade = "A"

else:

letter\_grade = "A-"

elif numeric\_grade >= 80:

if numeric\_grade >= 88:

letter\_grade = "B+"

elif numeric\_grade >= 83:

letter\_grade = "B"

else:

letter\_grade = "B-"

elif numeric\_grade >= 70:

if numeric\_grade >= 78:

letter\_grade = "C+"

elif numeric\_grade >= 73:

letter\_grade = "C"

else:

letter\_grade = "C-"

elif numeric\_grade >= 60:

if numeric\_grade >= 68:

letter\_grade = "D+"

elif numeric\_grade >= 63:

letter\_grade = "D"

else:

letter\_grade = "D-"

else:

letter\_grade = "F"

print("{} is equivalent to a {}.".format(numeric\_grade, letter\_grade))

 Question 5:

Write a program that asks the user for a number n and outputs the below pattern. For

example, if the user inputs 6 the following pattern is displayed:

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*

\*\*\*

\*\*

\*

Note: Whatever number n that the user inputs, the pattern will have (n\*2)-1 lines.

Remember the different ways we can use the range() function.

---------------------------------------------------------------------------------------------------------------------

 Question 6:

Write a program that will ask the user for two numbers, then it prints all the even

numbers between the first number and second number.

For example, if the user enters 5 and 14, the program will print 6, 8, 10, 12, 14.

Note: Make sure that the user cannot enter a second number that is smaller than the first

number, not just assume. (Hint: you will need to use a loop right after the input to ensure

that the input is correct).

Answer:

first\_number = int(input("Enter the first number: "))

second\_number = int(input("Enter the second number: "))

while second\_number < first\_number:

print("Second number should be greater than or equal to the first number.")

second\_number = int(input("Enter the second number: "))

for num in range(first\_number, second\_number + 1):

if num % 2 == 0:

print(num)

#Goodluck! 