

Name: Tom Owen

Student Number: 2520277

WORKSHEET 1 – TESTING (Using Variables)

1. Original code:

```
#24/02/2026 Worksheet 1
```

```
print("Please enter Surname")  
surname = input()
```

```
print("Please enter National Insurance")  
national_insurance = input()  
if len(national_insurance)<9:  
    print("Error must be 9 characters long")  
    sys.exit()  
    """national_insurance = input()"""
```

```
print("Please enter Age")  
age = int(input())  
while age < 18:  
    print("Not valid")  
    age = int(input())  
print("Thank you")
```

```
print("Enter Number of working hours per week")  
hours_per_week = int(input())
```

2. Code Testing

For each test below, run your code and enter the values provided. Present a screenshot of the programme running. Did the programme behave correctly or did it do something unexpected, and if so, why?

Surname:

- i. Enter *JONES*
- ii. Leave blank
- iii. Enter *A*
- iv. Enter *123*

National Insurance Number:

- i. Enter PE534121B
- ii. Leave blank
- iii. Enter AB11223344556677 (ie too long for National Insurance number format)
- iv. Enter 22PE4088A (ie wrong format)

Age

- i. Enter 34
- ii. Enter 150
- iii. Enter -4
- iv. Enter a letter

Number of working hours per week:

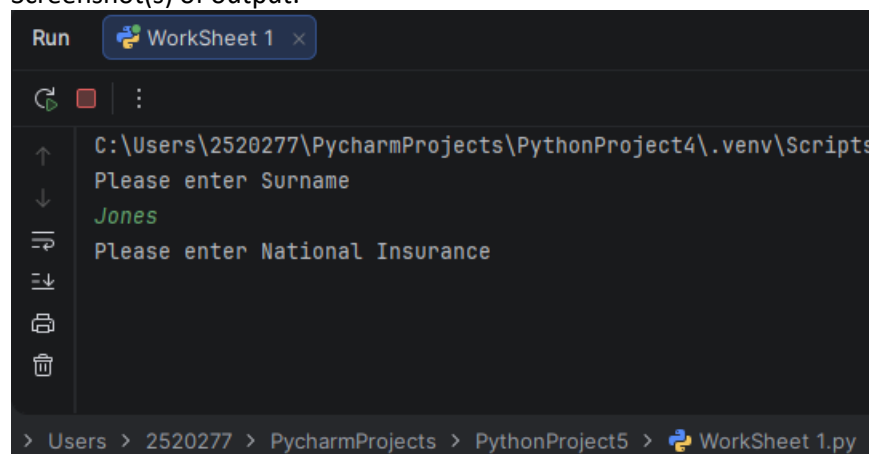
- i. Enter 37.5
- ii. Enter 150
- iii. Enter 0
- iv. Enter -2

Fill in a box for each test above. You should present 16 separate boxes, 1 for each test. (Note: the first two boxes are provided as a template)

Test:

Surname i. Enter JONES

Screenshot(s) of output:



```
Run  WorkSheet 1 x
Please enter Surname
Jones
Please enter National Insurance
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Scripts
> Users > 2520277 > PycharmProjects > PythonProject5 > WorkSheet 1.py
```

Result: Did the programme behave as expected ie syntactically and logically correct?

YES

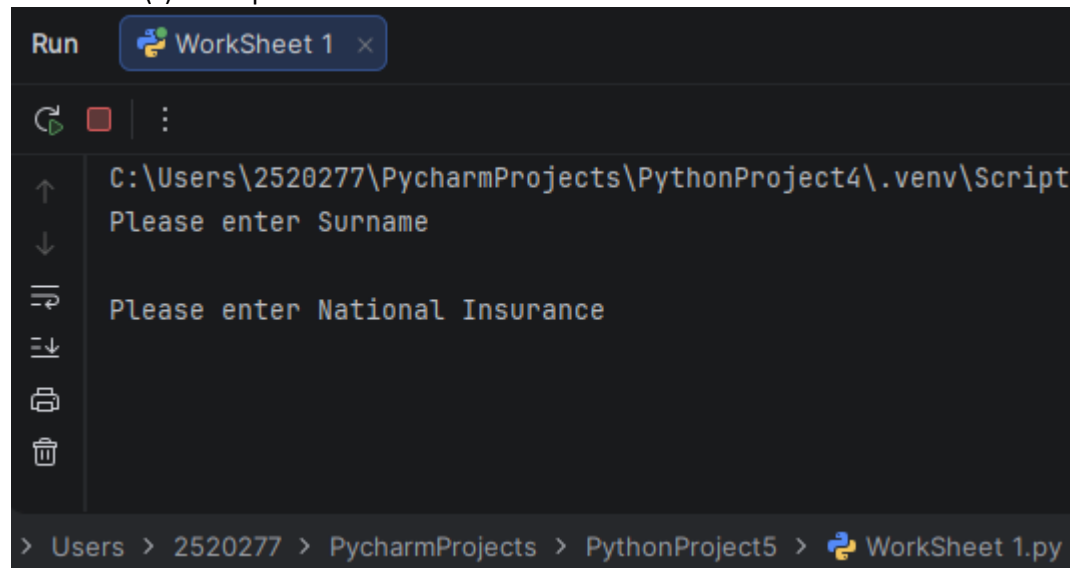
If YES, comment on what happened

The programme accepted the name and moved on to the next question asking for a national insurance number.

Test:

Surname ii Leave blank

Screenshot(s) of output:



```
Run  WorkSheet 1 x
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Script
Please enter Surname
Please enter National Insurance
> Users > 2520277 > PycharmProjects > PythonProject5 > WorkSheet 1.py
```

Result: Did the programme behave as expected ie syntactically and logically correct?

NO

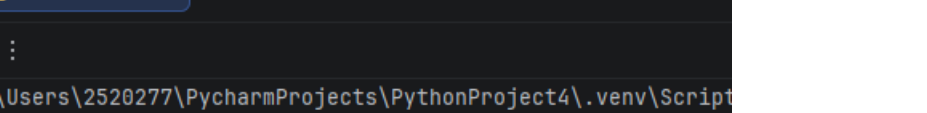
The programme did not crash, but it accepted the blank surname and moved on to the next question. This means it was syntactically correct, but in a logical sense it was not due to there being no validation added to check that the field was empty

Continue the rest of the tests below. (Copy and paste a box for each one)

Test:

Surname: Enter A

Screenshot(s) of output:



Run

WorkSheet 1

```
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Script
Please enter Surname
a
Please enter National Insurance
```

> Users > 2520277 > PycharmProjects > PythonProject5 > WorkSheet 1.py

Result: Did the programme behave as expected ie syntactically and logically correct?

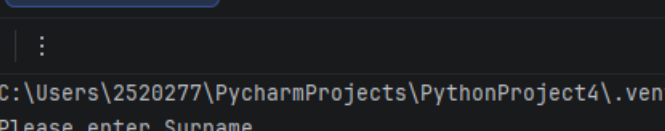
YES

The programme accepted A as the surname and continued to the next input. It was syntactically correct and logical because A is still a non-blank text entry. And although uncommon there are instances where some people have one letter surnames.

Test:

Surname: Enter 123

Screenshot(s) of output:



The screenshot shows the 'Run' window in PyCharm. The top bar indicates the file being executed is 'WorkSheet 1'. The console output is as follows:

```
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Script
Please enter Surname
123
Please enter National Insurance
```

The bottom status bar shows the file path: > Users > 2520277 > PycharmProjects > PythonProject5 > WorkSheet 1.py

Result: Did the programme behave as expected ie syntactically and logically correct?

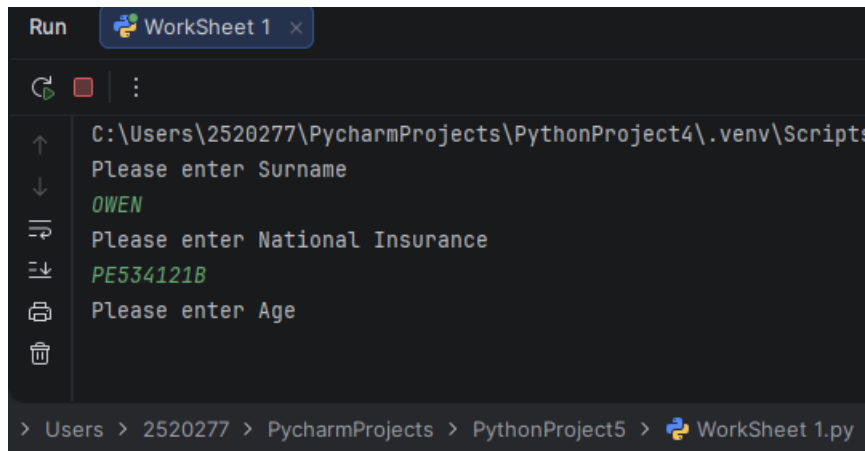
NO

The programme accepted 123 as a surname and continued to ask for the next input. It did not crash so it was syntactically correct, but it was not logically correct because a surname should not be numeric and additionally there was no validation had been implemented to prevent this.

Test:

NI number: Enter PE534121B

Screenshot(s) of output:



```
Run Worksheet 1 x
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Scripts\PythonProject5 > Worksheet 1.py
Please enter Surname
OWEN
Please enter National Insurance
PE534121B
Please enter Age
```

Result: Did the programme behave as expected ie syntactically and logically correct?

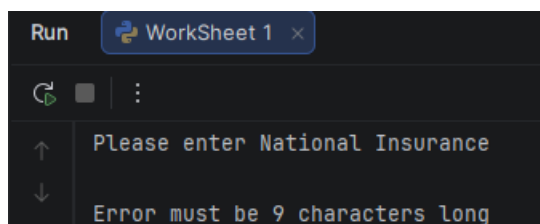
YES

The programme accepted PE534121B and continued to the next input. It behaved as expected because the entry was 9 characters long, so it passed the current validation check.

Test:

National Insurance Number ii. Leave blank

Screenshot(s) of output:



```
Run Worksheet 1 x
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Scripts\PythonProject5 > Worksheet 1.py
Please enter National Insurance
Error must be 9 chracters long
```

Result: Did the programme behave as expected ie syntactically and logically correct?

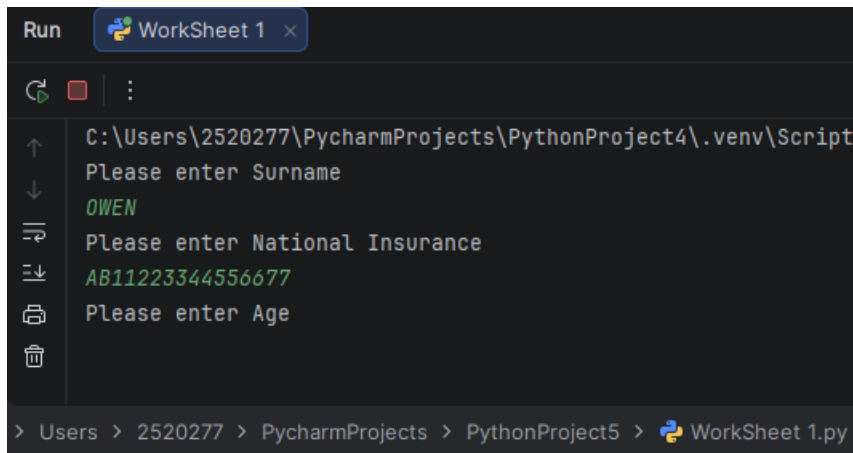
NO

The programme displays the error message telling the user it must be 9 chracters long and then stopped. This is syntactically correct, but not logically ideal because the user is not given a chance to re-enter their National Insurance Number.

Test:

National Insurance Number iii. Enter AB11223344556677

Screenshot(s) of output:



```
Run  WorkSheet 1 x
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Script
Please enter Surname
OWEN
Please enter National Insurance
AB11223344556677
Please enter Age
> Users > 2520277 > PycharmProjects > PythonProject5 > WorkSheet 1.py
```

Result: Did the programme behave as expected ie syntactically and logically correct?

NO

The program accepted the data and moved on to ask the user for the next input. It did not crash, but logically it was not correct because the national insurance number was too long and should have been rejected. The code only checks if the entry is less than 9 characters long, so entries longer than 9 characters are still accepted. Even though they are invalid.

Test:

National Insurance Number iv. Enter 22PE4088A

Screenshot(s) of output:

```
Run Worksheet 1 x
C:\Users\2520277\PycharmProjects\PythonProject4\.venv\Script
Please enter Surname
OWEN
Please enter National Insurance
22PE4088A
Please enter Age
> Users > 2520277 > PycharmProjects > PythonProject5 > Worksheet 1.py
```

Result: Did the programme behave as expected ie syntactically and logically correct?

NO

The program accepted 22PE4088A and continued to the next input. It did not crash, but it was not logically correct because the National Insurance Number was in the wrong format and should have been rejected. The code only checks the length and does not check the required format.

Test:

Age i. Enter 34

Screenshot(s) of output:

```
Run Worksheet 1 x
OWEN
Please enter National Insurance
PG986744B
Please enter Age
34
Thank you
Enter Number of working hours per week
> Users > 2520277 > PycharmProjects > PythonProject5 > Worksheet 1.py
```

Result: Did the programme behave as expected ie syntactically and logically correct?

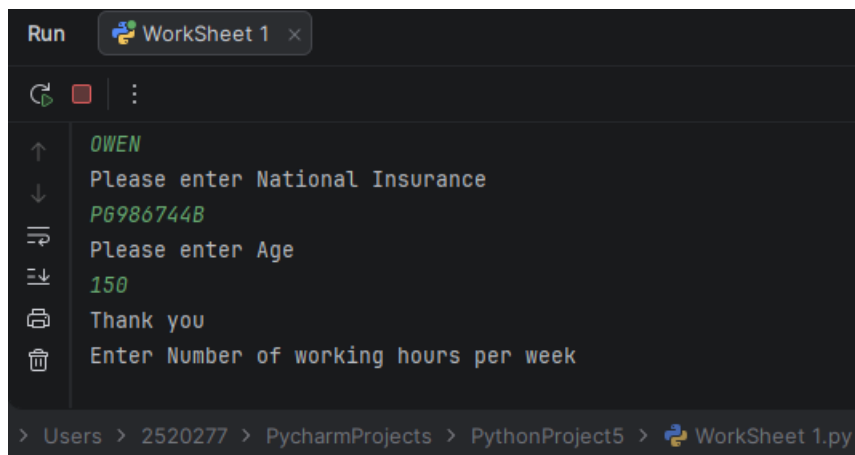
YES

The program accepted 34 as the age and continued to the next input. This was correct because 34 is above 18, so it passed the validation check.

Test:

Age ii. Enter 150

Screenshot(s) of output:

A screenshot of a PyCharm Run window titled 'WorkSheet 1'. The output console shows the following text: 'OWEN' (in green), 'Please enter National Insurance' (in green), 'P6986744B' (in green), 'Please enter Age' (in green), '150' (in green), 'Thank you' (in green), and 'Enter Number of working hours per week' (in green). The bottom of the window shows the file path: 'Users > 2520277 > PycharmProjects > PythonProject5 > WorkSheet 1.py'.

Result: Did the programme behave as expected ie syntactically and logically correct?

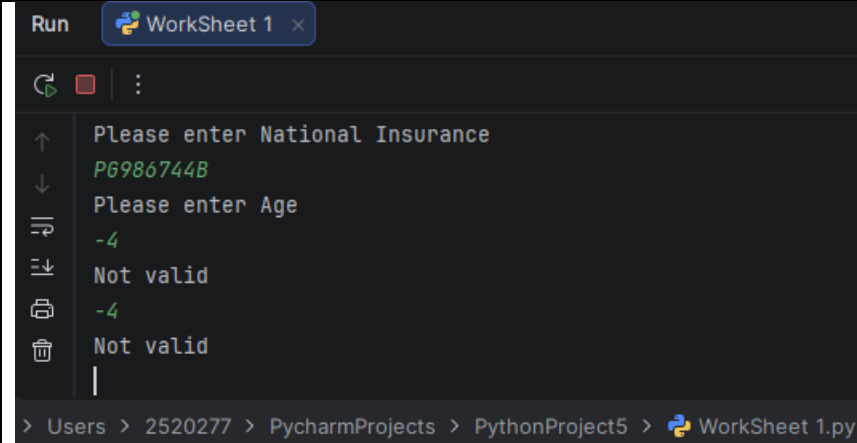
NO

The program accepted 150 as the age and continued to the next input. It did not crash, but it was not logically correct because 150 is not a realistic age and should have been rejected. The code only checks whether the age is below 18, so very high invalid ages are incorrectly accepted.

Test:

Age iii. Enter -4

Screenshot(s) of output:



```
Run Worksheet 1 x
Please enter National Insurance
P6986744B
Please enter Age
-4
Not valid
-4
Not valid
|
> Users > 2520277 > PycharmProjects > PythonProject5 > Worksheet 1.py
```

Result: Did the programme behave as expected ie syntactically and logically correct?

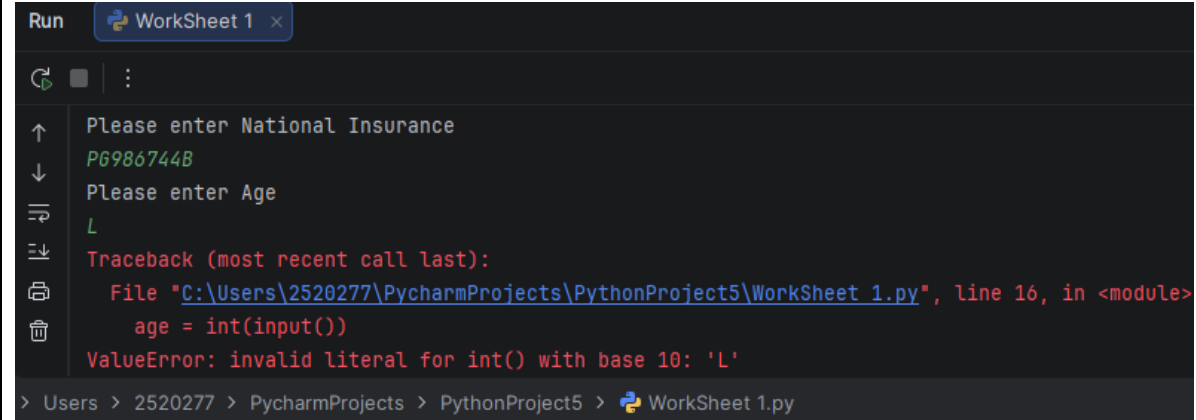
YES

The program correctly identified -4 as an invalid age, displayed “Not valid”, and allowed the user to enter the age again. This was logically correct because the code is designed to keep asking until a valid age of 18 or over is entered.

Test:

Age iv. Enter a letter

Screenshot(s) of output:



```
Run Worksheet 1 x
Please enter National Insurance
P6986744B
Please enter Age
L
Traceback (most recent call last):
  File "C:\Users\2520277\PycharmProjects\PythonProject5\Worksheet 1.py", line 16, in <module>
    age = int(input())
ValueError: invalid literal for int() with base 10: 'L'
> Users > 2520277 > PycharmProjects > PythonProject5 > Worksheet 1.py
```

Result: Did the programme behave as expected ie syntactically and logically correct?

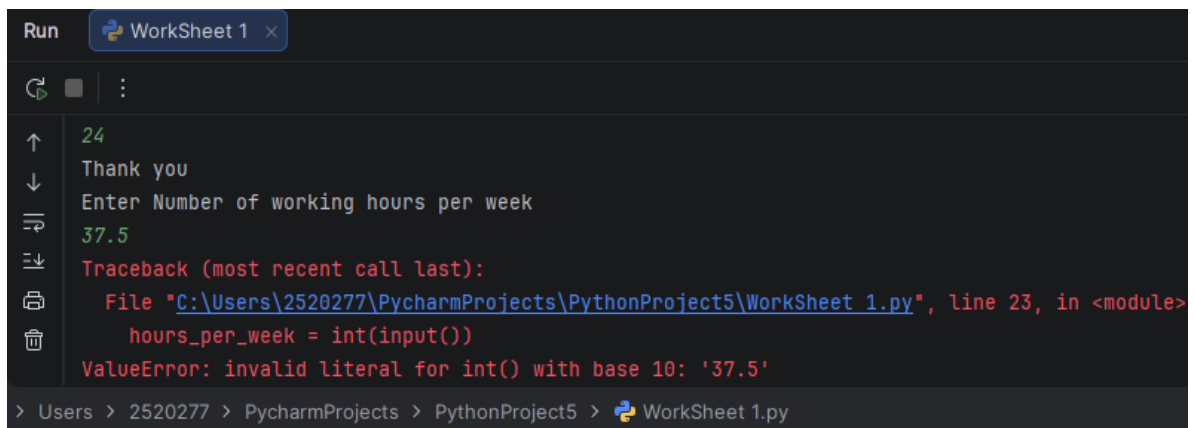
NO

The program crashed when a letter was entered for age because the code uses `int(input())`, which can only convert whole numbers. Entering a letter causes a conversion error, so the program does not continue normally.

Test:

Number of working hours per week i. Enter 37.5

Screenshot(s) of output:



```
Run  Worksheet 1 x
24
Thank you
Enter Number of working hours per week
37.5
Traceback (most recent call last):
  File "C:\Users\2520277\PycharmProjects\PythonProject5\WorkSheet 1.py", line 23, in <module>
    hours_per_week = int(input())
ValueError: invalid literal for int() with base 10: '37.5'
> Users > 2520277 > PycharmProjects > PythonProject5 > Worksheet 1.py
```

Result: Did the programme behave as expected ie syntactically and logically correct?

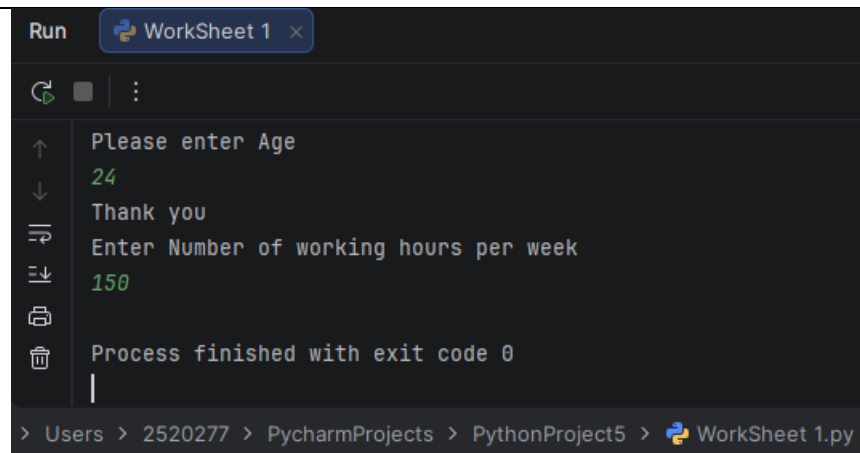
NO

The program crashed when 37.5 was entered because the code uses `int(input())` for working hours, and 37.5 is a decimal number. Since working hours can include half hours, this field should use `float` instead of `int`.

Test:

Number of working hours per week ii. Enter 150

Screenshot(s) of output:



```
Run Worksheet 1 x
Please enter Age
24
Thank you
Enter Number of working hours per week
150
Process finished with exit code 0
> Users > 2520277 > PycharmProjects > PythonProject5 > Worksheet 1.py
```

Result: Did the programme behave as expected ie syntactically and logically correct?

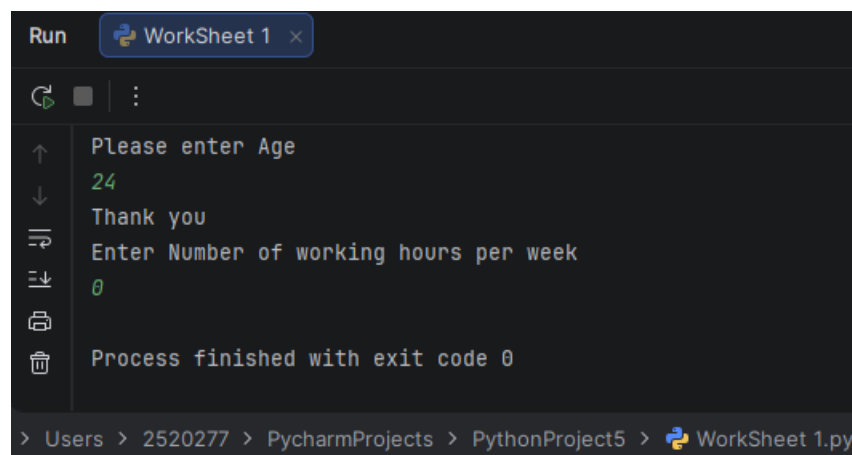
NO

The program accepted 150 as the number of working hours per week and continued. It did not crash, but this was not logically correct because 150 hours is not a realistic number of working hours in one week and should have been rejected. The code currently has no validation for the range of hours entered.

Test:

Number of working hours per week iii. Enter 0

Screenshot(s) of output:



```
Run Worksheet 1 x
Please enter Age
24
Thank you
Enter Number of working hours per week
0
Process finished with exit code 0
> Users > 2520277 > PycharmProjects > PythonProject5 > Worksheet 1.py
```

Result: Did the programme behave as expected ie syntactically and logically correct?

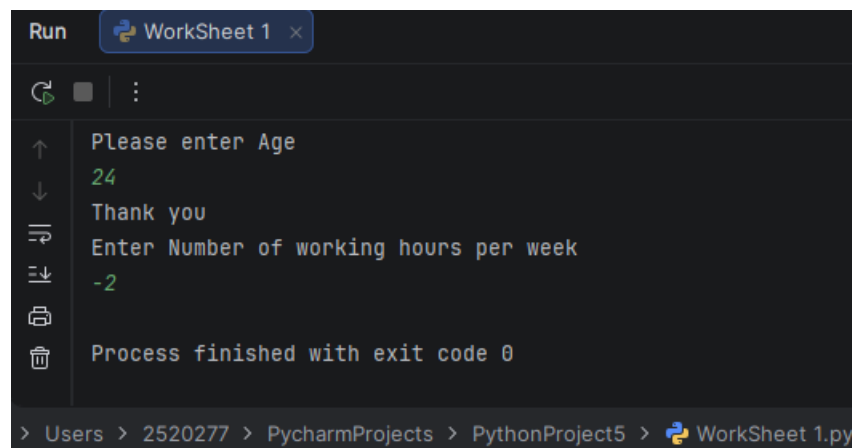
YES

The program accepted 0 as the number of working hours per week and continued. It did not crash. This may be acceptable because it is possible for someone to work 0 hours in a week, depending on the situation.

Test:

Number of working hours per week iv. Enter -2

Screenshot(s) of output:



```
Run  Worksheet 1 x
Please enter Age
24
Thank you
Enter Number of working hours per week
-2
Process finished with exit code 0
> Users > 2520277 > PycharmProjects > PythonProject5 > Worksheet 1.py
```

Result: Did the programme behave as expected ie syntactically and logically correct?

NO

The program accepted -2 as the number of working hours per week and continued. It did not crash, but it was not logically correct because working hours cannot be negative. The code currently has no validation to reject negative values.

3. Consider the results of the above tests and revisit your code in order to solve the problems. (Note: you may not be able to solve all of them, but solve as many as you can.)

Paste your revised code here:

[Copy and paste your revised code here – note, do not use a screenshot]

```
print("Please enter Surname")
surname = input()

while surname == "":
    print("Not valid")
    surname = input("Please enter Surname ")

print("Please enter National Insurance")
national_insurance = input()

while len(national_insurance) != 9:
    print("Error must be 9 characters long")
    national_insurance = input("Please enter National Insurance: ")

print("Please enter Age")
age = int(input())

while age < 18 or age > 100:
    print("Not valid")
    age = int(input("Please enter Age: "))

print("Enter Number of working hours per week")
hours_per_week = float(input())

while hours_per_week < 0 or hours_per_week > 100:
    print("Not valid")
    hours_per_week = float(input("Enter Number of working hours per week: "))

print("Thank you")
print("Surname:", surname)
print("National Insurance:", national_insurance)
print("Age:", age)
print("Hours per week:", hours_per_week)
```