

Lecture 2: Input and Conditionals

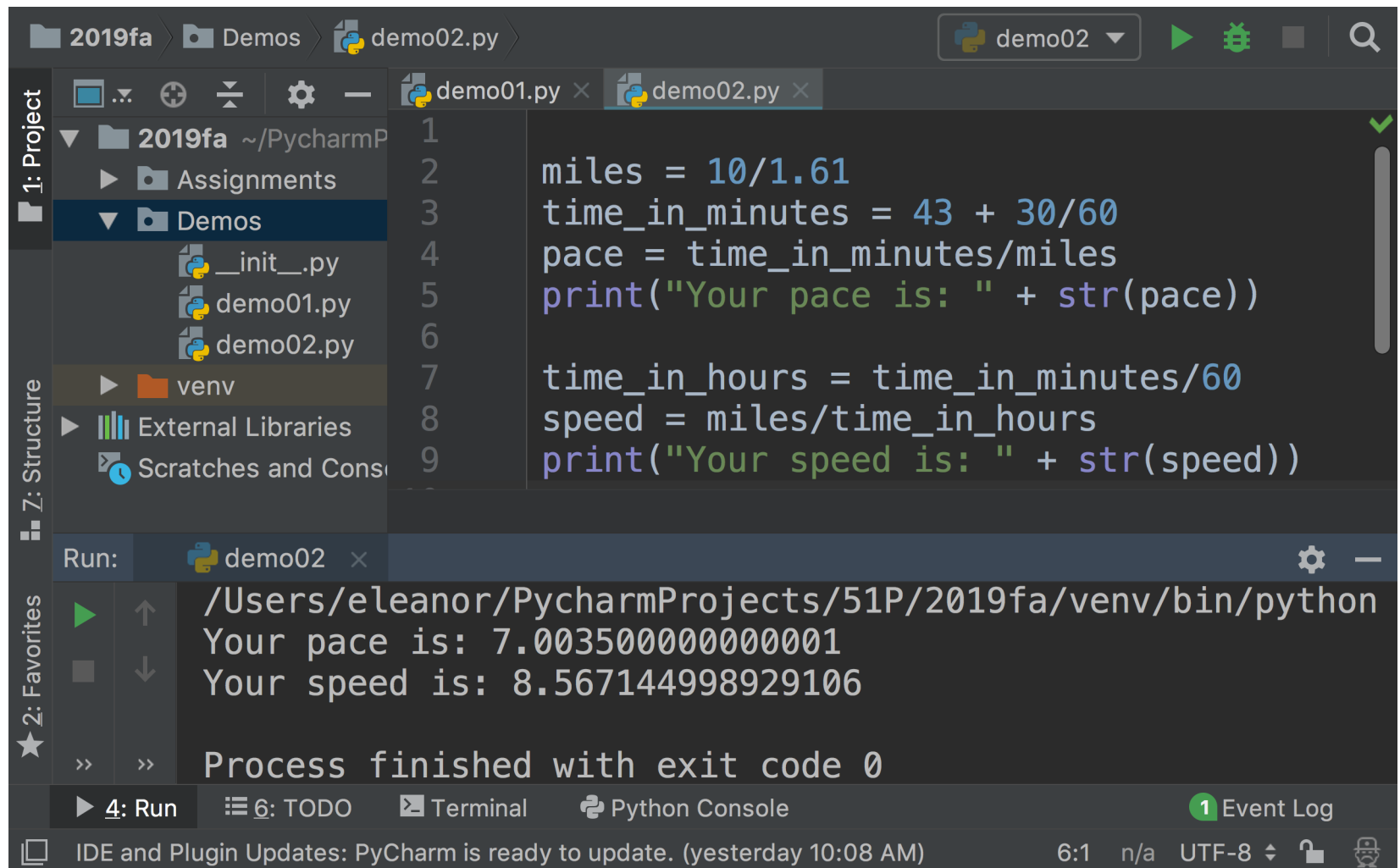
Eleanor Birrell

September 9, 2019

Announcements

- A0 has been graded
 - Remember to check your filenames
 - Remember to check that you've included your name
 - You must submit all files together in the same submission
 - You may submit as many times as you like; only the last submission will be graded
- Lab tonight/tomorrow in Edmunds 229. Attend the lab you are enrolled in
- A1 has been posted
- Slides are posted before class, handouts and demos are posted after
- Optional readings are listed
- Links to (free) textbook, Piazza, Submit, Gradescope

Review: Values, Types, Operations, Expressions, Variables



The screenshot shows the PyCharm IDE interface. The main editor displays a Python script named `demo02.py` with the following code:

```
1 miles = 10/1.61
2
3 time_in_minutes = 43 + 30/60
4
5 pace = time_in_minutes/miles
6 print("Your pace is: " + str(pace))
7
8 time_in_hours = time_in_minutes/60
9 speed = miles/time_in_hours
10 print("Your speed is: " + str(speed))
```

The Run tool window at the bottom shows the execution output:

```
Run: demo02 x
/Users/eleanor/PycharmProjects/51P/2019fa/venv/bin/python
Your pace is: 7.003500000000001
Your speed is: 8.567144998929106
Process finished with exit code 0
```

The IDE interface includes a Project view on the left showing the file structure, a Run toolbar at the top right, and a status bar at the bottom with various utility icons and system information.

Review: Useful (simple) Functions

- Example Functions
 - `print()`
 - `type()`
 - `str()`, `int()`, `float()`

```
print(1+2.001)
```

```
print(1*3)
```

```
print("1.0"+"2.0")
```

```
print("Happy"*2 + "?!"*3)
```

```
print(1*2 + "2"*2)
```

```
print(type("2"))
```

```
print(1*2 + int("2")*2)
```

```
print(str(1)*2 + "2"*2)
```

Input

- `input()` is a function that allows you to get a value from your user

Example

How would you change this program so that it asks the user for their time in minutes and seconds and then prints out their average time per mile for the 10K race?

```
miles = 10/1.61
time_in_minutes = 43 + 30/60
pace = time_in_minutes/miles
print("Your pace is: " + str(pace))

time_in_hours = time_in_minutes/60
speed = miles/time_in_hours
print("Your speed is: " + str(speed))
```

Exercise

Write a program that asks the user for word and a number, then prints the word followed by an exclamation point and a space that many times. For example:

Enter a word:

happy

Enter a positive integer:

3

happy! happy! happy!

What about?

```
what's your favorite positive integer?  
    13  
that's my favorite number too!
```

```
what's your favorite positive integer?  
    20  
mine is 13.
```


if-statements

- syntax

condition

```
if x == 13 :  
    print("that's my favorite number too!")
```

spaces matter!

- condition must be an expression that evaluates to True or False (type bool)

bool datatype - values

- Values: True and False
- Operators: and, or, not
- `<bool expression> and <bool expression>`
 - True if both expressions True, False otherwise
- `<bool expression> or <bool expression>`
 - True if at least one expression True, False otherwise
- `not <bool expression>`
 - returns True if expression is False, False otherwise

Boolean Expressions

- Examples of operations that return a bool:
 - relational operators: ==, !=, >, <, >=, <=

```
10 < 0
11 >= 11
11 > 11.0
10 == 10.1
```

```
x != 3
y <= x
x = 5
```

Example

Assume the variable `n` currently stores the value 25.

What will the following expressions evaluate to?

`2 < n and n < 30`

`(2 > n) or (n == 25)`

`not (n == n)`

`not (n != 3)`

Exercise

After executing each snippet of code, the variable `val` stores a value of type `bool`. What value does it store?

A) `val = (5 > 0 and 5 < 10)`

B) `n = 25`
`val = (n % 2 == 0 or n % 3 == 0)`

C) `s = "string"`
`val = ("String" != s)`

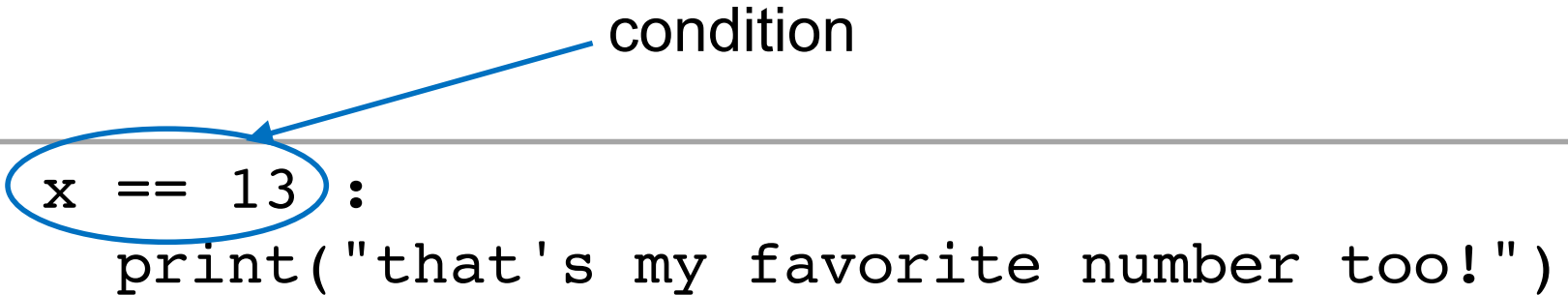
D) `val = not(True and False)`

if-statements

- syntax

condition

```
if x == 13 :  
    print("that's my favorite number too!")
```



- condition must be an expression that evaluates to True or False (type bool)
 - Booleans: True, False
 - relational operators: ==, !=, >, <, >=, <=
 - logical operators: and, or, not
 - functions that evaluate to type bool

if-else statements

- syntax

```
if x == 13:  
    print("that's my favorite number too!")  
else:  
    print("mine is 13")
```

Exercise

```
s1 = int(input())
if s1 % 4 == 0:
    print(4*s1)
if s1 % 2 == 0:
    print(2*s1)
else:
    print(s1)
```

What gets printed if the user inputs 2?

If they input 3?

If they input 4?

If they input 5.0?