PyLadies

Vienna 30.11.2019

Who?

International mentorship group with a focus on helping more women become active participants and leaders in the Python open-source community.

Our mission is to promote, educate and advance a diverse Python community through outreach, education, conferences, events and social gatherings.

PyLadies

- Regular meetups
- Courses
- Workshops on specific topics
- Mentorship
- Open-source community

Agenda for today

---

1. Python Fundamentals II
2. Lightning talk - Bioinformatics
3. Exercises, networking, discussion of own projects
4. Join us for a beer (or anything else) later

Conditions - recap

if something, do something… else?

```
if a = 3:
    print(“a is equal to 3!”)
else:
    print(“a is not 3”)
```

- if you want to check more conditions, use elif

Loops - For Loop

```
for greetings in 'Hi', 'Hello', 'Hola', 'Hey', 'Hallo':

    print(greetings  + '!')
```

- Repeat part of the code
- Variable is set subsequently to a, b, c, ...

For - understand how it works

```
sum_a = 0

for number in 8, 45, 9, 21:

    sum_a = sum_a + number

print(sum_a)
```

Loops - While

- With **for** loop, we know how many times we will repeat a command
- With **while** we are looking for condition to meet. Body of code repeats until condition says to stop.

```
response = input('Say aaa! ')

while text != ‘Hi’:
    print('wrong, try again')
    text = input('Say Hi! ')
```

While

- it is easy to create infinite loop - to break it, press **Ctrl + C** → creates an error and ends the execution
- Magic world **break** - ends the **while** (or any other loop) and continue with the code written after

```python
while True:
    answer= input('Say hi! ')
    if answer == 'Hi':
        print('Hello there!')
        break
    print('Wrong, try again')
print('Success')
```

While - exercise

- Start at zero and with every iteration add 1. If the sum is divisible by 10 then add 15. If the sum is greater than 157, end to code.

Exercises

1. Write a Python program to guess a number between 1 to 9.

   User is prompted to enter a guess. If the user guesses wrong then the prompt appears again until the guess is correct, on successful guess, user will get a "Well guessed!" message, and the program will exit. Hint: `from random import randint   number = randint(a, b)`

2. Write a Python program to print those numbers which are divisible by 7 and multiple of 5, between 1500 and 2700 (both included).

Functions

- We used functions written by someone else:
    - print(“Hello World”)
- Now we will create our own functions:

```
def rectangle_circumference(width, height):
    "Return circumference of rectangle with given sizes"
    return 2 * (width + height)
print(rectangle_circumference(4, 2))
```

Functions

- When executing a function, arguments are assigned to variables in body of the function.
- **Return** will end a function!
- Try to write a function which computes area of ellipse
  - formula is: **S = π * a * b** where **a, b** are length of axes
  - `from math import pi`

Functions

- What is the difference between print and return?
- Why to use return?

- Because we can use the result later in our code

Exercises

- Use **input** function to ask the user about ellipse axes length - eg. `x = float(input('Insert the length of 1. axis: '))`
- Use given values as arguments for your ellipse function
- What happens if there is no return?

Local vs. Global

- Functions can use variables from “outside”
- All variables introduced “inside” function are local and exist only inside this function

```
x = 0
def set_x(value):
    x = value # Assign to local variable
set_x(40)
print(x)
```

Turtle exercises - time for something fun!

- First start with the console Python:
    - `from turtle import forward`
    - `forward(50)`
- To reveal the turtle:
    - `from turtle import shape`
    - `shape('turtle')`
- https://docs.python.org/3.7/library/turtle.html?highlight=turtle

Turtle can also move!

```
from turtle import left, right

forward(50)

left(60)

forward(50)

right(60)

forward(50)
```

Turtle

- Back to the files - create a new file eg. **pyladies02.py**
- !important! do not name it as **turtle.py**
- Always use **exitonclick()** function in the end of your code - why?
- Create a square!

Turtle solution

```
from turtle import forward, left, exitonclick
forward(50)
left(90)
forward(50)
left(90)
forward(50)
left(90)
forward(50)
left(90)
```

Combine everything with turtle

1. Create a new square using **forward** only twice in total.
2. Using functions **penup** and **pendown** you can say turtle to stop and continue drawing. Draw a **dashed** line.
3. Draw 3 squares, each turned by 20°
4. Draw a rainbow - `from turtle import pencolor`, `pencolor('red')`

Lists

- new data structure, useful for storing elements (items)

```
numbers = [1, 1, 2, 3, 5, 8, 13]
print(numbers)

for number in numbers:
    print(number)
```

Lists

- Can have mixed data types inside
- Accessing elements with indexing
- We can use them for a lot useful operations:
  - `print(numbers[2])`
  - `print(numbers[2:-3])`
- Unlike strings or numbers, lists can be modified
  - `prime_numbers= [2, 3, 5, 7, 11, 13, 17]`
  - `print(prime_numbers)`
  - `prime_numbers.append(19)`
  - `print(prime_numbers)`

Lists - methods

- extend
  - `another_prime_numbers = [23, 29, 31]`
  - `prime_numbers.extend(another_prime_numbers)`
  - `print(prime_numbers)`
- change number
  - `numbers = [1, 0, 3, 4]`
  - `numbers[1] = 2`
  - `print(numbers)`
- delete number
  - `numbers= [1, 2, 3, 4, 5, 6]`
  - `del numbers[-1]`
  - `print(numbers)`

Lists - methods

- Sort
  - `list_a= [4, 7, 8, 3, 5, 2, 4, 8, 5]`
  - `list_a.sort(reverse=True)`
  - `print(list_a)`
- `.count()`
- `len(list)`
- `.index()`
- `.clear()`
- split
  - `words = 'This sentence is too complicated, split it into words!'.split()`
  - `print(words)`

Randomness in coding

- module called **random**
- random.shuffle()
- random.choice()

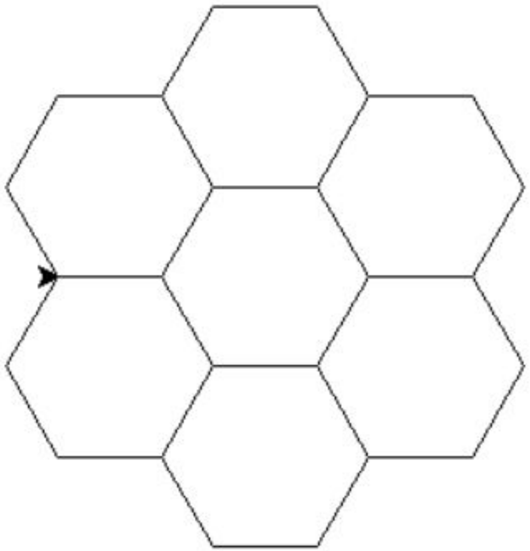
How we can use this methods? Think about some examples of usage

Lists exercises

1. Write a program to get the smallest number from a list.
2. Write a program to sum all the items in a list.
3. Write a **function** to print a list after removing the 0th, 4th and 5th elements.
4. Write a program to shuffle and print 1st element of new list

Homeworks :) - turtle

1. Draw stairs
2.



Homeworks :)

3. Write a Python program that prints all the numbers from 0 to 6 except 3 and 6. Note : Use '**continue**' statement.
Expected Output : 0 1 2 4 5

4. Exercise 1-5 from list with exercises

[Extra] Exercise 7 from list with exercises

Resources and materials

- advent of code - adventofcode.org
- hackerrank - hackerrank.com
- Django Girls - django tutorial (**in April in Vienna**)🎉
- https://www.practicepython.org
- Nice Python exercices at one place https://github.com/tystar86/python_exercises/tree/master/Tasks
- https://automatetheboringstuff.com
- https://diveintopython3.problemsolving.io

Next topics - 18. 1. 2020

Dictionaries

Handling the files

Numpy - arrays

Git - version control

Jupyter notebooks

Study session

- Study session on 18.12. from 18:00 for two hours
- No lecture, just practice and working on personal projects or homeworks :)

Thank you and see you next time

Suggest a topic you are interested in!

If you want to help with organization, let me know!

Please fill the survey I showed during meetup :)