INDEX PAGE
Pages
Lab 1
FAQ
Lab 2
Lab 3
Lab 7
Lab 6
Testing tips
Lab 4
Lab 10
Lab 5
Module overview
Support
Installing and finding Python
Snippets
Lecture slides
Lab 8
Lab 9
First steps
Testing demo

Home | **Lab 2**

Laboratory 2: simple functions and conditional branching (if)

Prerequisits: functions, import statements, math, int, float, if-then-else, PEP8

Contents

In Laboratory 1 you have familiarized yourself with the submission and testing system. In Laboratory 2 the task is to practice functions and if-else conditional statements. We will create file `lab2.py`, which can be submitted (and re-submitted) for automatic testing by sending an email to feeg1001@soton.ac.uk with subject `lab 2` and an attachment `lab2.py`.

PEP8

To monitor the coding style we turn on the PEP8 style guide monitoring in Spyder by:

1. Going to preferences in Spyder menu
2. Clicking on `Editor` in the selection list on the left
3. Clicking on `Code Introspection/Analysis` (top right corner)
4. Ticking the box `Real-time code style analysis`
5. Clicking `Apply` and `OK` (bottom right corner)

Exercices

Please define the following functions in the file `lab2.py`:

1. A function `box_volume(a, b, c)` that calculates and returns the volume of a cuboid with edge lengths `a`, `b`, `c`.

   Examples:

   ```
   >>> print(box_volume(1, 1, 1))
   1
   >>> print(box_volume(1, 2, 3.5))
   7.0
   >>> print(box_volume(1, 1, 0))
   0
   ```

   Examples (IPYthon):

   ```
   In [ ]: box_volume(1, 1, 1)
   Out[ ]: 1

   In [ ]: box_volume(1, 2, 3.5)
   Out[ ]: 7.0

   In [ ]: box_volume(1, 1, 0)
   Out[ ]: 0
   ```

2. A function `fall_time(h)` that returns the time (in seconds) needed for an object falling from a tower of height h (in meters) to hit the ground (ignoring air friction). Use

   $$h(t) = \frac{1}{2} g t^2 \quad \text{with} \quad g = 9.81\,\frac{\mathrm{m}}{\mathrm{s}^2}$$

   with h being the fall distance after time t.

   Examples:

```
>>> print(fall_time(10))
1.4278431229270645
>>> print(fall_time(1))
0.4515236409857309
```

Examples (IPython):

```
In [ ]: fall_time(10)
Out[ ]: 1.4278431229270645

In [ ]: fall_time(1)
Out[ ]: 0.4515236409857309
```

3. A function `interval_point(a,b,x)` that takes three numbers and interprets `a` and `b` as the start and end point of an interval, and `x` as a fraction between 0 and 1 that determines how far to go towards b, starting at a.

   Examples:

   ```
   >>> print(interval_point(100, 200, 0.5))
   150.0
   >>> print(interval_point(100, 200, 0.2))
   120.0
   ```

   Examples (IPython):

   ```
   In [ ]: interval_point(100, 200, 0.5)
   Out[ ]: 150.0

   In [ ]: interval_point(100, 200, 0.2)
   Out[ ]: 120.0
   ```

4. A function `impact_velocity(h)` that returns the velocity (in metre per second) with which an object falling from a height of `h` meters will hit the ground. Use v(t)=g*t, with v(t) the velocity at time t, and g=9.81 m/s^2.

5. A function `signum(x)` that:
   - returns `1` if x > 0
   - returns `0` if x = 0
   - returns `-1` if x < 0

   Example:

   ```
   >>> print(signum(2012))
   1
   ```

   Example (IPython):

   ```
   In [ ]: signum(2012)
   Out[ ]: 1
   ```

6. A function `seconds2days(n)` which accepts a number of seconds (either as int or float) and converts the number of seconds in the corresponding number of days. The function should return the number of days as a floating point variable.

   Example:

   ```
   >>> seconds2days(43200)
   0.5
   ```

   Example (IPython):

   ```
   In [ ]: seconds2days(43200)
   Out[ ]: 0.5
   ```

   For your entertainment: how many days are 10! seconds? (10!=3628800).

7. A function `box_surface(a,b,c)` that computes and returns the surface area of a box (i.e. cuboid) with these edge lengths `a`, `b`, and `c`. Imagine we need to paint the surface of the box and thus need to know its area to buy the right amount of paint.

Example:

```
>>> print(box_surface(1, 1, 1))
6
>>> print(box_surface(2, 2, 0))
8
```

Example (IPython):

```
In [ ]: box_surface(1, 1, 1)
Out[ ]: 6

In [ ]: box_surface(2, 2, 0)
Out[ ]: 8
```

8. A function `triangle_area(a,b,c)` that computes and returns the area A of a triangle with edge lengths a, b, and c. You can use the equation

$$A = \sqrt{s(s-a)(s-b)(s-c)} \quad \text{with} \quad s = \frac{a+b+c}{2}$$

Remember to carefully check each function for correct implementation (see Testing tips).

Now submit your `lab2.py` file (repeatedly if necessary) by emailing it to feeg1001@soton.ac.uk using `lab 2` as the subject line. Wait for an email reply and improve your functions (if necessary) until it passes all the tests. Seek help of demonstrators to interpret the error messages if necessary -- it is important that you learn how to read those.

Last updated: 2019-01-06 at 17:34

Return to Top

Page last modified Sun Nov 18 17:47:59 2018. |