Exercise Sheet 03 Solution – `True` or `False`?

Sebastian Höffner Aline Vilks

Deadline: Mon, 24 Apr 2017 08:00 +0200

Exercise 1: Fizz Buzz

File: `code/fizz_buzz.py`

```python
def fizz(number):
    return not number % 3


def buzz(number):
    return not number % 5


def fizz_buzz(to):
    for number in range(1, to + 1):
        answer = ''
        if fizz(number):
            answer += 'fizz'
        if buzz(number):
            answer += 'buzz'

        if answer != '':
            print(answer)
        else:
            print(number)


fizz_buzz(20)  # Play to 20
# expecting fizz: 3, 6, 9, 12, 18
# expecting buzz: 5, 10, 20
# expecting fizzbuzz: 15
```

Output:

```
1
2
fizz
4
buzz
fizz
7
8
fizz
buzz
11
fizz
13
14
fizzbuzz
16
17
fizz
19
buzz
```

Exercise 2: N bottles

File: code/n_bottles.py

```python
def bottles(n):
    return ('1 bottle' if n == 1 else str(n) + ' bottles') + ' of beer'


def n_bottles(n):
    if not 5 <= n <= 99:
        print('I want to sing funnier songs than "' + bottles(n) + '".\n')
        return
    while n > 0:
        print(bottles(n) + ' on the wall,\n  ' + bottles(n) + '.')
        n = n - 1
        print('Take one down and pass it around,\n  ' +
              bottles(n if n > 0 else 'no more') +
              ' on the wall.\n')


n_bottles(2)
n_bottles(1013)
n_bottles(5)
```

Output:

```
I want to sing funnier songs than "2 bottles of beer".

I want to sing funnier songs than "1013 bottles of beer".

5 bottles of beer on the wall,
  5 bottles of beer.
Take one down and pass it around,
  4 bottles of beer on the wall.

4 bottles of beer on the wall,
  4 bottles of beer.
Take one down and pass it around,
  3 bottles of beer on the wall.

3 bottles of beer on the wall,
  3 bottles of beer.
Take one down and pass it around,
  2 bottles of beer on the wall.

2 bottles of beer on the wall,
  2 bottles of beer.
Take one down and pass it around,
  1 bottle of beer on the wall.

1 bottle of beer on the wall,
  1 bottle of beer.
Take one down and pass it around,
  no more bottles of beer on the wall.
```

Exercise 3: Turtle Drawings

File: `code/turtle_drawing.py`

```python
import time
import turtle



LENGTH = 5
ANGLE = 40



def draw_tree(h):
    if h == 0:
        return
    turtle.forward(LENGTH * h)
    turtle.left(ANGLE)
    draw_tree(h - 1)
    turtle.right(2 * ANGLE)
    draw_tree(h - 1)
    turtle.left(ANGLE)
    turtle.backward(LENGTH * h)


def draw_house():
    height = 5
    width = 7
    roofside = (width ** 2 / 2) ** (1 / 2)
    turtle.forward(LENGTH * height)  # left wall
    turtle.right(45)  # roof
    turtle.forward(LENGTH * roofside)
    turtle.right(90)
    turtle.forward(LENGTH * roofside)
    turtle.right(45)
    turtle.forward(LENGTH * height)  # right wall
    turtle.right(90)
    turtle.forward(LENGTH * width)  # bottom line
    turtle.right(90)


def draw_world(curvature_step=0):
    if curvature_step > 0:
        villages = 360 // 4 // curvature_step
    else:
        villages = 5

    for i in range(villages):
        prepare_drawing()
        draw_house()
        finish_drawing()

        turtle.right(curvature_step)
        turtle.forward(LENGTH * 11)

        for j in range(3):
            prepare_drawing()
            draw_tree(3 + j % 2 * 2)
            finish_drawing()
            turtle.right(curvature_step)
            turtle.forward(LENGTH * 3)

        turtle.forward(LENGTH)


def init():
    turtle.reset()
    turtle.shape('turtle')
    turtle.speed('fastest')
    turtle.up()


def prepare_drawing():
    turtle.down()
    turtle.left(90)


def finish_drawing():
    turtle.right(90)
    turtle.up()


def draw_flat_world():
    init()
    turtle.goto(-300, 0)

    draw_world()


def draw_round_world():
    init()
    turtle.goto(0, 300)
    turtle.hideturtle()

    draw_world(5)


def draw():
    draw_flat_world()
    time.sleep(5)
    draw_round_world()
    turtle.done()
```

```
draw()
```