# Solution 2: Give me your feature name and I'll call you ETH Zurich

#### 1 Zurich needs more stations

Listing 1: More feature calls

```
explore
      – Modify the map.
 do
    Zurich.add_station ("Zoo", 1600, 500)
    Zurich.connect_station (6, "Zoo")
    Zurich\_map.update
    Zurich\_map.fit\_to\_window
    wait(3)
    Zurich_map.station_view (Zurich.station ("Zoo")).highlight
    Zurich\_map.station\_view (Zurich.station ("\mathbf{Zoo}")).unhighlight
    wait(1)
    Zurich\_map.station\_view\ (Zurich.station\ ("Zoo")).highlight
    Zurich\_map.station\_view\ (Zurich.station\ ("Zoo")).unhighlight
    wait (1)
    Zurich_map.station_view (Zurich.station ("Zoo")).highlight
    wait(1)
    Zurich\_map.station\_view\ (Zurich.station\ ("Zoo")).unhighlight
 end
```

# 2 Introducing yourself

Listing 2: Introduction

```
execute

-- Run application.

do

Io.put_string ("Name: ")

Io.put_string ("John Smith")

Io.new_line

Io.put_string ("Age: ")

Io.put_integer (20)

Io.new_line

Io.put_string ("Mother tongue: ")

Io.put_string ("English")

Io.new_line

Io.put_string ("Has a cat: ")
```

 $Io.put\_boolean (True)$  end

## 3 Command or Query?

- 1. name is a query.
- 2. buildings is a query.
- 3. add\_line is a command.
- 4. connecting\_lines is a query.
- 5.  $move\_all$  is a command.
- 6. north is a query.

### 4 MOOC: Objects and Classes

The order in which the questions and the answers appear here in the solution may vary because they are randomly shuffled at each attempt.

- One class is a template for defining a set of possible objects.
- Each object is an instance of its generating class.
- While classes exist only in the software text, objects exist during execution as well.
- In software text objects are visible and represented by names denoting run-time instances of classes.
- One class represents a category of things. One object represents one of these things.