CHAPTER 1

Expressions and Messages - Solution

This exercise is about reading and understanding Pharo expressions, and differentiating between different types of messages and receivers.

Note that in the expressions you will be asked to read and executed, you can assume that the implementation of methods generally corresponds to what their message names imply (i.e., `2 + 2 = 4`).

In addition, most of the expressions we use in the exercises are expressions that you can execute in Pharo, so do not hesitate.

Exercise: Literal objects

What kind of object does the following literal expressions refer to?

Exercise:

```
'Hello, Dave'
```

Solution.

```
a string
```

Exercise:

```
1.3
```

Solution.

```
a float
```

Exercise:

```
#node1
```

Solution.

```
  a symbol (unique string)
```

Exercise:

```
#(2 33 4)
```

Solution.

```
  an array
```

Exercise:

```
[ :each | each scale: 1.5 ]
```

Solution.

```
  a block (lexical closure)
```

Exercise:

```
$A
```

Solution.

```
  a character
```

Exercise:

```
true
```

Solution.

```
  a boolean
```

Exercise:

```
1
```

Solution.

```
  a smallinteger
```

1.1 Exercise: Messages

For each of the expressions below, fill in the answers:

- What is the receiver object?
- What is the message selector?
- What is/are the argument (s)?
- What is the result returned by this expression execution?

Exercise:

```
3 + 4
```

Solution.

```
  receiver: 3
  selector: +
  argument: 4
```

Exercise:

```
Date today
```

Solution.

```
  receiver: Date
  selector: today
  argument: none
```

Exercise:

```
anArray at: 1 put: 'hello'
```

Solution.

```
  receiver: anArray
  selector: at:put:
  argument: 1 and 'hello'
```

Exercise:

```
anArray at: i
```

Solution.

```
  receiver: anArray
  selector: at:
  argument: i
```

Exercise:

```
#(2 33 -4 67) collect: [ :each | each abs ]
```

Solution.

```
  receiver: #(2 33 -4 67)
  selector: collect:
  argument:  [ :each | each abs ]
```

Exercise:

```
25 @ 50
```

Solution.

```
  receiver: 25
  selector: @
  argument: 50
```

Exercise:

```
SmallInteger maxVal
```

Solution.

```
  receiver: SmallInteger
  selector: maxVal
  argument: none
```

Exercise:

```
#(a b c d e f) includesAll: #(f d b)
```

Solution.

```
  receiver: #(a b c d e f)
  selector: includesAll:
  argument: #(f d b)
```

Exercise:

```
true | false
```

Solution.

```
  receiver: true
  selector: |
  argument: false
```

Exercise:

```
Point selectors
```

Solution.

```
  receiver: Point
  selector: selectors
  argument:
```

Exercise: Scope

Exercise:

- What can one assume about a variable named `Transferator`?

`Transferator` is a global variable: either a class, a global variable or a class variable.

Solution.

Exercise:

- What can one assume about a variable named `rectangle`?

Solution. `rectangle` is a local variable: either a temporary, an instance variable, or a method argument.