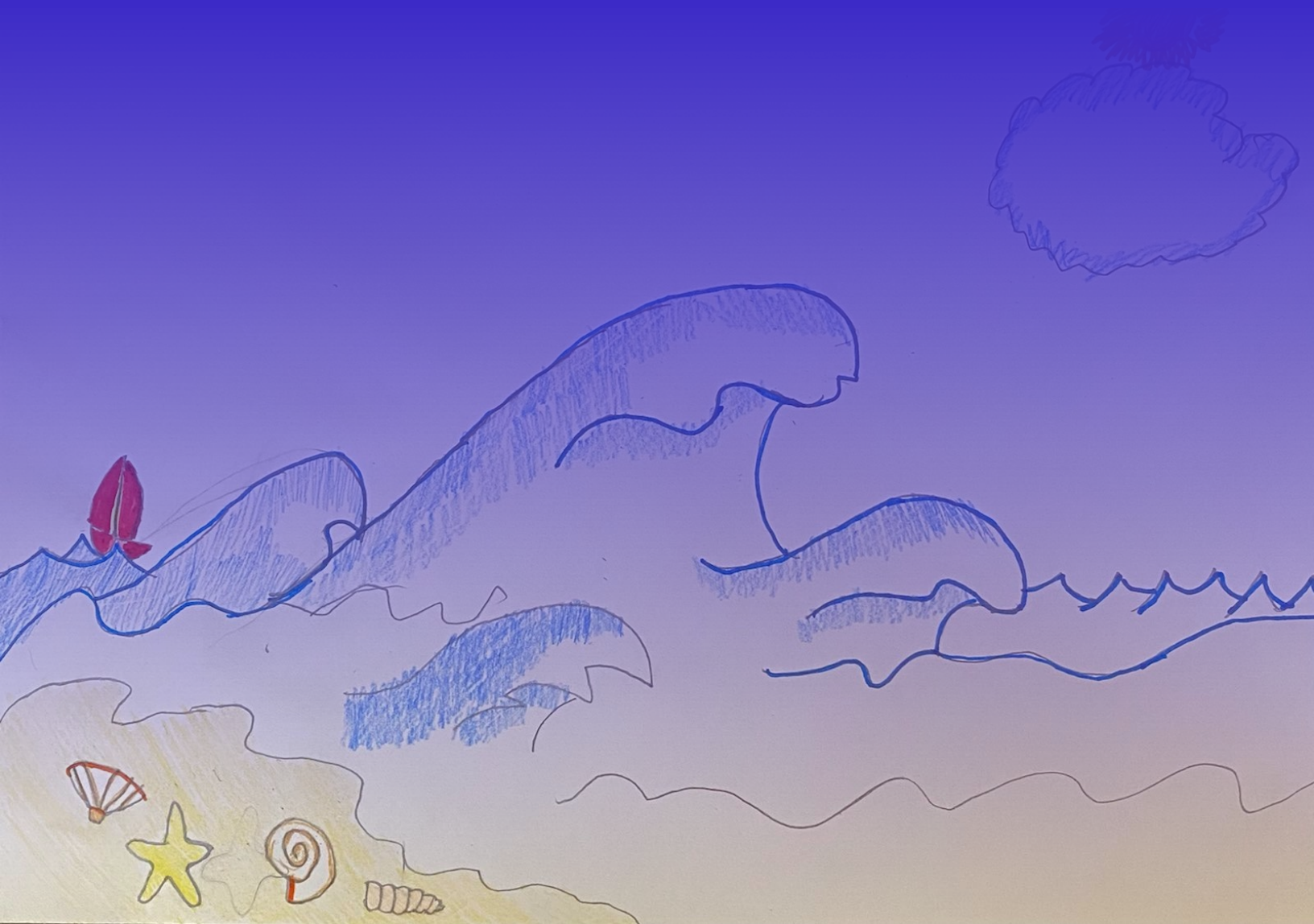

Introduction to Programming in Python

by David Oniani



Contents

1	Data Types	3
1.1	int	3
1.2	float	3
1.3	bool	3
2	The print Function	4

1 Data Types

We have different kinds of built-in *things* in Python. These *things* are called data types. Here is their table:

Data Type	Category	Examples
<code>int</code>	Integers	1, 2, 3, -1, -2, -3, 0
<code>float</code>	Floating point numbers	0.5, 1.5, 2.5, -0.5, -1.5, -2.5
<code>bool</code>	Boolean	<code>False</code> , <code>True</code>
<code>complex</code>	Complex numbers	$1 + i$, $2 + 2i$, $-3 + 5i$
<code>str</code>	Text sequence type	<code>"a"</code> , <code>"abc"</code> , <code>"Hello, world!"</code>
<code>list</code>	Sequence type	<code>[]</code> , <code>[0]</code> , <code>[1, 2, 3]</code> , <code>["ab", "bc", "cd"]</code>
<code>tuple</code>	Sequence type	<code>()</code> , <code>(0)</code> , <code>(1, 2, 3)</code> , <code>("ab", "bc", "cd")</code>
<code>range</code>	Sequence type	<code>range(10)</code> , <code>range(1, 10)</code> , <code>range(3, 8, 2)</code>
<code>set</code>	Set type	<code>{}</code> , <code>{0}</code> , <code>{1, 2, 3}</code> , <code>{"ab", "bc", "cd"}</code>
<code>frozenset</code>	Set type	<code>frozenset({})</code> , <code>frozenset({0, 1})</code>
<code>dict</code>	Mapping type	<code>{}</code> , <code>{0: 1}</code> , <code>{"a": 2, "b": 3}</code>

Table 1: Built-in Data Types

1.1 `int`

Counting is something that we all do in our everyday lives. It would be rather inconvenient if Python did not provide support for counting numbers (i.e., 1, 2, 3, etc.).

On the other hand, counting numbers also have their negative counterparts and there's also zero. Having them is equally important.

Positive counting numbers, their counterparts, and zero, together make a set of numbers called integers!

Hence, we need to be able to represent integers. Python has `int` data type for this purpose. Python's `int` data type can represent negative integers, zero, and positive integers.

1.2 `float`

While we do have integers covered, we have not covered decimals. Python has our back! `float` can be used to represent both positive and negative decimals such as -0.5 and 0.5.

1.3 `bool`

`bool` represents a boolean type. Boolean can only take two values - `True` and `False`. That is all we need to know about this type.

2 The `print` Function

Just like in math, we have functions in Python as well. While we will cover Python functions in a greater detail in later chapters, the `print` function is so useful that we will start by learning how it works!

```
>>> print("Hello, world!")  
Hello, world!
```

The way `print` works is that you write out these characters `p-r-i-n-t`, followed by the left paren `(`, followed by whatever we want to print, and finally the right paren `)`.

```
>>> print(0)  
0  
>>> print(1)  
1  
>>> print(9)  
9  
>>> print(10)  
10
```