

VALUES

The Python Bakery

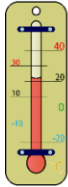
Values Represent the World



Programming is all about inputting data so that we can manipulate it to get some output.

Therefore, we use values to represent the real world as data in the computer.

Values



→ 23.4



→ 74



"It was the best of
times, it was the
worst of times."

Values can represent anything in the real world that we can measure and concretely describe.

Distances, temperatures, peoples' ages, our names, even the text of entire books.

Writing Values

```
74  
23.4  
"Hello world"
```

When we write values directly in code, we call them "Literal Values", as opposed to received as input from a user.

The values are literally written there in the code.

Sometimes we also call this "Hard Coding a value".

These hard-coded values are useful for developing simple programs.

However, they are not very exciting on their own.

If you run this program, then nothing will appear in the console.

We have not told Python to print anything, and we have not requested any kind of user input.

Printing Values

```
print(74)
print(23.4)
print("Hello world")
```

Once we have put data into the computer as values, we need to do something with those values.

A very simple action is to print the values out.

Soon, we will learn more complex actions.

Look at these examples of how we can write literal values and then print them out.

Abstraction

- Abstraction: Representing the real world as data by ignoring details

When Computer Scientists use data to represent reality, they are necessarily removing details.

When we use the value 74 to represent someone's age, we are not including any information about

how they lived those 74 years. If we describe the current temperature as 23.4, then the computer

does not understand whether that is hot or cold.

This process of reducing detail in order to make the information easier to fit inside a computer is called "Abstraction".

Although that word might sound a little complicated, the definition we have is quite simple.

Abstraction is the process of representing information about the real world inside of a computer, by choosing to ignore unnecessary details.

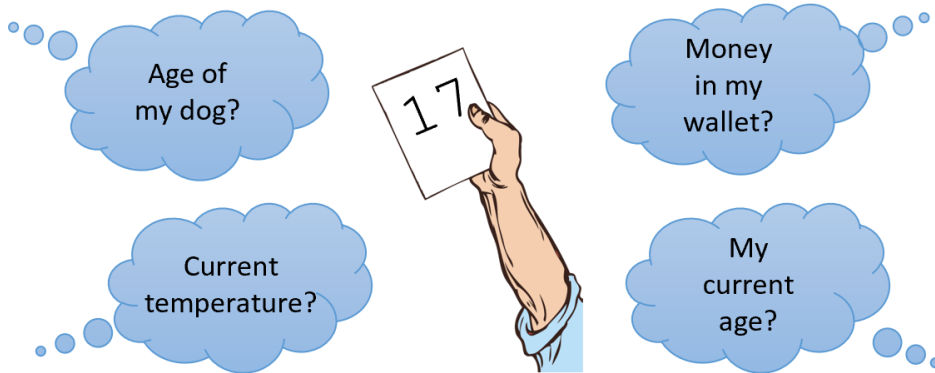
The details that will be necessary or unnecessary are relative to whatever problem the program is meant to solve.

In order to determine how old someone will be next year, we only need to know how old they are - not how they celebrated their birthday last year, or how many times they have been to Canada.

By abstracting away those other details, it is easy to represent someone's age in the

computer.

Understanding Values



It is very easy to put literal values into a program.

However, it is very difficult to immediately understand what those values mean.

It is up to us programmers to communicate the meaning of values.

The value 17 could be used to represent the age of a person, or the age of a dog, or the amount

of money in a wallet, or the number of times they have traveled to Canada.

We will learn more ways to write programs so that it is easier to understand what a value represents.

For now, though, it is important to understand that values on their own have no meaning.