INPUT AND OUTPUT

INPUT AND OUTPUT

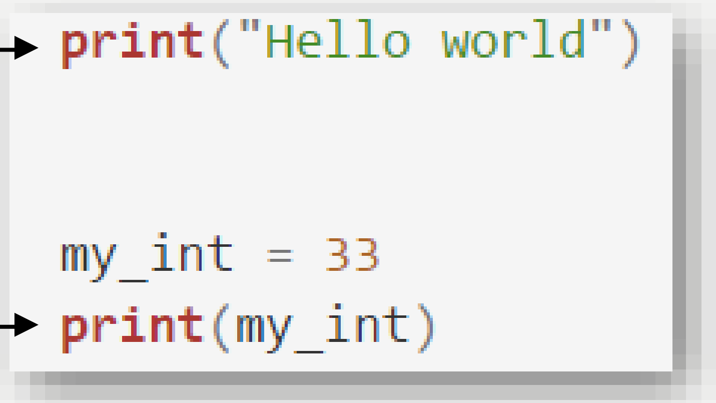
- We want the users of our programs to be able to interact with the programs we build
- We can do that by inputing data to our programs and making the programs output data to the screen(or to a file or a printer)
- We can use two functions(ísl. fall) for input and output
  - These functions are called **input()** and **print()**
  - **You might be thinking, what is a function?**
    - Function will be explained in detail later in the course but for now you can think of them as tiny programs that do a particular task very well
    - Lets see some examples!

INPUT AND OUTPUT

- Lets start with the print function
  - The print function will print whatever is put between the parentheses to the screen

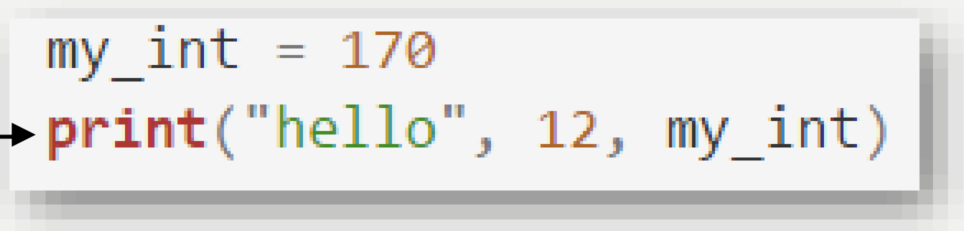
The string "Hello world" will be printed to the screen

The integer 33 will be printed to the screen



```
print("Hello world")


my_int = 33
print(my_int)
```

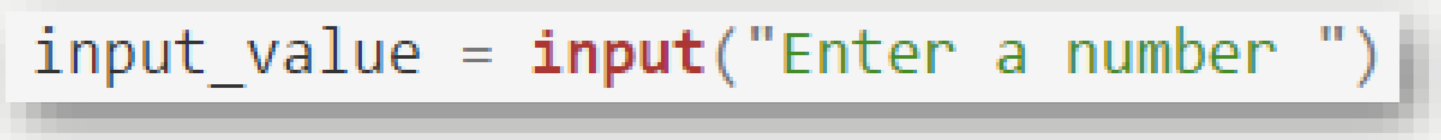
We can pass multiple values to the print function. But we must seperate them with a comma



```
my_int = 170
print("hello", 12, my_int)
```

INPUT AND OUTPUT

Consider the following code



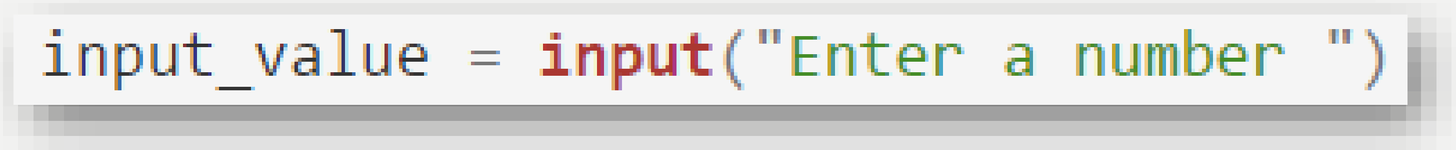
```
input_value = input("Enter a number ")
```

The input function will print "Enter an number " to the screen and wait until the user types something, ending with Enter

- Whatever the user types will then be stored in the variable input_value
- There is however one thing we must always bare in mind when working with the input function. The value the user types will always be stored as the type **string**!
  - So even if the users input the number 1 it will be stored as the string '1' but not as the integer 1

INPUT AND OUTPUT

- Sometimes we want the input to be a string but sometimes we don´t
- In this example we probably want the input to be an integer



```
input_value = input("Enter a number ")
```

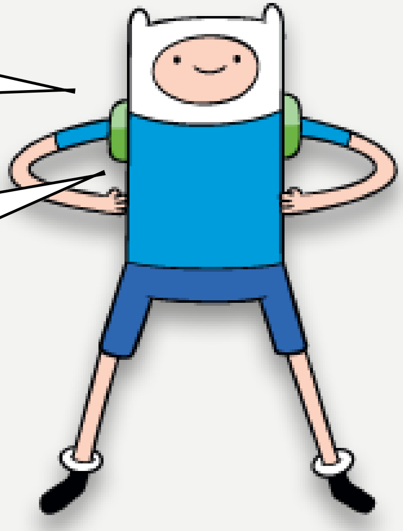
- In Python we can convert values to a different type with several function
  - int(some_value) will convert some_value to an integer
  - float(some_value) will convert some_value to a floating point number
  - str(some_value) will convert some_value to a string

INPUT AND OUTPUT

But why do we need to convert types when receiving input from the user?

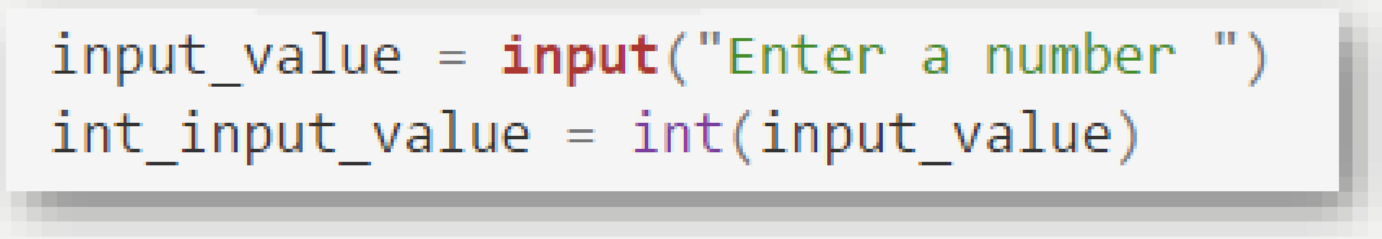
We don´t always need to convert types when receiving input from the user. It depends on what you are going to do with the input!

For example if you are going to do some arithmetic operations with the input you will need to convert it to an int or float because it is not possible to do arithmetic with strings



INPUT AND OUTPUT

- Consider the following code, lets say that the user inputs 12
  - The value 12 will be stored as the string "12" in the variable input_value
  - In the next line we call the int() function and pass it the value of input_value.
    The int() function will convert the value "12" to the integer 12 and store it in the variable int_input_value



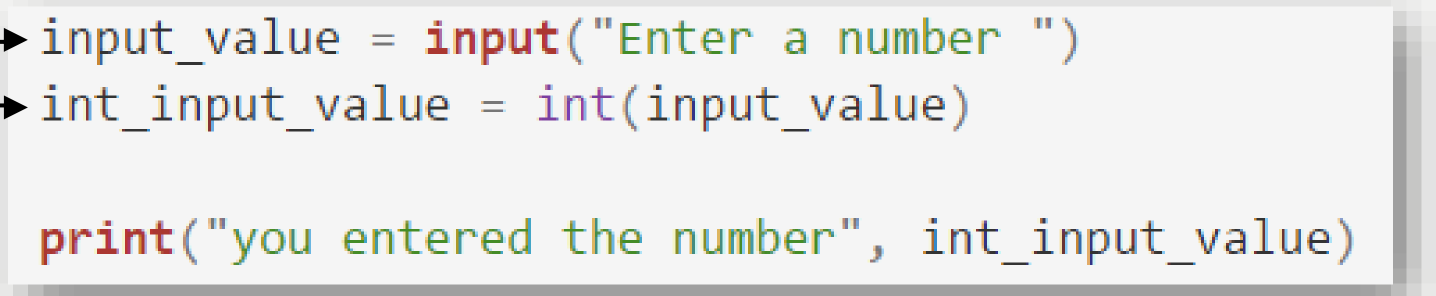
```
input_value = input("Enter a number ")
int_input_value = int(input_value)
```

INPUT AND OUTPUT

- Consider this example

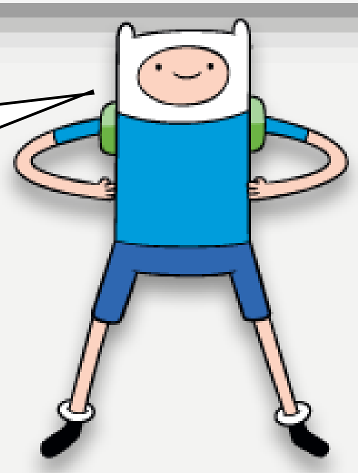
The value the user types in will be stored as a string in the variable input_value

Here we convert the value of the variable input_value to an integer and store it in the variable int_input_value



```
input_value = input("Enter a number ")
int_input_value = int(input_value)

print("you entered the number", int_input_value)
```

Don´t worry! We will see more examples of type conversion and how useful it can be later in the course



POP QUIZ

- What does the print() function do?
- Can you put multiple values between the parentheses of the print() function?
- What type does the input() function return?
- Are "2" and 2 the same value?
- Are 3 and 3.0 the same thing?
- What will str(5) return?
- What will int("5") return?

POP QUIZ - ANSWERS

- What does the print() function do?

  It prints whatever is put between the parentheses
- Can you put multiple values between the parentheses of the print() function?

  Yes
- What type does the input() function return?

  A string
- Is "2" and 2 the same value?

  No
- Is 3 and 3.0 the same thing?

  No
- What will str(5) return?

  The string "5"
- What will int("5") return?

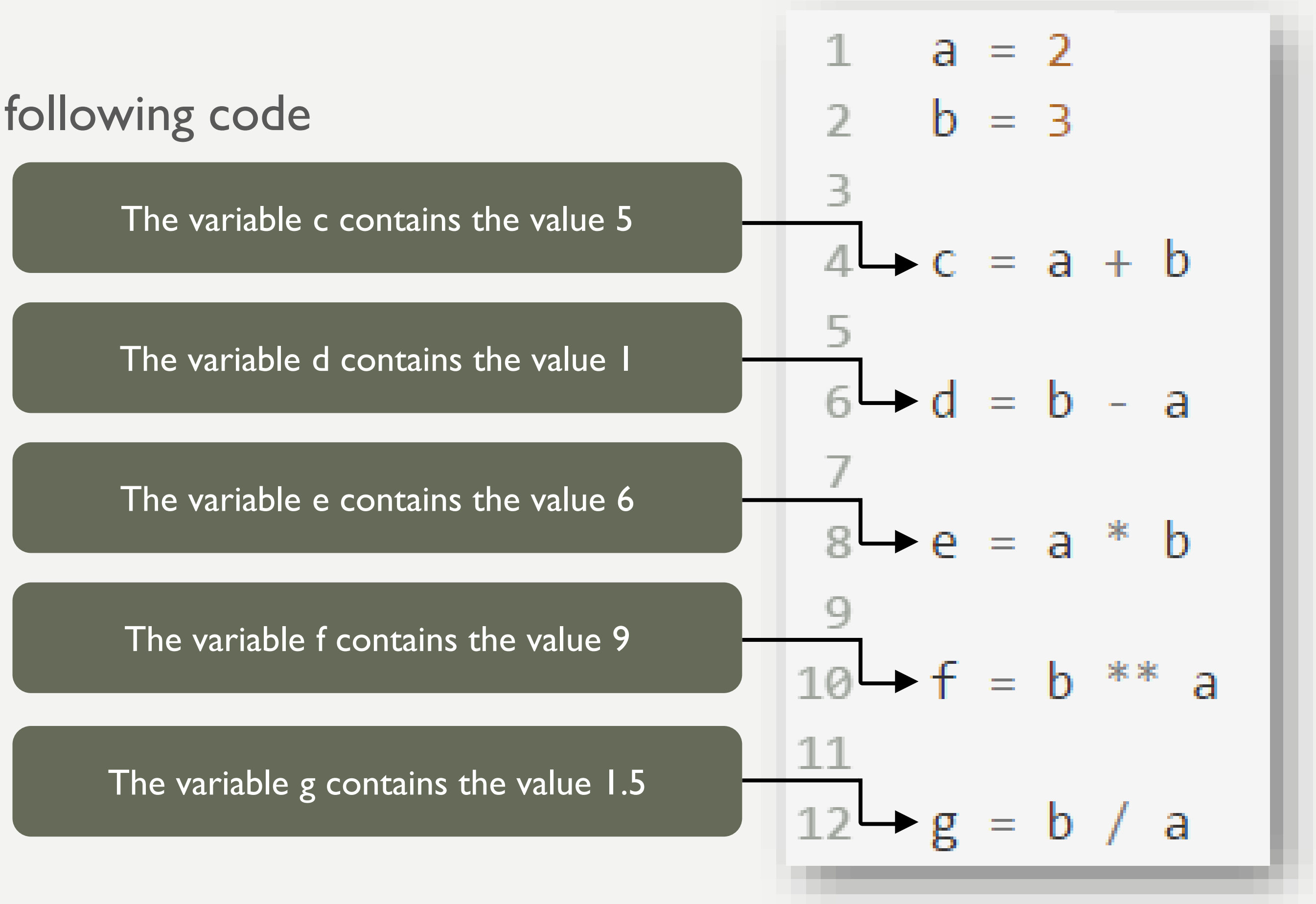
  The integer 5

ARITHMETIC IN PROGRAMMING

- In Python we can use all the basic arithmetic operators and some Python specific operators to do all the basic math operations
- We can use
    - + for addition
    - - for subtraction
    - * for multiplication
    - / for division
    - // for integer division (also called floor division)
    - % for modulo (remainder, ísl. afgangur)
    - ** for exponent (ísl. Hefja í veldi)

ARITHMETIC IN PROGRAMMING

- Consider the following code



The variable c contains the value 5

The variable d contains the value 1

The variable e contains the value 6

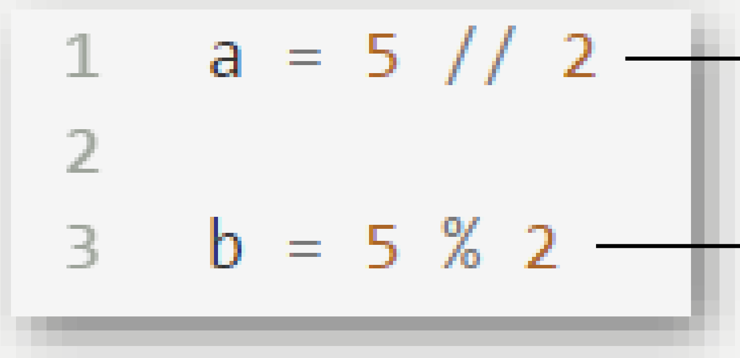
The variable f contains the value 9

The variable g contains the value 1.5

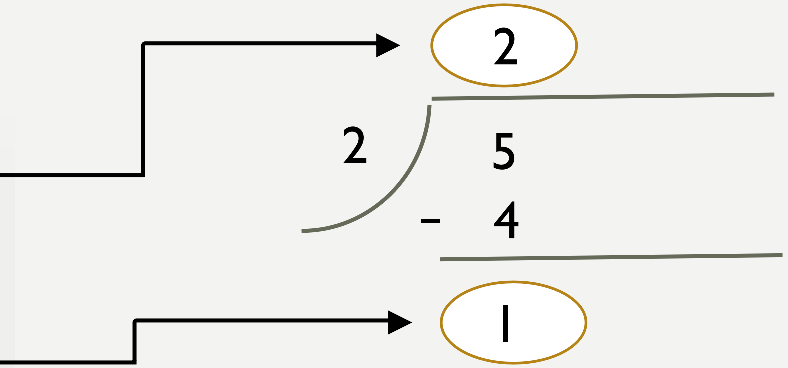
```
a = 2
b = 3

c = a + b

d = b - a

e = a * b

f = b ** a

g = b / a
```

ARITHMETIC IN PROGRAMMING

- The integer division(//) and modulo(%) can be a bit confusing for beginners
- They perform the same arithmatic operation but return a different result
  - The // returns the integer quotient
  - The % returns the remainder

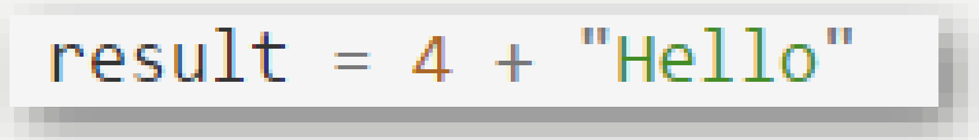


```
a = 5 // 2

b = 5 % 2
```



ARITHMETIC AND TYPES

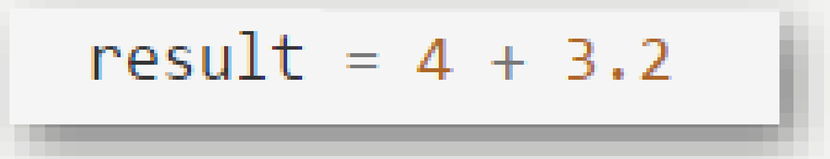
- What happens if we do arithmetic operations on different types such as:



```
result = 4 + "Hello"
```

  – Trying to do add the integer 4 to the string "Hello,, will result in an error and the program will crash

- What about working with integers and floats interchangably, like this:



```
result = 4 + 3.2
```

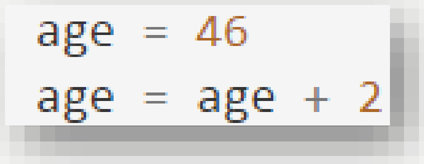
  – This kind of arithmetic is performed so that no data will be lost, i.e the integer 4 will be cast to a float value of 4.0 and then the arithmetic is performed so the value of result will be the floating point number 7.2

ARITHMETIC PRECEDENCE RULES

- Computers are pretty good at math and use the same precence rules as are used in mathematics
- 18 / 3 * 5
  - Division and multiplication have the same precedence value so we go from left to right and start with the division
- = 6 * 5
- = 30
- 18 / (3 * 2)
  - parentheses have a higher precedence than division so we start with what´s inside them
- = 18 / 6
- = 3

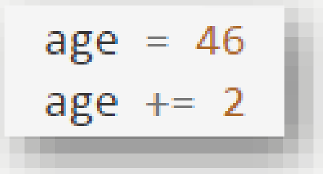
OPERATOR SHORTHANDS

- Some expressions occur so often in programming that many programming languages contains shorthand operators for them
- All arithmetic operators can be used this way
  - +=



```
age = 46
age = age + 2
```

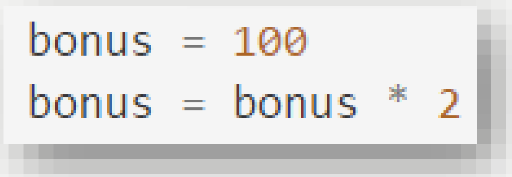
=



```
age = 46
age += 2
```

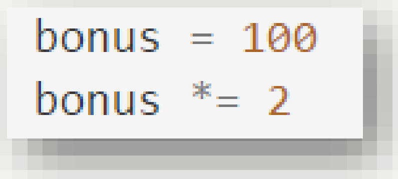
The variable age will store the value 48

  - *=



```
bonus = 100
bonus = bonus * 2
```

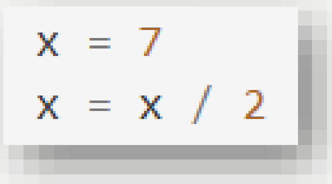
=



```
bonus = 100
bonus *= 2
```

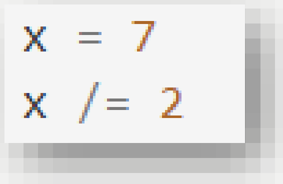
The variable bonus will store the value 200

  - /=



```
x = 7
x = x / 2
```

=



```
x = 7
x /= 2
```

The variable x will store the value 3.5