Parameters and arguments

Parameters and arguments

A **parameter** is a variable in a method definition. After the name of the method, zero or more parameters are defined.

When a method is called, the **arguments** are the data you pass into the method's parameters.

For examples of methods with parameters – methods that receive arguments – see Methods

- When calling a method, arguments are passed to the method. Which arguments can be passed depends on which parameters have been defined in the method definition.

- When more than one parameter is needed, the arguments must be passed in sequence, or by mentioning the name of the parameter. In the case of a named parameter, this parameter and its value are separated by a colon.

    - Example 1: passing arguments in sequence:
      methodOne("", 0, false, 12); Even when the first three paramaters have default values, when passing the fourth (12), the others must be passed first.

    - Example 2: passing arguments to named parameters:
      methodOne(amount: 12); The fourth parameter (named amount) is passed directly with value 12.

- Often the parameters are complex types (combined, existing of more than one value). They are classes that need to be instantiated with new.

- Example: maxSize: new Size(10, 10); The (named) parameter maxSize takes as arguments a class, named Size(). Between brackets the constructor parameters of this class will be passed, in our example 10 and 10 for width and height.