

Method Overriding

All Versions

Examples

Basic method overriding

Here is an example of basic overriding in Python (for the sake of clarity and compatibility with both Python 2 and 3, using [@ new style class](#) and `print with ()`):

```
class Parent(object):
    def introduce(self):
        print("Hello!")

    def print_name(self):
        print("Parent")

class Child(Parent):
    def print_name(self):
        print("Child")

p = Parent()
c = Child()

p.introduce()
p.print_name()

c.introduce()
c.print_name()

$ python basic_override.py
Hello!
Parent
Hello!
Child
```

When the `Child` class is created, it inherits the methods of the `Parent` class. This means that any methods that the parent class has, the child class will also have. In the example, the `introduce` is defined for the `Child` class because it is defined for `Parent`, despite not being defined explicitly in the class definition of `Child`.

In this example, the overriding occurs when `Child` defines its own `print_name` method. If this method was not declared, then `c.print_name()` would have printed "Parent". However, `Child` has overridden the `Parent`'s definition of `print_name`, and so now upon calling `c.print_name()`, the word "Child" is printed.

Syntax

Parameters

Remarks