Method Overriding in Python

Robots

In the template below you can see a hierarchy of robots. There is a base class `Robot` and a subclass `ServiceRobot`. Objects of the class `Robot` have two attributes: `name` and `variety`. variety represents the function of the robot, for example, service or military. The `get_info` method returns information about a particular instance.

The `ServiceRobot` represents the **"service"** variety of robots. Objects of this class have only the `name` attribute.

Your task is to change the `__init__` method of the class `ServiceRobot` so that `get_info()` returns the correct information for the objects of `ServiceRobot`. There are several ways to do it, but we ask you to use the `super()` function.

In [13]:
```
class Robot:
    def __init__(self, name, variety):
        self.name = name
        self.variety = variety
        print("Robot")

    def get_info(self):
        return "{} is a {} robot".format(self.name, self.variety)


class ServiceRobot(Robot):
    def __init__(self, name):
        self.name = name


chappi = ServiceRobot("Chappi")
print(chappi.get_info())
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[13], line 17
     13         self.name = name
     16 chappi = ServiceRobot("Chappi")
---> 17 print(chappi.get_info())

Cell In[13], line 8, in Robot.get_info(self)
      7 def get_info(self):
----> 8     return "{} is a {} robot".format(self.name, self.variety)

AttributeError: 'ServiceRobot' object has no attribute 'variety'
```

Source: JetBrains Academy