# Did You Really Understand Super?

Damien Cassou, Stéphane Ducasse and Luc Fabresse

W6S01

# What You Will Learn

Revisit

- super
- Message lookup
- Class methods

# A Little Puzzle

```
Die class >> new

    | inst |
    inst := super new.
    inst initialize.
    ^ inst
```

We execute the following expression: Die new

# Questions

```
Die class >> new

| inst |
inst := super new.
inst initialize.
^ inst
```

- What is inst?
- What is super?
- What is super new?

# Hint: super is Not...

```
Die class >> new

    | inst |
    inst := super new.
    inst initialize.
    ^ inst
```

- No super is not the superclass
- No inst is not an instance of the superclass

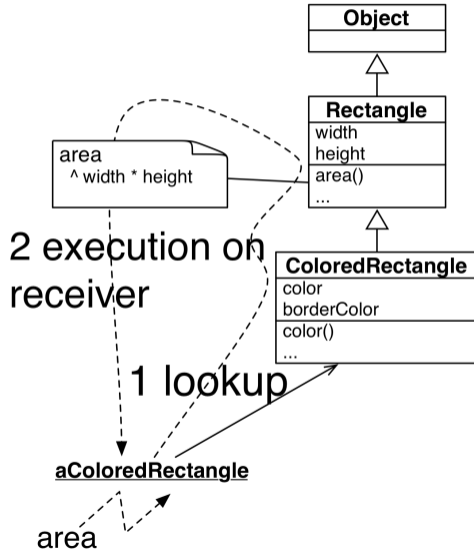# Hint 2: super is the Message Receiver

```
Die class >> new

  | inst |
  inst := super new.
  inst initialize.
  ^ inst
```
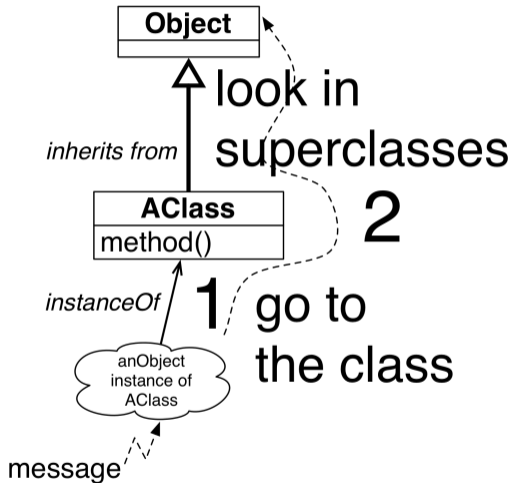
- The message is Die new
- So the receiver is the class Die

# Sending a Message: Lookup + Apply on Receiver



Object

Rectangle
width
height
area()
...

area
^ width * height

ColoredRectangle
color
borderColor
color()
...

2 execution on receiver

1 lookup

aColoredRectangle

area

# Remember: Method Lookup



**Object**

look in
superclasses

*inherits from*

**2**

**AClass**
method()

*instanceOf*

**1** go to
the class

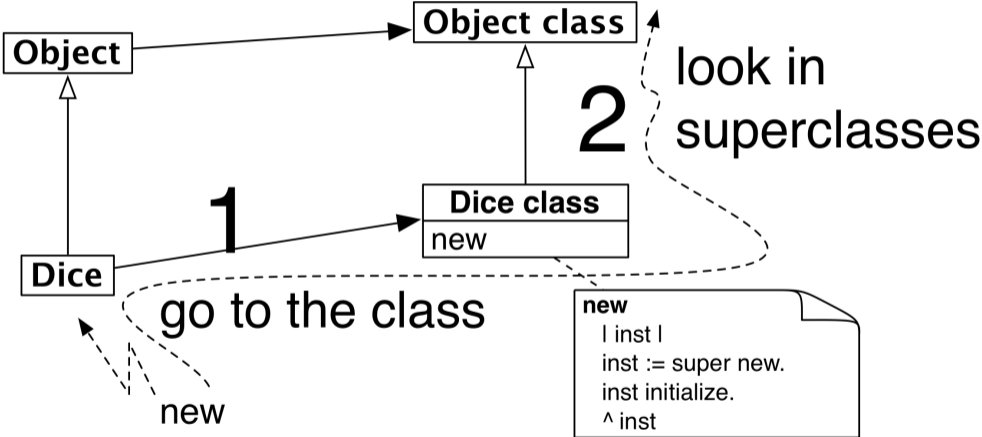anObject
instance of
AClass

message

# Solution

```
Die class >> new

    | inst |
    inst := super new.
    inst initialize.
    ^ inst
```

- super is the receiver: the class Die
- Look for new in the superclass of the class Die class (Pay attention not Die)
- Once found we apply to the receiver: Die
- We get an instance of the class Die and send it initialize and return it

# Solution

# Summary

- Sending a message is looking up for the method and applying it on the receiver
- Now you should really understand super :)
- super is the receiver of the message and the method lookup starts in the superclass of the class containing the expression

# Challenge

Imagine we have:

```
A >> foo
  ^ super class == self class
```

What is the result of A new foo and why?

A course by



and



in collaboration with