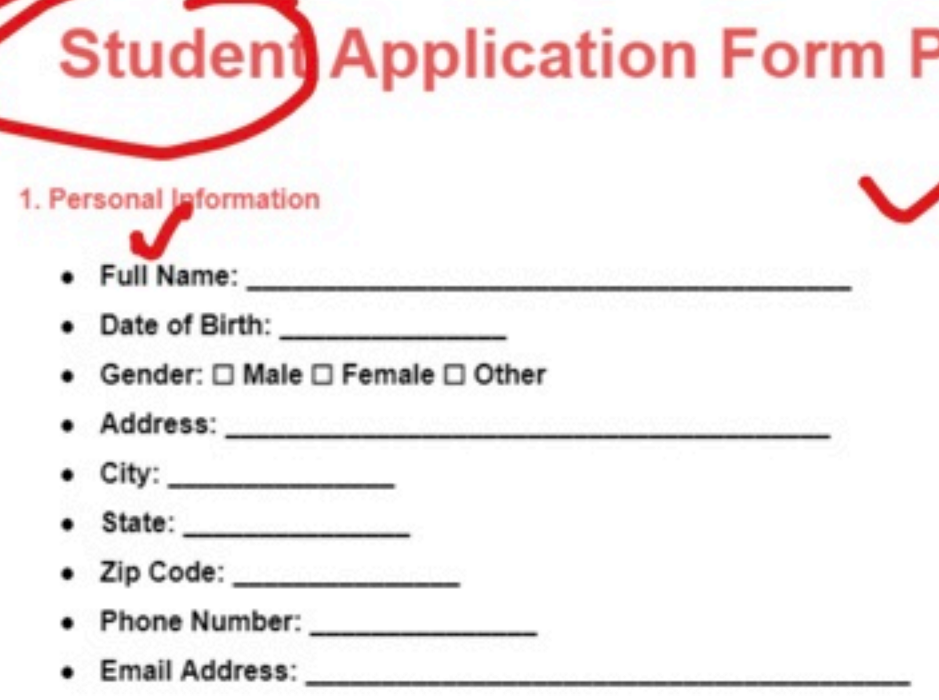
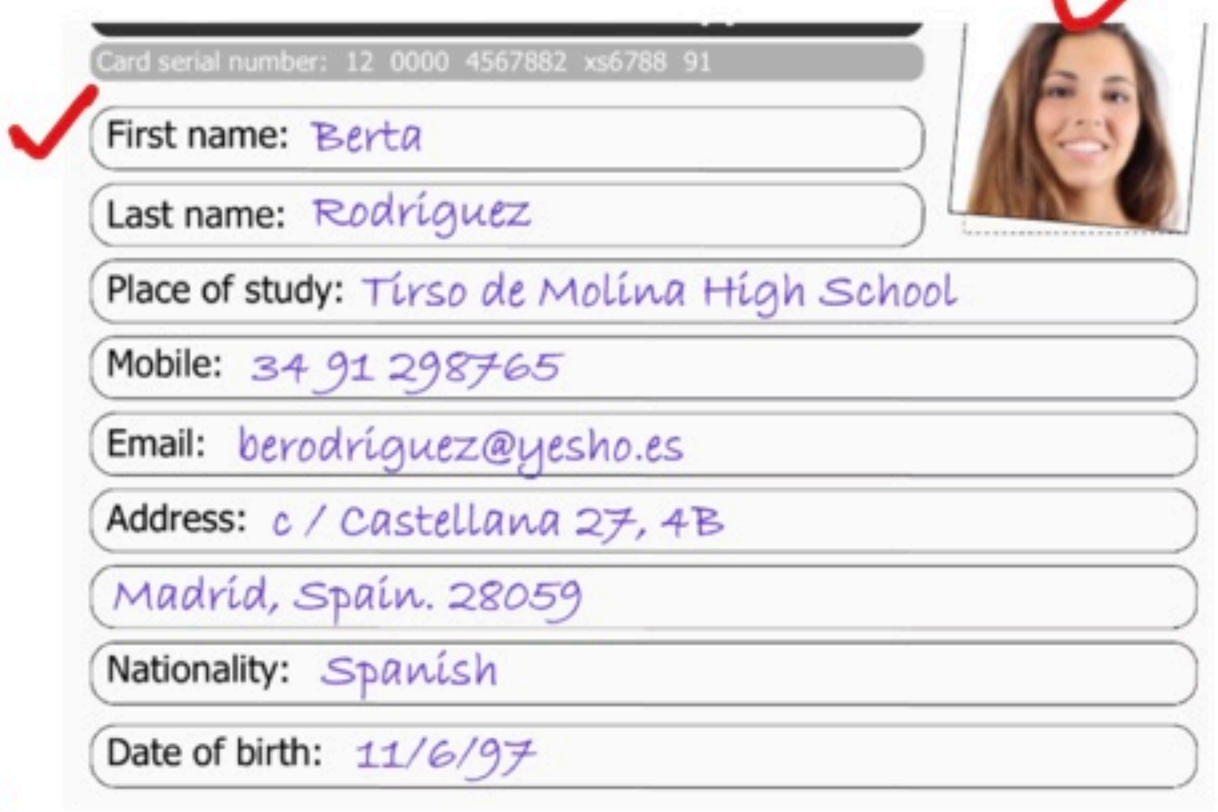
Class



Student Application Form PDF

1. Personal Information

- Full Name: ______________________________
- Date of Birth: ___________
- Gender: □ Male □ Female □ Other
- Address: ______________________________
- City: ___________
- State: ___________
- Zip Code: ___________
- Phone Number: ___________
- Email Address: ______________________________



Card serial number: 12 0000 4567882 xs6788 91

First name: Berta

Last name: Rodriguez

Place of study: Tirso de Molina High School

Mobile: 34 91 298765

Email: berodriguez@yesho.es

Address: c / Castellana 27, 4B

Madrid, Spain. 28059

Nationality: Spanish

Date of birth: 11/6/97

blue print

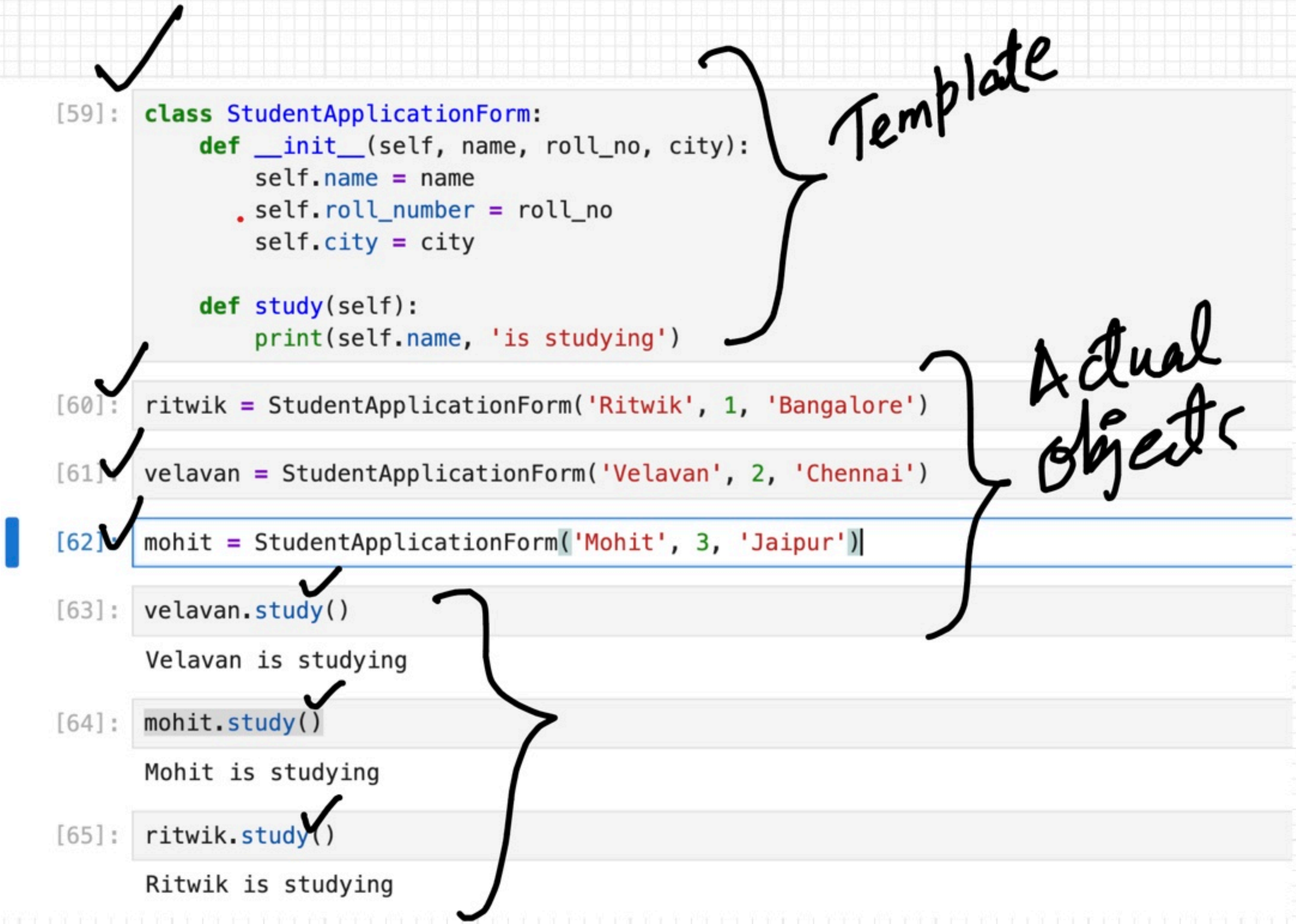
template

CLASS

More object

Object of

Class "Student"



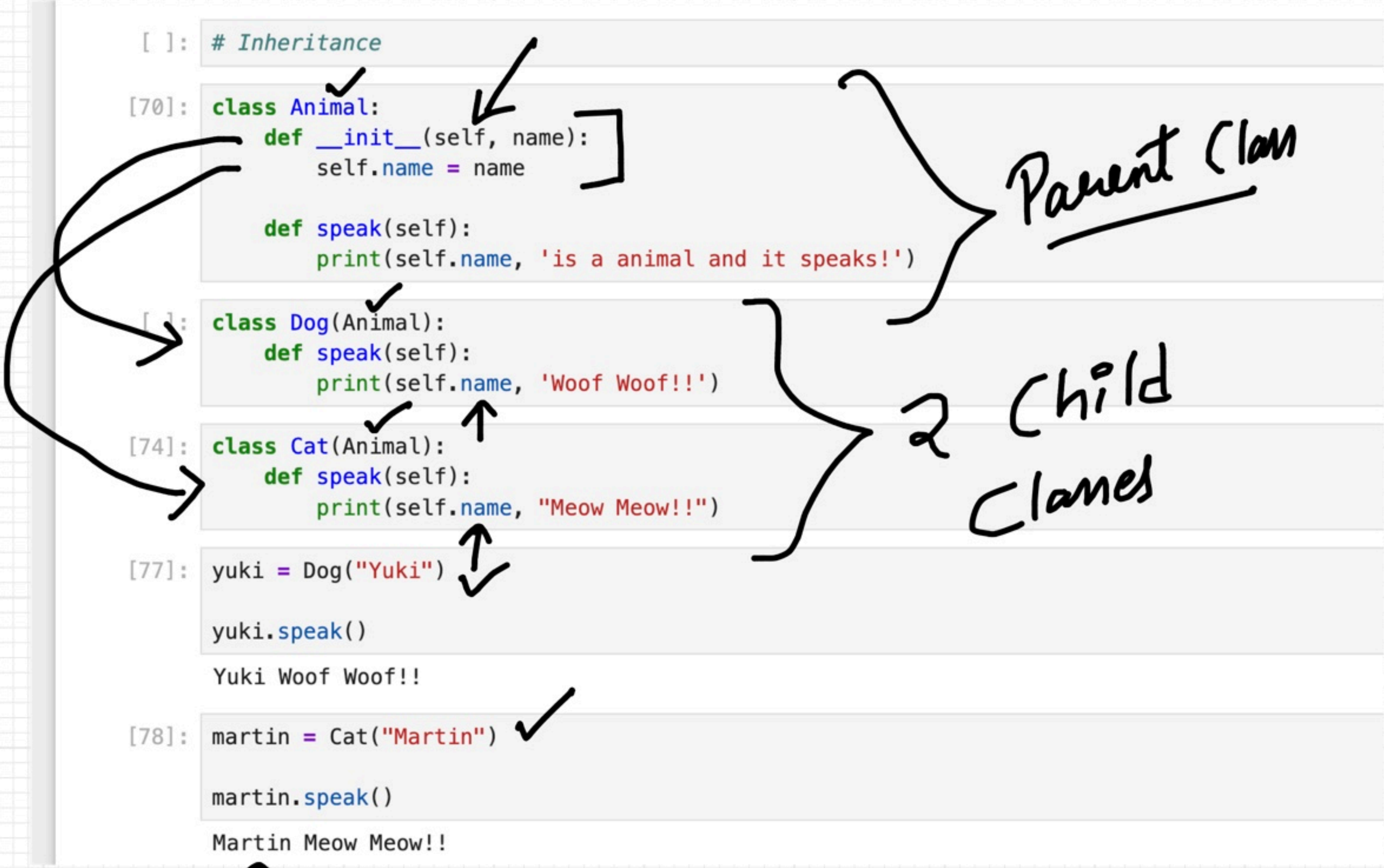
```
[59]: class StudentApplicationForm:
          def __init__(self, name, roll_no, city):
              self.name = name
              self.roll_number = roll_no
              self.city = city

          def study(self):
              print(self.name, 'is studying')
```

Template

```
[60]: ritwik = StudentApplicationForm('Ritwik', 1, 'Bangalore')

[61]: velavan = StudentApplicationForm('Velavan', 2, 'Chennai')

[62]: mohit = StudentApplicationForm('Mohit', 3, 'Jaipur')
```

Actual objects

```
[63]: velavan.study()

      Velavan is studying

[64]: mohit.study()

      Mohit is studying

[65]: ritwik.study()

      Ritwik is studying
```



```
[ ]: # Inheritance
```

```
[70]: class Animal:
          def __init__(self, name):
              self.name = name

          def speak(self):
              print(self.name, 'is a animal and it speaks!')
```

```
[ ]: class Dog(Animal):
         def speak(self):
             print(self.name, 'Woof Woof!!')
```

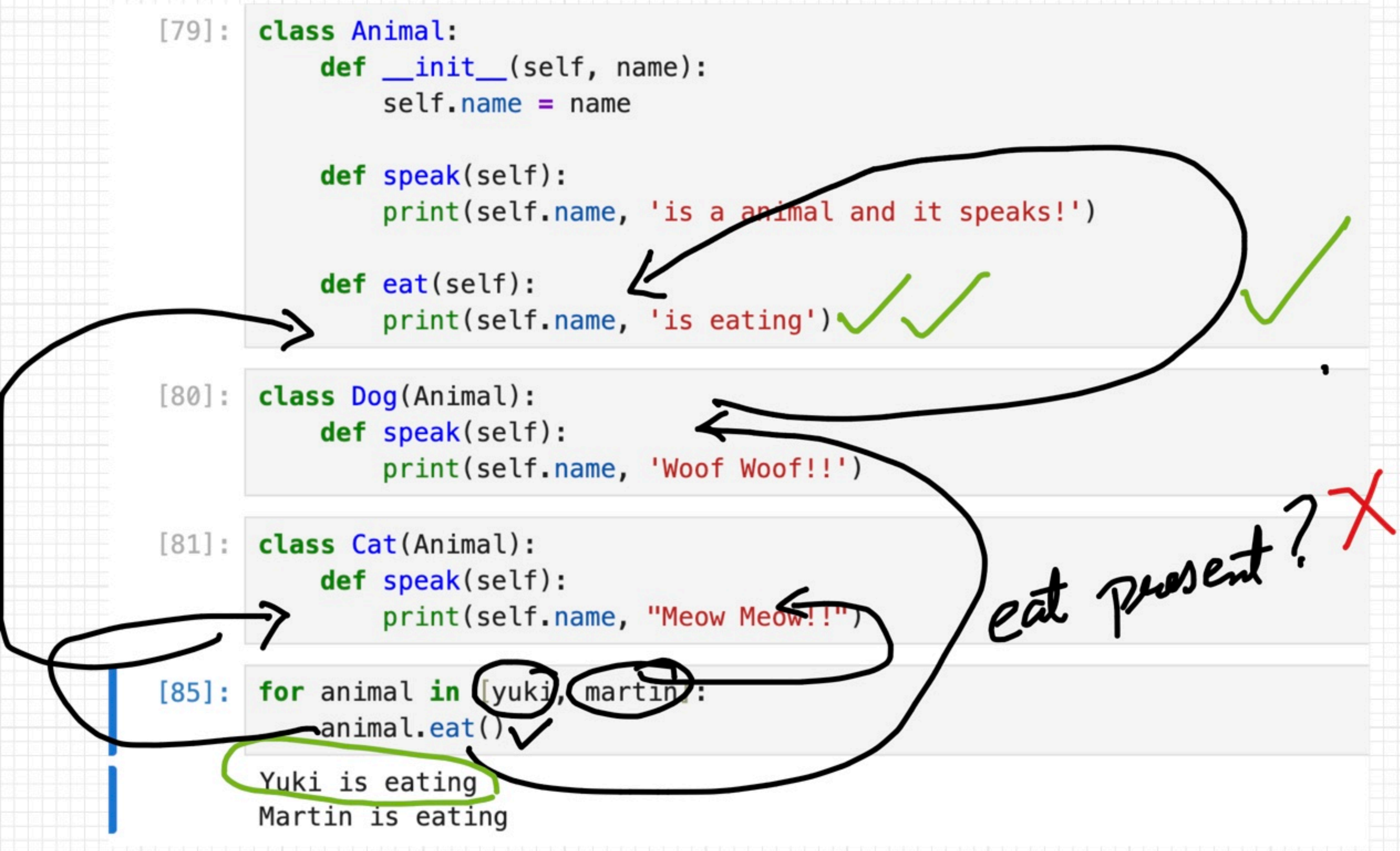
```
[74]: class Cat(Animal):
          def speak(self):
              print(self.name, "Meow Meow!!")
```

```
[77]: yuki = Dog("Yuki")

      yuki.speak()

      Yuki Woof Woof!!
```

```
[78]: martin = Cat("Martin")

      martin.speak()

      Martin Meow Meow!!
```

Parent Class

2 Child Classes



```
[79]: class Animal:
          def __init__(self, name):
              self.name = name

          def speak(self):
              print(self.name, 'is a animal and it speaks!')

          def eat(self):
              print(self.name, 'is eating')
```

```
[80]: class Dog(Animal):
          def speak(self):
              print(self.name, 'Woof Woof!!')
```

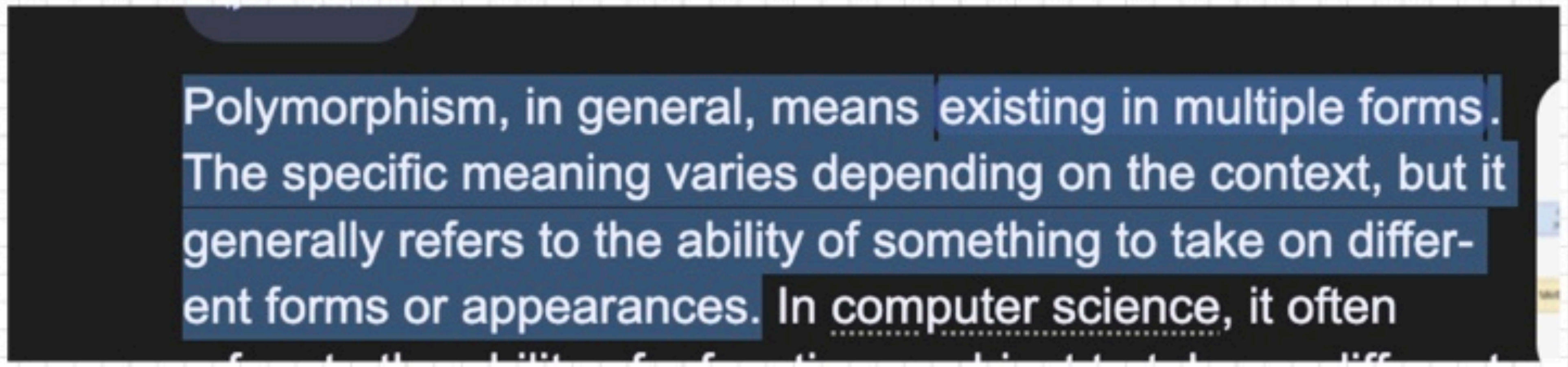
```
[81]: class Cat(Animal):
          def speak(self):
              print(self.name, "Meow Meow!!")
```

```
[85]: for animal in [yuki, martin]:
          animal.eat()
```

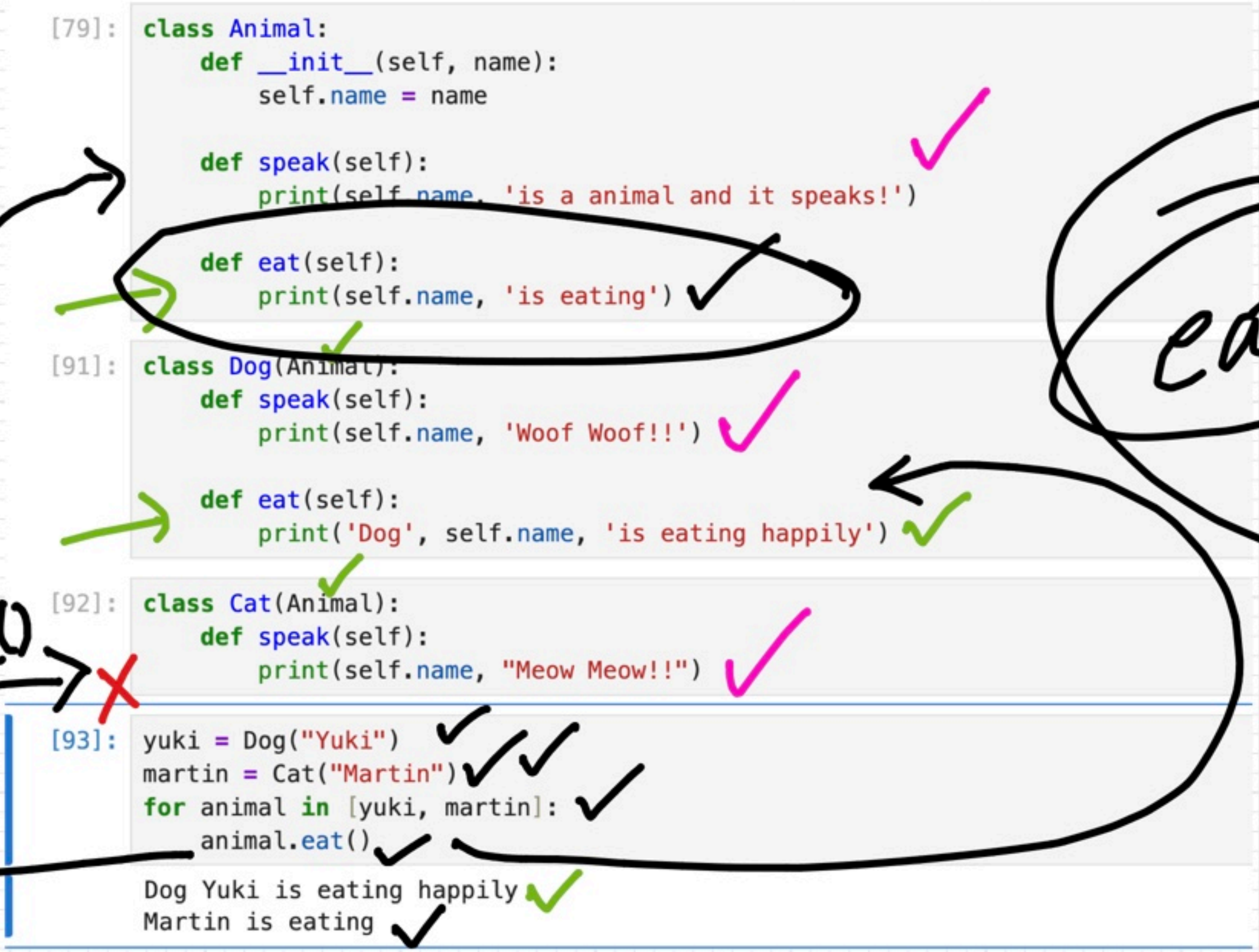
```
Yuki is eating
Martin is eating
```

eat present? ✗

Override



Polymorphism, in general, means existing in multiple forms. The specific meaning varies depending on the context, but it generally refers to the ability of something to take on different forms or appearances. In computer science, it often



```
[79]: class Animal:
          def __init__(self, name):
              self.name = name

          def speak(self):
              print(self.name, 'is a animal and it speaks!')

          def eat(self):
              print(self.name, 'is eating')
```

```
[91]: class Dog(Animal):
          def speak(self):
              print(self.name, 'Woof Woof!!')

          def eat(self):
              print('Dog', self.name, 'is eating happily')
```

```
[92]: class Cat(Animal):
          def speak(self):
              print(self.name, "Meow Meow!!")
```

```
[93]: yuki = Dog("Yuki")
      martin = Cat("Martin")
      for animal in [yuki, martin]:
          animal.eat()

      Dog Yuki is eating happily
      Martin is eating
```

Eat
Parent()

eat() is behaving different for diff object

Eat() present