

sen hiç nesne gördün mü?



A black and white portrait of a man with long, dark, wavy hair, looking slightly to his right. He is wearing a plaid shirt over a dark t-shirt. The background is a dark, textured surface, possibly a couch or wall. The text "Andaç Karay Kudu" is overlaid in white, sans-serif font across the lower half of the image.


Andaç Karay Kudu

A black and white portrait of a man with long, dark, wavy hair, looking slightly to his right. He is wearing a plaid shirt. The background is dark and out of focus.

# Andaç Karay Kudu – Yazılım Ustası



www: [akkudu.com](http://akkudu.com)  
twitter: [@akkuducom](https://twitter.com/akkuducom)  
email: [hey@akkudu.com](mailto:hey@akkudu.com)



Sorularınızı @akkuducum'a tweetlerseniz  
size daha sonra cevap yazabilirim.  
(şu an meşgulüm)

sen hiç nesne gördün mü?

ruby ile  
nesne tabanlı programlama











Avril 1912 -

report  
des du mois  
reporter -  
divers  
à reporter -

200	4
110	20
80	80
3	52
86	25

Mois d'Avril 1912 -

21. 2 timbres à <sup>report.</sup> 0,25  
23. tram 2 f.  
24. payé note 2 pourboires  
pharmacien malade  
timbre 0,25 + 0,15  
cure oreilles  
crème pour barbe

divers (bain)

60	70
0	50
0	30

46	20
0	70
0	40
0	60
0	80
110	20
3	55
113	75

Total

somut ve soyut kavramların  
bilgisayar ortamında modellenmesi  
temeline dayalı bir disiplindir

ruby?



uby is designed to make programmers happy  
Yukihiro “Matz” Matsumoto

100% nesne tabanlıdır

nesne nedir?  
yenir mi?





nesne, bir değer barındıran  
ve bir tanımlayıcı ile ulaşılan  
hafıza bölgesidir.

1

=> 1

"Ruby"

=> "Ruby"

[1, 2, 3]

=> [1, 2, 3]

one = 1

=> 1

one

=> 1

ruby = "Ruby"

=> "Ruby"

ruby

=> "Ruby"

```
def foo
```

```
  # ...
```

```
end
```

```
=> :foo
```

bu nesneler ile ne yapabiliriz?

Onlarla konuşabiliriz

```
'ruby'.send :upcase  
=> "RUBY"
```

```
2.send :*, 2  
=> 4
```

```
[1, 2, 3].send :nil?  
=> false
```



```
'ruby'.upcase  
=> "RUBY"
```

```
2.*, 2  
=> 4
```

```
[1, 2, 3].nil?  
=> false
```

```
matz = "Matz"
```

```
matz.greet
```

```
=> NoMethodError: undefined method `greet' for "Matz":String
```

```
matz = "Matz"
```

```
def matz.greet  
  puts "Hello, i am #{self}"  
end
```

```
matz.greet  
=> "Hello, i am Matz"
```

nesne, bilgiyi ve işlemi  
bir arada bulunduran yapıdır.

nesne = bilgi + işlem



puts “Think **elegant.**”

nesne = özellik + davranış

iki gönüllüye  
ihtiyacım var!



sinif / class

sınıf nedir?  
geçilir mi?

nesnelerin  
ortak özellik ve davranışlarını  
gruplandırduğımız  
nesne şablonlarıdır.

```
class Person
  attr_accessor :name

  def greet
    puts "Hello, i am #{name}"
  end
end
```

```
matz = Person.new  
person.name = "Matz"  
matz.greet
```

```
=> "hello, i am Matz"
```

```
class Person
  attr_accessor :name

  def initialize(name)
    @name = name
  end
end
```

```
matz = Person.new("Matz")  
matz.greet
```

```
=> "hello, i am Matz"
```





```
class Bird
  attr_accessor :name, :sound

  def initialize(name, sound)
    @name, @sound = name, sound
  end

  def sing
    puts "#{@sound * 3}"
  end
end
```





penguin = Bird.new("Kowalski", WTF?!?)

```
class Bird
  attr_accessor :name

  def initialize(name)
    @name = name
  end
end
```



Bird



Bird

SingingBird

SingingAndFlyingBird



Bird

SingingBird

SingingAndFlyingBird

NonSingingButFlyingBird

Bird

SingingBird

SingingAndFlyingBird

NonSingingButFlyingBird

SingingButNonFlyingBird

Bird

SingingBird

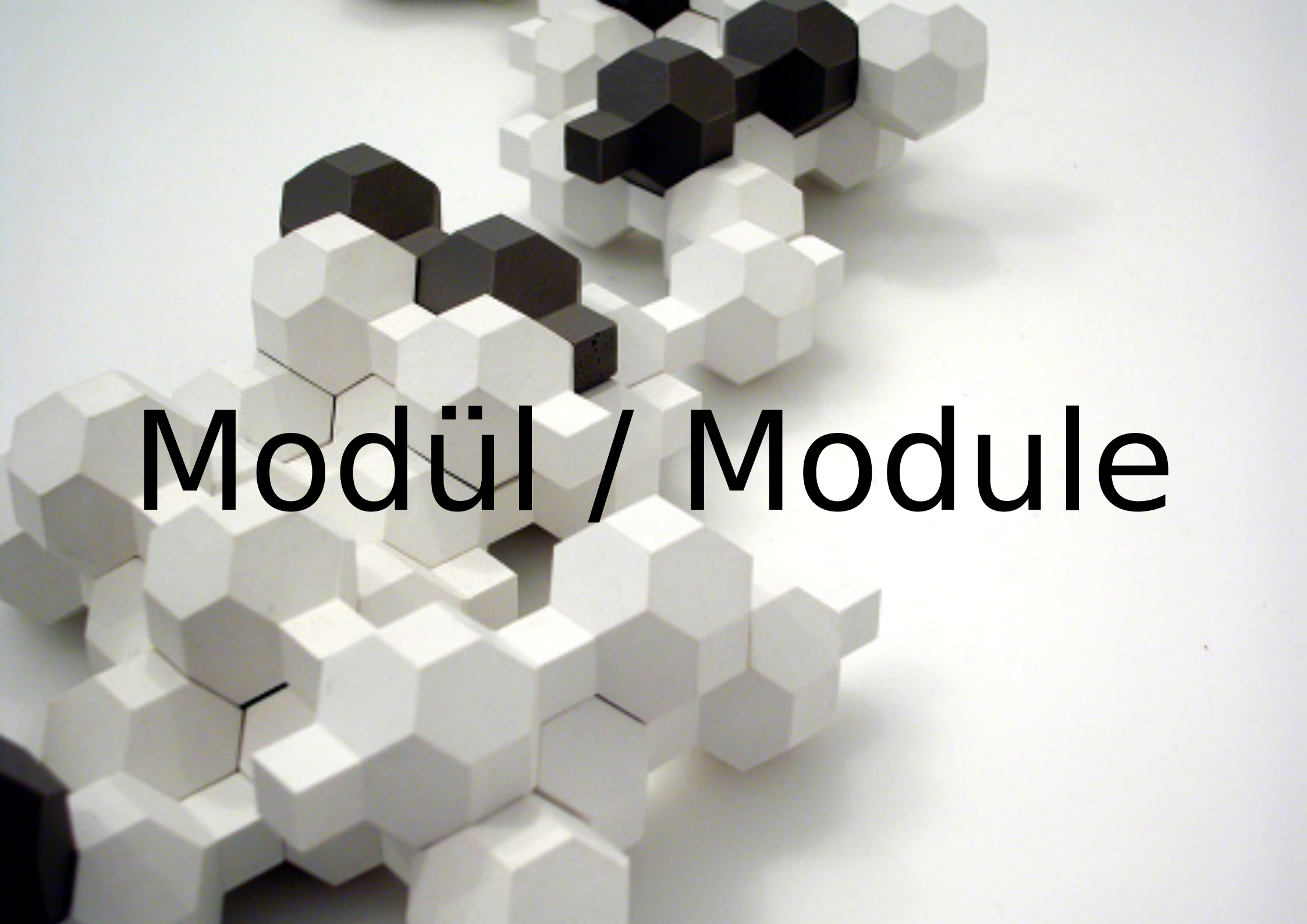
SingingAndFlyingBird

NonSingingButFlyingBird

SingingButNonFlyingBird

NonSingingAndNonFlyingBird

...



Modül / Module

```
module Flier
  def fly
    puts "#{name} is flying!"
  end
end
```



```
jonathan = Bird.new('Livingstone')
```

```
jonathan.extend Flier
```

```
jonathan.fly  
=> "Livingstone is flying!"
```

```
module Bird
  attr_accessor :name

  def initialize(name)
    @name = name
  end
end
```



```
class Canary  
  include Bird  
  include Flier  
end
```

```
tweety = Canary.new("Tweety")  
tweety.fly
```

=> Tweety is flying!

```
module Swimmer  
  def swim  
    puts "#{name} is swimming!"  
  end  
end
```

```
class Penguin  
  include Bird  
  include Swimmer  
end
```

```
kowalski = Penguin.new("Kowalski")  
kowalski.swimming
```

=> Kowalski is swimming!

Sorularınız?

Teşekkürler ...

A black and white portrait of a man with long, dark, wavy hair, looking slightly to his right. He is wearing a dark, patterned jacket over a dark shirt. The background is a plain, light-colored wall.

Andaç Karay Kudu

www: [akkudu.com](http://akkudu.com)

twitter: [@akkuducom](https://twitter.com/@akkuducom)

email: [hey@akkudu.com](mailto:hey@akkudu.com)