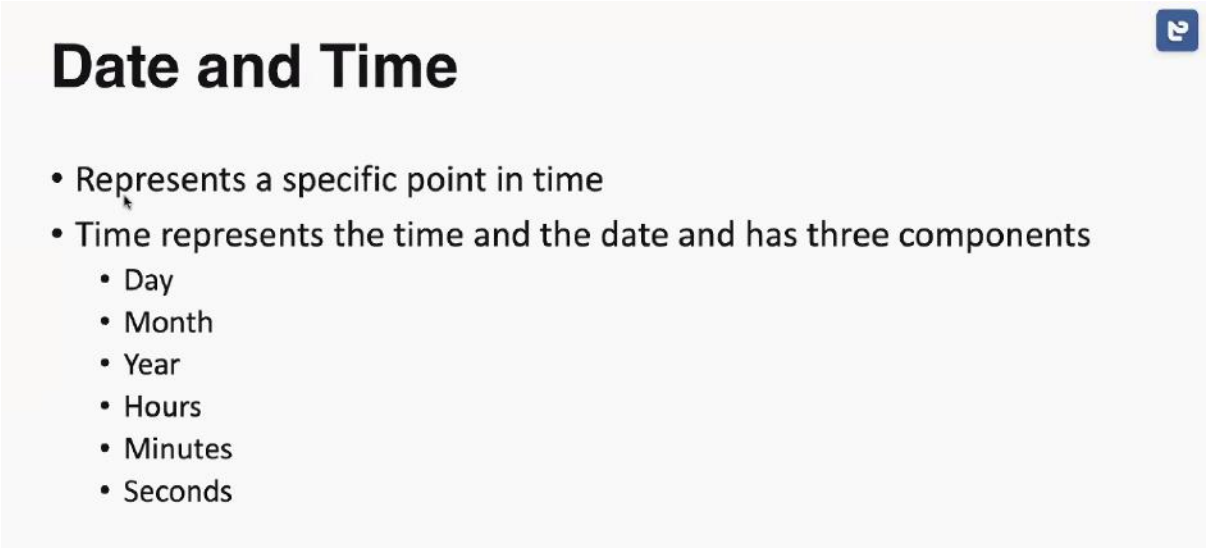
Date and time

They are ruby class
These classes have a lot of methods and utilities



- Time.now
- Time.at(linuxtimestamp)
- Others

```ruby
Time.now
# 2018-10-19 15:43:20 +0200

Time.new(2018, 1, 1)
# 2018-01-01 00:00:00 +0100

Time.at(15000000000)
# 2017-07-14 04:40:00 +0200
```

Linux timestamp
Quantity of time since 1970, it is on milliseconds

add text

```ruby
t = Time.now

puts t.day
puts t.month
puts t.hour
```

Time.local -> declare time on the current time zone
You can also pass the time-zone as a parameter

Date and Time

- Time also has an equal operator <=>
- You can convert the time to an ASCII string

```
Time.now.asctime    #=> "Wed Apr  9 08:56:03 2003"
Time.now.ctime      #=> "Wed Apr  9 08:56:03 2003"
```

Date and Time – Daylight Saving Time

- Click to add text

```
Time.local(2000, 1, 1).dst?    #=> false
```

to know if we are on daylight saving time.

Date and Time - strftime

- Convert a formatted string into a Time

```
-   don't pad a numerical output
_   use spaces for padding
0   use zeros for padding
^   upcase the result string
#   change case
:   use colons for %z
```

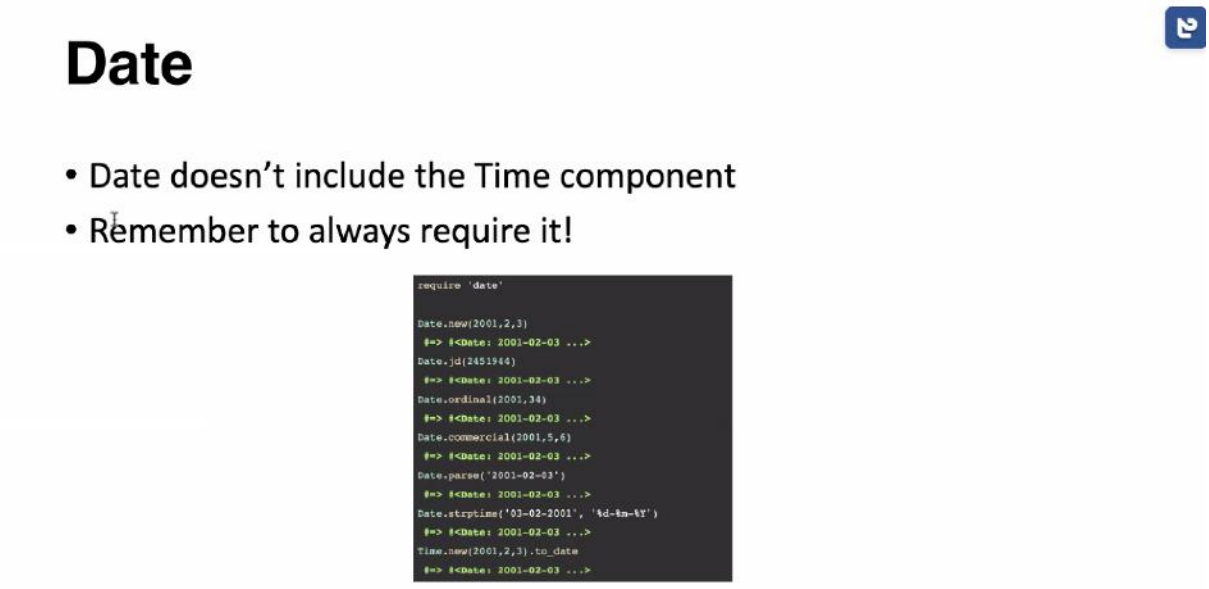
THIS IS A REALLY USEFUL METHOD

```ruby
time = Time.new

time.strftime("%d/%m/%Y")        # "05/12/2015"
time.strftime("%k:%M")           # "17:48"
time.strftime("%I:%M %p")        # "11:04 PM"
time.strftime("Today is %A")     # "Today is Sunday"
time.strftime("%d of %B, %Y")    # "21 of December, 2015"
time.strftime("Unix time is %s") # "Unix time is 1449336630"
```

Converts current time to string and it will format it

Date



It by default does not include the time part.
You need to require 'date' to use it.

Date

- All Date objects are immutable
  - They can't modify themselves
- Accessing specific values with .year, .mon, .mday, .wday

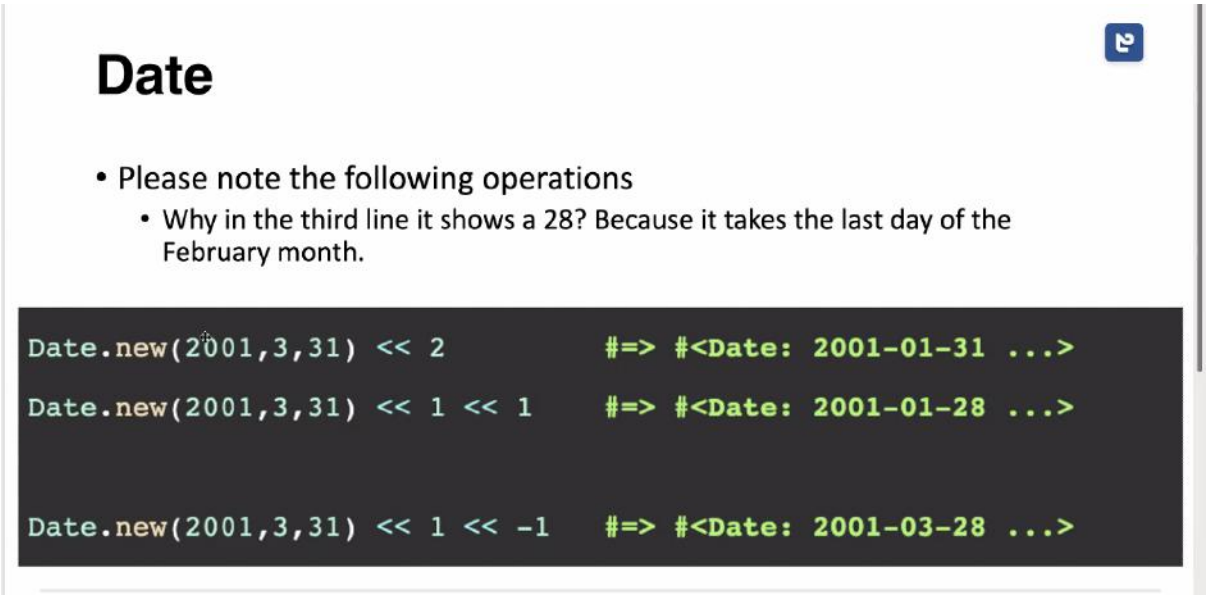
```
d = Date.parse('3rd Feb 2001')
                                #=> #<Date: 2001-02-03 ...>
d.year                          #=> 2001
d.mon                           #=> 2
d.mday                          #=> 3
d.wday                          #=> 6
d += 1                          #=> #<Date: 2001-02-04 ...>
d.strftime('%a %d %b %Y')       #=> "Sun 04 Feb 2001"
```

Date

- Available constants in the Date Module

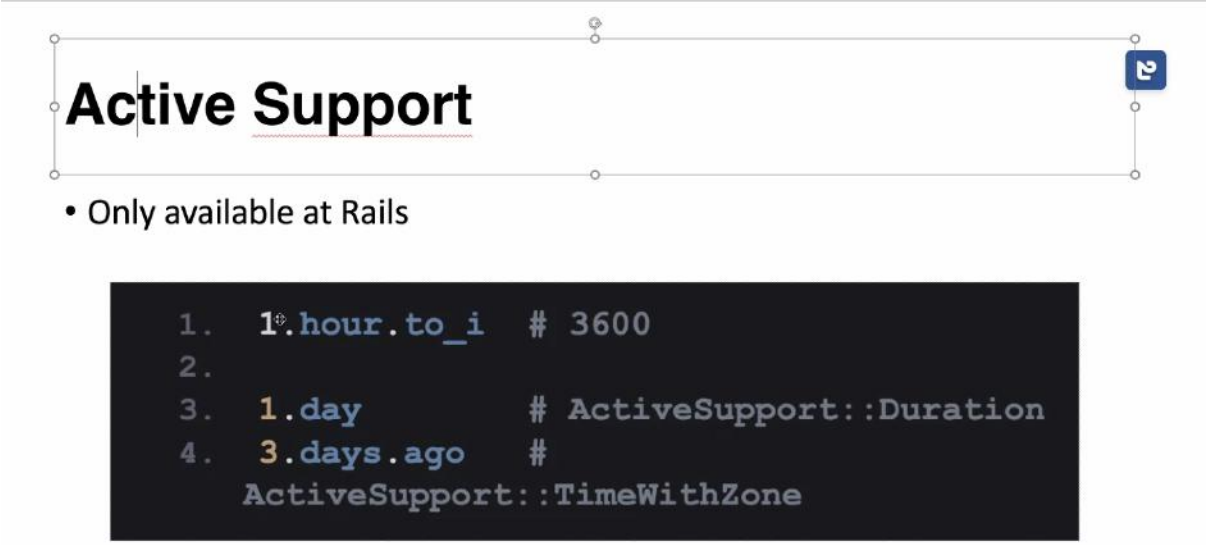
Constants I

Date::ABBR_DAYNAMES	Array of abbreviated day names
Date::ABBR_MONTHNAMES	Array of abbreviated month names
Date::DAYNAMES	Array of full names of days of the week
Date::ENGLAND	The Julian day number of the day of calendar reform for England and her colonies
Date::GREGORIAN	The Julian day number of the day of calendar reform for the proleptic Gregorian calendar
Date::ITALY	The Julian day number of the day of calendar reform for Italy
Date::JULIAN	The Julian day number of the day of calendar reform for the proleptic Julian calendar
Date::MONTHNAMES	Array of full month names

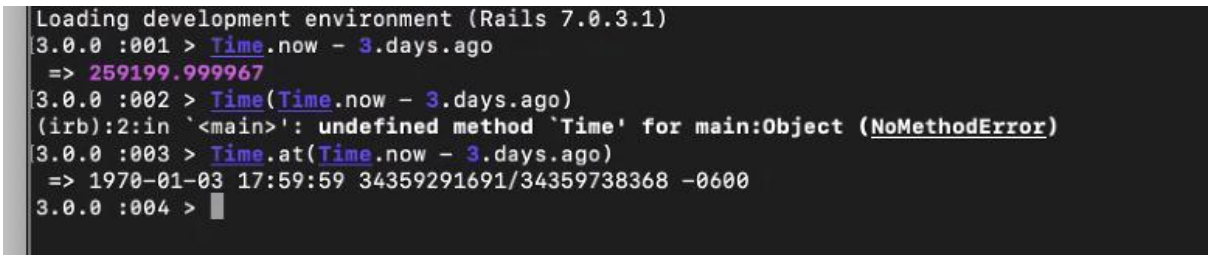


These operators reduce the months from the date.
We also go back to the last day of the month if we put a day greater than the max quantity of days that the month we are trying to get has.

Active support



With active support we can do these type of things



Homework
We are gonna use three active support methods related to date and time. Represent that time formatted.