



# Exam Next Friday (08/10/2021)

Revise the last this TD correct and the previous one  
Revise Ruby Courses

# Setup

1. launch the VM
2. launch Sublime Editor
3. launch a terminal
4. Use IRB to test some snippets of code

```
→ hugues$ irb
irb(main):001:0> puts "Hello, world!"
Hello, world!
=> nil
irb(main):002:0> exit
→ hugues $
```

# Write the class in a file

```
# user.rb
class User
  # class constants
  ME_NAME = 'Hugues'
  ME_LASTNAME = 'Bernet-Rollande'

  # instance variables
  attr_accessor :firstname, :lastname

  # initialize
  def initialize(firstname:, lastname:)
    self.firstname = firstname
    self.lastname = lastname
  end

  # instance method
  def name
    "#{firstname} #{lastname}"
  end

  # class method
  def self.me
    new(firstname: ME_NAME, lastname: ME_LASTNAME)
  end
end
```

# Use it in IRB

```
→ hugues$ irb
irb(main):001:0> load 'user.rb'
=> true
irb(main):002:0> User.new(firstname: "Steve", lastname: "Jobs").name
→ "Steve Jobs"
irb(main):002:0> User.me.name
→ "Steve Jobs"
```

**!!** You can redo `load 'user.rb'`  
to reload the file

# HTTParty

## Terminal

```
gem install httparty  
irb
```

## IRB

```
require 'httparty'  
result = HTTParty.get(  
  'https://slack.com/api/conversations.list?limit=50',  
  headers: {  
    Authorization: "Bearer xoxp-2486113197334-2492860403907-2492926538098-76ac2d6b0dcc5d6a24b3c72889355468"  
  }  
)  
channels = result.parsed_response['channels']
```

Given the following code blocks

# Fetch a list of channels

```
require 'httparty'
authorization = "Bearer xoxp-2486113197334-2492860403907-2492926538098-76ac2d6b0dcc5d6a24b3c72889355468"
response = HTTParty.get(
  'https://slack.com/api/conversations.list?limit=50',
  headers: {
    Authorization: authorization
  }
)
channels = response.parsed_response['channels']
for channel in channels
  puts "#{channel['id']}: #{channel['name']}"
end
# => C02E24403SB: general
# ...
```

# Fetch a list of messages

```
# require 'httparty' # no need to re-include if already loaded
# authorization = ... # no need to redefine if already defined
channel = 'C02E24403SB'
response = HTTParty.get(
  "https://slack.com/api/conversations.history?limit=50&channel=#{channel}",
  headers: {
    Authorization: authorization
  }
)
messages = response.parsed_response['messages']
for message in messages
  puts "#{message['ts']}: #{message['text']}"
end
# => 1632912974.000500: Hello
# ...
```



# Post a message

```
# require 'httparty' # no need to re-include if already loaded
# authorization = ... # no need to redefine if already defined
# channel = ... # no need to redefine if already defined
text = "Hello World"
response = HTTParty.post(
  "https://slack.com/api/chat.postMessage?channel=#{channel}&text=#{CGI.escape(text)}",
  headers: { Authorization: authorization }
)
message = response.parsed_response['message']
puts "#{message['ts']}: #{message['text']}"
# => 1632912974.000500: Hello World
```

# Update a message

```
ts = '1632920336.000100'
text = "Hello World, again!"
response = HTTParty.put(
    "https://slack.com/api/chat.update?channel=#{channel}&text=#{CGI.escape(text)}&ts=#{ts}",
    headers: { Authorization: authorization }
)
message = response.parsed_response['message']
puts "#{message['ts']}: #{message['text']}"
# => 1632920410.000300: Hello World
```

# Delete a message

```
ts = '1631695747.001300'
response = HTTParty.delete(
    "https://slack.com/api/chat.delete?channel=#{channel}&ts=#{ts}",
    headers: { Authorization: authorization }
)
success = response.parsed_response['ok']
puts success
# => true
```

## A. Write a slack client Class in a file `slack_client.rb` which allow to access the list of conversation

```
load 'slack_client.rb' # load the class written in the slack_client.rb
authorization = "Bearer xoxp-2486113197334-2492860403907-2492926538098-76ac2d6b0dcc5d6a24b3c72889355468"
client = Slack::Client.new(authorization: authorization)
conversations = client.get_conversations
# puts conversations.inspect # optional to "inspect" the content of conversations
conversations.each do |conversation|
  puts "#{conversation['id']}: #{conversation['name']}"
end
# => C02E24403SB: general
# ...
```

B. Add a default client initialize with the correct authorization.

```
load 'slack_client.rb'
client = Slack::Client.default_client
conversations = client.get_conversations
conversations.each do |conversation|
  puts "#{conversation['id']}: #{conversation['name']}"
end
# => C02E24403SB: general
# ...
```

## C. Add messages fetching to your Slack Client

```
load 'slack_client.rb'
client = Slack::Client.default_client
conversations = client.get_conversations
conversation_id = conversations.first['id']
messages = client.get_messages(
  conversation_id: conversation_id
)
# puts messages.inspect
messages.each do |message|
  puts "#{message['ts']}: #{message['text']}"
end
# => 1632912974.000500: Hello
# => 1632912975.000500: World
```

## D. Add message posting to your slack client

```
load 'slack_client.rb'
client = Slack::Client.default_client
conversations = client.get_conversations
conversation_id = conversations.first['id']
message = client.post_message(
  conversation_id: conversation_id,
  text: 'Hello'
)
puts "#{message['ts']}: #{message['text']}"
# => 1632912974.000500: Hello
```

## E. Add message updating to your slack client

```
load 'slack_client.rb'
client = Slack::Client.default_client
conversations = client.get_conversations
conversation_id = conversations.first['id']
message = client.post_message(
  conversation_id: conversation_id,
  text: 'Hello'
)
puts message.inspect
puts "#{message['ts']}: #{message['text']}"
# => 1632912974.000500: Hello
message = client.update_message(
  conversation_id: conversation_id,
  message_id: message['ts'],
  text: 'World'
)
puts message.inspect
puts "#{message['ts']}: #{message['text']}"
# => 1632912974.000500: World
```



## F. Add message deleting to your slack client

```
load 'slack_client.rb'
client = Slack::Client.default_client
conversations = client.get_conversations
conversation_id = conversations.first['id']
message = client.post_message(
  conversation_id: conversation_id,
  text: 'Hello'
)
puts message.inspect
puts "#{message['ts']}: #{message['text']}"
# => 1632912974.000500: Hello
success = client.delete_message(
  conversation_id: conversation_id,
  message_id: message['ts']
)
puts success
```

## G. Define a new class to encapsulate the conversations and messages Hashes

```
load 'slack_client.rb'
client = Slack::Client.default_client
conversations = client.get_conversations
conversations.each do |conversation|
  puts "#{conversation.id}: #{conversation.name}" # note the `.id` instead of conversation['id']
end
# => C02E24403SB: general
# ...
conversation = conversations.first
messages = conversation.get_messages
messages.each do |message|
  puts "#{message.id}: #{message.text}"
end
# => 1632912974.000500: Hello
# ...
message = conversation.create_message(text: 'Hello')
puts "#{message.id}: #{message.text}"
# => 1632912974.000500: Hello
message = message.update(text: 'World')
puts "#{message.id}: #{message.text}"
# => 1632912856.000300: World
message.delete
```