

# Module 3

## HTTP, RESTful APIs

CS W169A: Software Engineering  
John Yang | Summer 2020

### 1 Github Routes

Read through and try to answer the following questions. Work in pairs if you can!

#### 1.1 Requests

Without looking at any documentation, discuss what the following HTTP requests might do (no specific answer format required, just a general idea). \*Note\*: These links do not work, but determine what the request type and arguments would do if they did exist.

- GET <https://github.com/orgs/cs169/repos>
- GET <https://github.com/repos/cs169/homework2>
- POST <https://github.com/orgs/cs169/repos>

#### 1.2 API Documentation

Read through GitHub's API documentation, which can be found at <https://developer.github.com/v3>. Find answers to the following questions.

- How can I get publicly available information about a user?
- How can I list all pull requests of a repository?
- How can I get all closed pull requests of a repository?
- How can I create a new pull request?
- Which input values are required to create a new pull request?

### 2 API Implementation

Finish the following ruby implementation for a simple todo API. Assume you have access to a params hash with any necessary query parameters, along with a `format_as_json(object)` function. Return results as JSON when applicable.

```
require 'sinatra'

class User
  attr_reader :id
  attr_accessor :todo

  def initialize
    @id = app.get_new_user_id
    @todo = Todo.new
  end
end
```

```

end

class Todo
  def initialize
    @items = Array.new
  end

  def add item
    if not @items.include?
      item @items << item
    end
  end

  def delete item
    @items.delete item
  end

  def view_item item_id
    # assume each item instance has an accessible id field,
    # and all items/ids are unique
    item = _____
    format_as_json item
  end
end

get '/user/:id/todo' do
  user = get_user_by_id params[:id]
  _____
  _____
end

get '/user/:id/todo/:item_id' do user = get_user_by_id params[:id]
  _____
  _____
end

post '/user/:id/todo' do
  user = get_user_by_id params[:id] request.body.rewind
  raw_item = JSON.parse request.body.read
  _____
end

delete '/user/:id/todo/:item' do user = get_user_by_id params[:id]
  _____
end

```