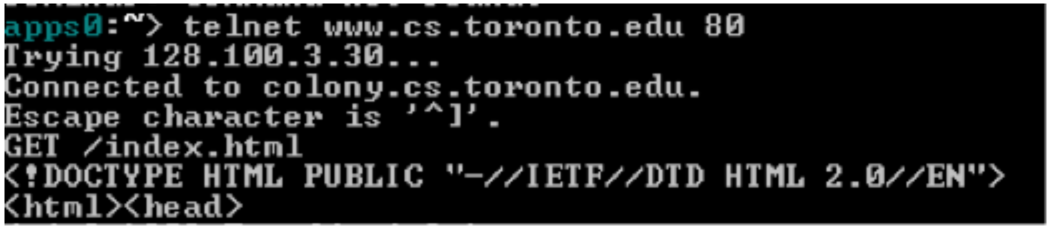
REST API

UNIVERSITY OF
TORONTO

Hypertext Transport Protocol (HTTP)

- The set of commands understood by a web server and sent from a browser
- Runs on top of TCP/IP
- Managed by the World Wide Web Consortium (W3C)
- Current version HTTP 1.1, IETF RFC 2616
- Two phase protocol
  - Request followed by response
- Simulating a browser with a terminal window:



```
apps0:~> telnet www.cs.toronto.edu 80
Trying 128.100.3.30...
Connected to colony.cs.toronto.edu.
Escape character is '^]'.
GET /index.html
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
```

Uniform Resource Locator (URL)

→ An identifier for the location of a document on a web site

Format

<scheme>:<scheme-specific-address>

<scheme> = http,file,ftp

For HTTP

// domain-name/path-to-document

www.cs.toronto.edu/~delara/courses/csc309/index.html

REST - Representational state transfer

“REST is just a set of conventions about how to use HTTP”

Instead of having randomly named setter and getter URLs and using `GET` for all the getters and `POST` for all the setters, we try to have the URLs identify resources, and then use the HTTP actions `GET`, `POST`, `PUT` and `DELETE` to do stuff to them. So instead of

```
GET /get_article?id=1

POST /delete_article id=1
```

You would do

```
GET /articles/1/

DELETE /articles/1/
```

REST API

We need two basic URLs per resource

→ One for a collection of the resource

```
/dogs
```

→ The other one is for a particular resource.

```
/dogs/1
```

REST API

GET

Read a specific resource (by an identifier) or a collection of resources.

PUT

Update a specific resource (by an identifier) or a collection of resources. Can also be used to create a specific resource if the resource identifier is know before hand.

REST API

DELETE

Remove/delete a specific resource by an identifier.

POST

Create a new resource. Also a catch-all verb for operations that don't fit into the other categories.

REST API

To show association:

```
/owner/123/dogs
```

```
Or
```

A bit more complex

```
/dogs?color=write&location=toronto
```

REST API - Idempotence

Same result over unlimited number of calls, since we are dealing with same resources.

However, *delete* may return 404 error in subsequent calls if not handled properly.

REST API

Best Practices

→ think about “resources”

→ elements & collections

→ map out the 4 methods for each

→ Prefer Nouns, Plurals, Concrete

→ Use Parameters for more advanced queries