



TDDBUDDY.COM

# URL Parts

## Rules

1. Strictly practice TDD
  - a. Follow the: Test, Implement, Reflect, Refactor cycle.
  - b. Be sure to run your test after each reflect-refactor cycle.
2. Practice wishful coding
  - a. Do not write any production code until you have a failing test.
  - b. Or compilation failures to drive the need to implement the method.
3. Use your TDD Cube and Reference card if you have, be-sure to cycle the cube with each step

## The Kata

We all know about URLs like <http://www.tddbuddy.com>

Decomposes a given URL into its parts. In the above example, we would get the result

1. The **protocol**: "http"
2. The **subdomain**: "www"
3. The **domain name**: "tddbuddy.com"
4. The **port**: 80 ( Default for HTTP )
5. The **path**: an empty string in our example.

Only handle top level domains like .com or .net. Do not attempt to handle second level domains like .co.uk or co.za  
Only handle the protocols specified in the default ports section below.

## Default Ports

http: 80, https: 443, ftp: 21, sftp: 22

## Examples

- <http://foo.bar.com/foobar.html>
  - **Protocol**: http
  - **Subdomain**: foo
  - **Domain name**: bar.com
  - **Port**: 80
  - **Path**: foobar.html
- <ftp://foo.com:9000/files>
  - **Protocol**: ftp
  - **Subdomain**: empty string
  - **Domain name**: foo.com
  - **Port**: 9000
  - **Path**: files
- <https://www.foobar.com:8080/download/install.exe>
  - **Protocol**: https
  - **Subdomain**: www
  - **Domain name**: foobar.com
  - **Port**: 8080
  - **Path**: download/installer.exe

## Hints

Exclude the leading / when handling path. E.g. /download becomes download.

## Bonus

Refactor your code to be five lines of less per method.