Publishing

The basics of web hosting

Files are real

Monica Chin, **"File not found,"** *The Verge*, September 22, 2021

Files live on computers

Websites are files

HTML

```html
<!DOCTYPE html>
<html>
<head>
  <title>Page title</title>
</head>
<body>
  <h1>A heading</h1>
  <p>Some text</p>
</body>
</html>
```

CSS

```css
h1 {
  color: "red";
  text-align: center;
  margin-bottom: 10px;
}
```

Other files

Javascript

```javascript
const listItems = document.querySelecto

function toggleDone(e) {
  if (!e.target.className) {
    e.target.className = "done";
  } else {
    e.target.className = "";
  }
}

listItems.forEach((item) => {
  item.addEventListener("click", toggle
});
```

(Some are databases + files)

By WordPress Foundation - Own work, GPL, Link

Quarto websites are *static*

There’s no need for fancy web servers—
Quarto puts the whole rendered website in `_site/`

Your job is to put that `_site/` on a server.

Servers

Web servers are computers that (1) store files and (2) show those files as websites to web browsers.

Local files

`file:///Users/andrew/Desktop/example/index.html`

Local files

Creating basic websites – Qu
File
/Users/andrew/Library/CloudStorage/Dropbox/Teaching/•%20Statistical%20Horizons/quarto-website...
Quarto websites
Home
Setup
Materials
Intro to Quarto
Creating basic websites
Advanced website features
Publishing
Customization and branding
Interactivity
Colophon
Materials > Creating basic websites
Creating basic websites
Slides
On this page
Slides
Exercises
View source
_site
Recents
Shared
Favorites
Locations
_brand.yml
_dependencies.R
_extensions
_freeze
_quarto.yml
_site
colophon.qmd
css
img
index.qmd
LICENSE.md
materials
renv
renv.lock
colophon_files
colophon.html
img
index.html
materials
robots.txt
search.json
site_libs
sitemap.xml
index.html
HTML text - 36 KB
Information
Show Less
Created
Today, 01:56
Open in Positron
More...
1 of 9 selected, 632.31 GB available

Local server

http://localhost:3333/

Local server

Stephen A Smith
@stephenasmith
TAke a look, y'all: IMG_4346.jpeg
10:15 PM · May 26, 2015 · Twitter for iPhone
86.8K Retweets
4,939 Quote Tweets
121K Likes

Local server

Remote server

Remote server

Domain names

All computers on the internet have IP addresses

http://8.8.8.8

Domain names map IP addresses to memorable, human-readable names

Domain names

Costs

Server space

Can be free!

Typically $5+/month

Domain name

Not free!

$10–20+/year

Quarto hosting options

Deployment services

Quarto has built-in support for several different hosting services for publshing documents, presentations, dashboards, and websites

https://quarto.org/docs/publishing/

Easy

- **Quarto Pub**: Free!; everything's public; no custom domains
- **Posit Connect**: $$; public and private; custom domains

Medium

- **GitHub Pages**: Free!; everything's public; custom domains
- **Netlify**: Free and paid; public and private; custom domains

Advanced

- **DIY**: Set up a server yourself and do whatever you want

What I use

- **Quarto Pub**: **example** + **example**
- **GitHub Pages**: **example** + **example** + this very course website!
- **DIY**: **example** + **example**

What we'll do together

- **Quarto Pub**
- **GitHub Pages**

Deploying with Quarto Pub

Quarto Pub

https://quartopub.com/

- **What**: Free hosting service for all types of Quarto documents, run by Posit
- **When to use**: When you want to publish something on the internet quickly and you don't really care about (1) privacy or (2) the domain name
- **Difficulty level**: The easiest!

Complete documentation

Quarto Pub

andrewheiss
andrew@andrewheiss.com

sort by: Newest

pinr resources — Website
Last updated 3 days ago

ARNOVA 2025 proposal — Document
Last updated 2 weeks ago

Escaping inline R chunks — Document
Last updated 4 weeks ago

MMs and AMCEs — Document
Last updated last month

Fuzzy `rdrobust()` stuff — Document
Last updated last month

Bayesian {marginaleffects} — Document
Last updated 2 months ago

How to publish

1. Go to **quartopub.com** and create a free account. Make sure you're logged in.
2. In the terminal, run this:

   Terminal
   ```
   quarto publish
   ```

3. Select `Quarto Pub` and press enter. Answer all the other questions.
4. Wait for the site to render and upload.
5. Done!

Other details

- To update the site later, run `quarto publish` in the terminal again.
- Manage other site settings at your **quartopub.com** dashboard.

Your turn

Publish your playground website to Quarto Pub.

1. Go to **quartopub.com** and create a free account. Make sure you're logged in.
2. In the terminal, run this:

```
Terminal
quarto publish
```

3. Select `Quarto Pub` and press enter. Answer all the other questions.
4. Wait for the site to render and upload.

07:00

Deploying with GitHub Pages

GitHub Pages

- **What**: Free hosting service for repositories hosted on GitHub
- **When to use**: When you want a (free) longer term hosting solution (with a custom domain if you want) and you're already using GitHub for version control
- **Difficulty level**: Medium

Complete documentation

Three ways; three difficulties

1. **Easy**: Render your site to `docs/` and include that folder in your git repository.
2. **Medium**: Use the `quarto publish` terminal command to render the site and push it to a `gh-pages` branch.
3. **Hard**: Use GitHub Actions to build an on-the-fly virtual computer that remotely rebuilds and publishes your site every time you commit to your repository.

We're going to do Option 1!

Some git knowledge required

General version control

Remote version control hosting + collaboration

Helpful resources:

- **GitHub's Hello World tutorial**
- **Happy Git with R**

Interacting with git

GitHub Desktop | RStudio | Positron / VS Code | Terminal

Basic workflow

1. Make changes
2. Commit changes to lock them in
3. Push commits to remote server

Advanced stuff you don't have to worry about: branches, collaboration, pull requests, etc.

`.gitignore`

Use `.gitignore` to hide things from git.

General rules

Commit things that make outputs; don't commit outputs themselves.

Don't commit user-specific things (or passwords!)

.gitignore

```
.Rproj.user
.Rhistory
.RData
.Ruserdata

data/private-data.csv
output/*.png
all_my_passwords.txt
```

How to publish

Part 1: Get your project code on GitHub

1. Go to **github.com** and create a free account.
2. Add a `.gitignore` file to your website project with this:

.gitignore

```
.Rproj.user
.Rhistory
.RData
.Ruserdata

/_site/
```

3. Initialize a git repository for your website project.
4. Commit the files and push to GitHub.

How to publish

Part 2: Tell Quarto to use GitHub Pages

5. Change the `output-dir` option in `_quarto.yml` to `docs`.

   _quarto.yml

   ```yaml
   project:
     type: website
     output-dir: docs
   ```

6. Add an empty file named `.nojekyll` to the root of your website project (this tells GitHub to not run its own static site generator).
7. Render your site.
8. Commit the newly created `docs/` folder and push to GitHub.

How to publish

Part 3: Tell GitHub Pages to serve your site

9. From your GitHub repository, go to Settings > Pages and configure the repository to publish from the `docs` directory of your `main` branch.

10. Wait for GitHub to build and deploy the site.

Other details

- To update the site later, render the site, commit to GitHub, and wait for the site to rebuild.

  Track the progress under the “Actions” section of your GitHub repository.

- Add a custom domain at Settings > Pages

Your turn, part 1

Get your project code on GitHub

1. Go to **github.com** and create a free account.
2. Add a `.gitignore` file to your website project with this:

.gitignore

```
.Rproj.user
.Rhistory
.RData
.Ruserdata

/_site/
```

3. Initialize a git repository for your website project.
4. Commit the files and push to GitHub.

10:00

Your turn, part 2

Tell Quarto to use GitHub Pages

5. Change the `output-dir` option in `_quarto.yml` to `docs`.

```
_quarto.yml
project:
  type: website
  output-dir: docs
```

6. Add an empty file named `.nojekyll` to the root of your website project (this tells GitHub to not run its own static site generator).
7. Render your site.
8. Commit the newly created `docs/` folder and push to GitHub.

10:00

Your turn, part 3

Tell GitHub Pages to serve your site

9. From your GitHub repository, go to Settings > Pages and configure the repository to publish from the `docs` directory of your `main` branch.

10. Wait for GitHub to build and deploy the site.

10:00

What’s next?

Course outline

- ✅ ~~Intro to Quarto~~
- ✅ ~~Creating basic websites~~
- ✅ ~~Advanced website features~~
- ✅ ~~Publishing~~
- Customization and branding
- Interactivity