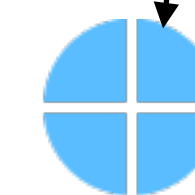
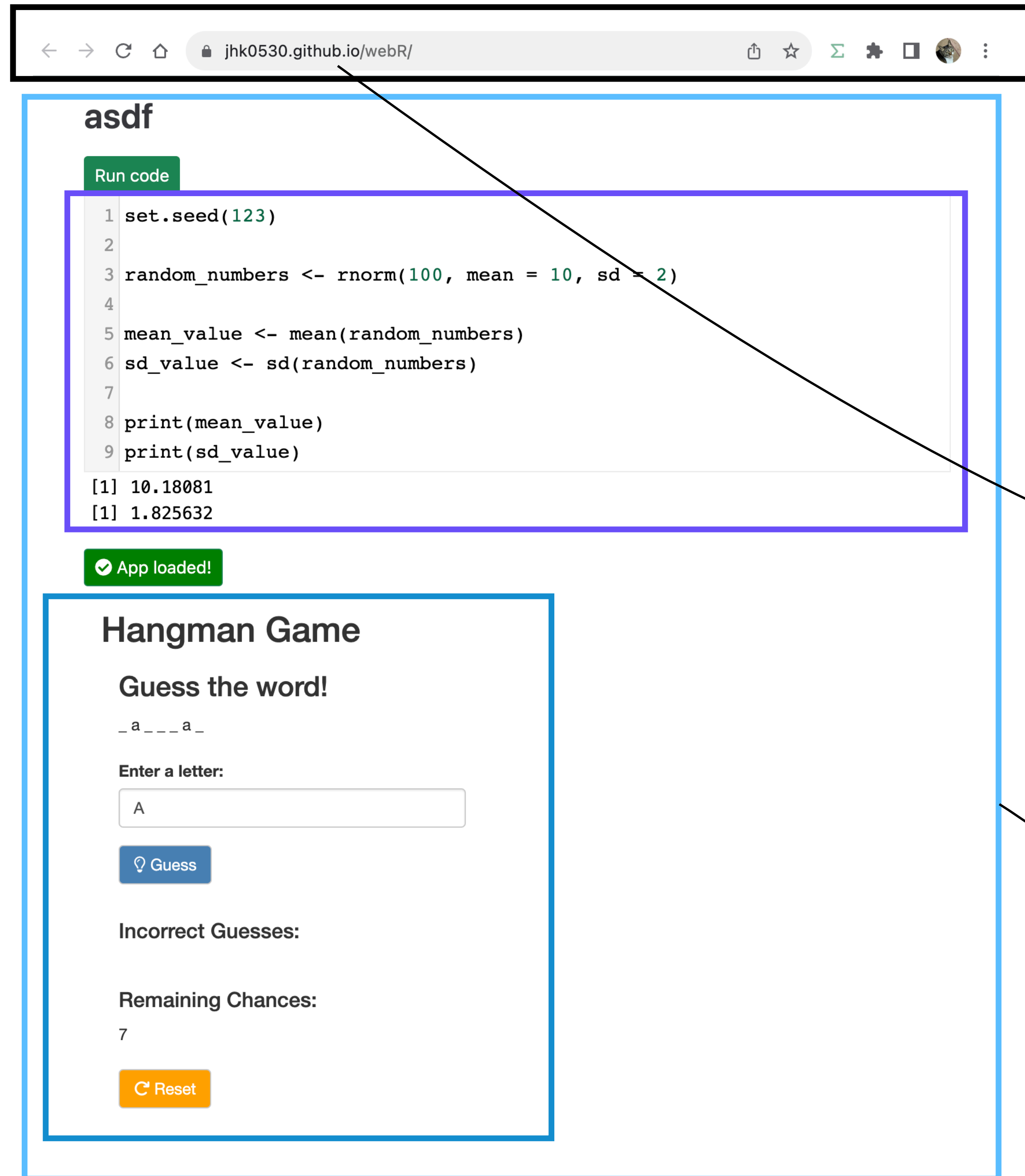


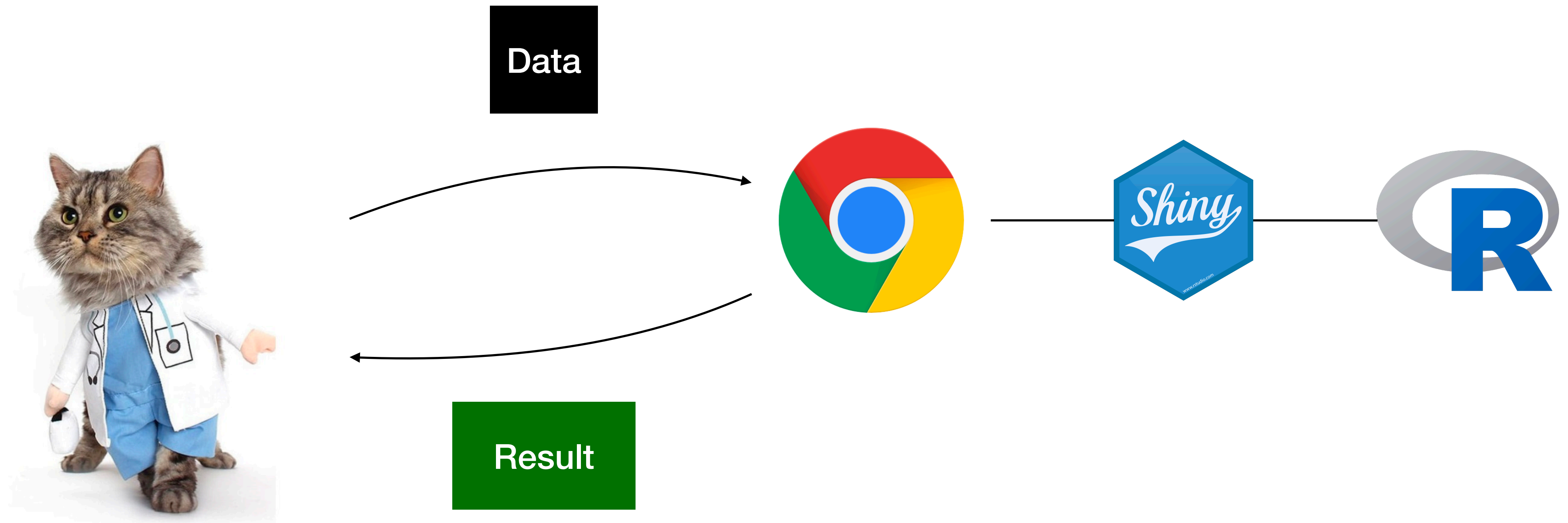
webR과 github page를 활용한 서버 리스 shiny application 구현

<https://jhk0530.github.io/webR/>



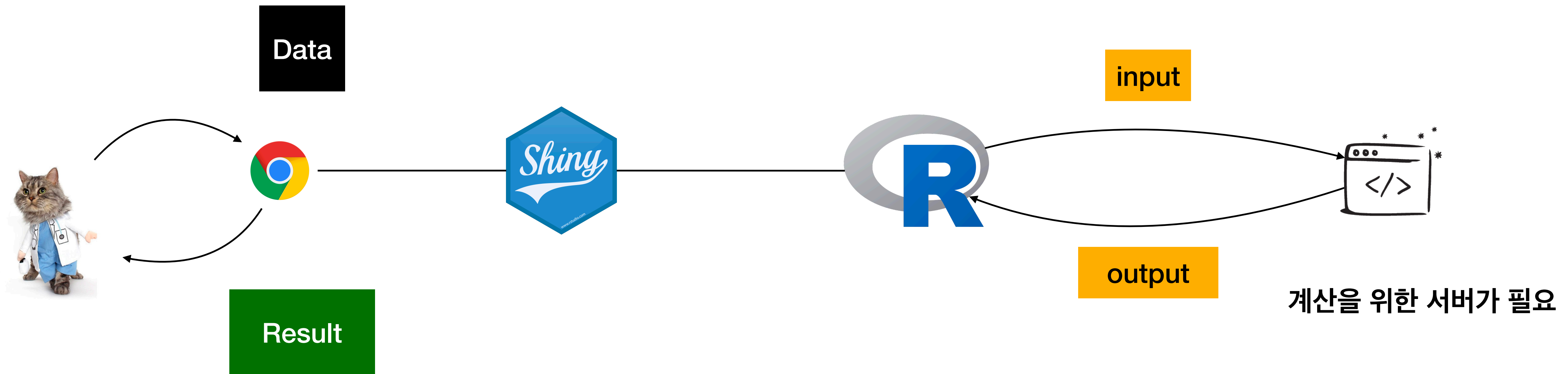
shiny

- R의 기능은 활용하고 싶지만, R을 개발하고 싶진 않은 사람을 위한 툴



behind shiny

- R의 기능은 활용하고 싶지만, R을 개발하고 싶진 않은 사람을 위한 툴



계산을 위한 서버 (개인적 크립토나이트)



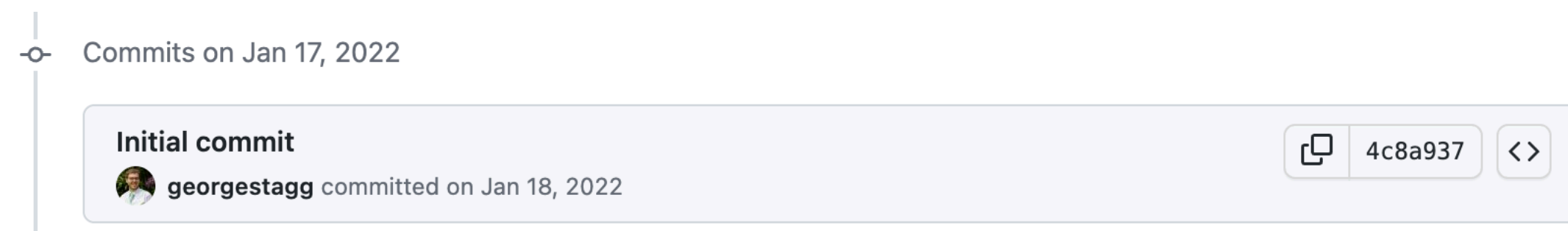
Free & limited

Paid & unlimited

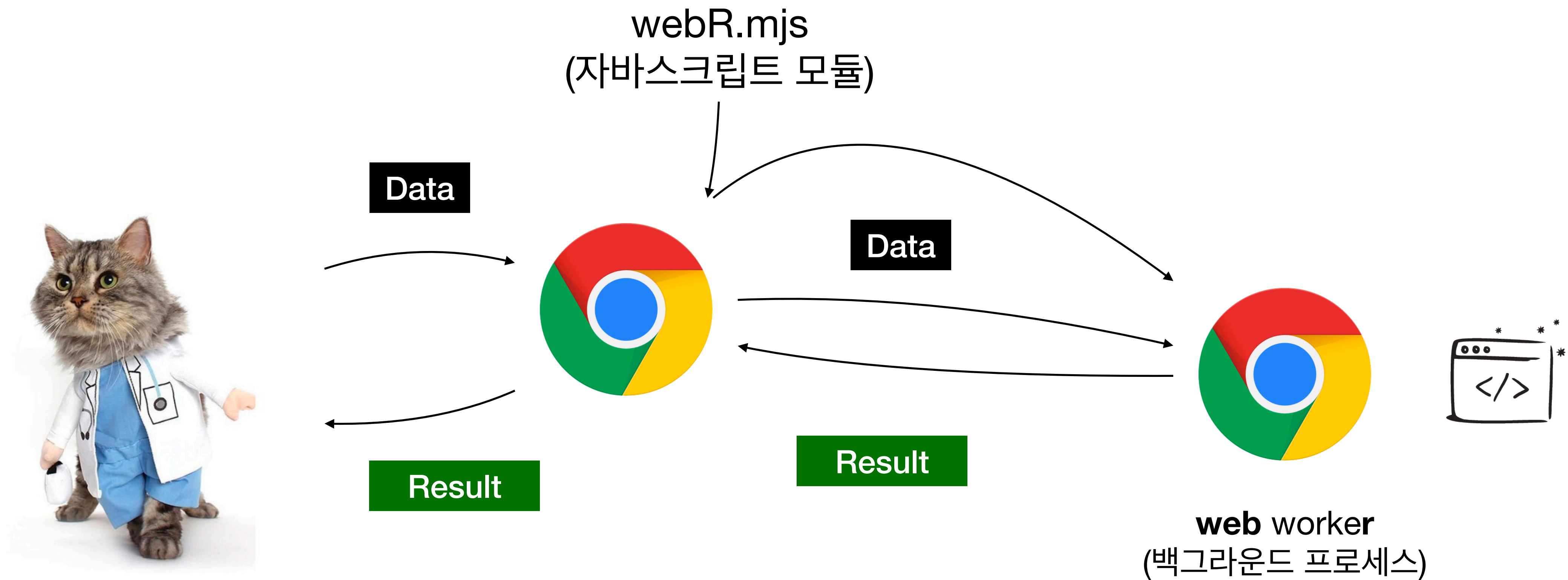
wasm

- 2017년에 첫 개념 등장.
- **Web ASseMbly**: 웹 브라우저에서 실행할 수 있는 컴파일 언어.
- shiny for python 에서는 이미 shinylive로 제공
- R은 2022년 1월에 첫 작업 시작 ~ 진행중 (webR)

<https://docs.r-wasm.org/webr/latest/>



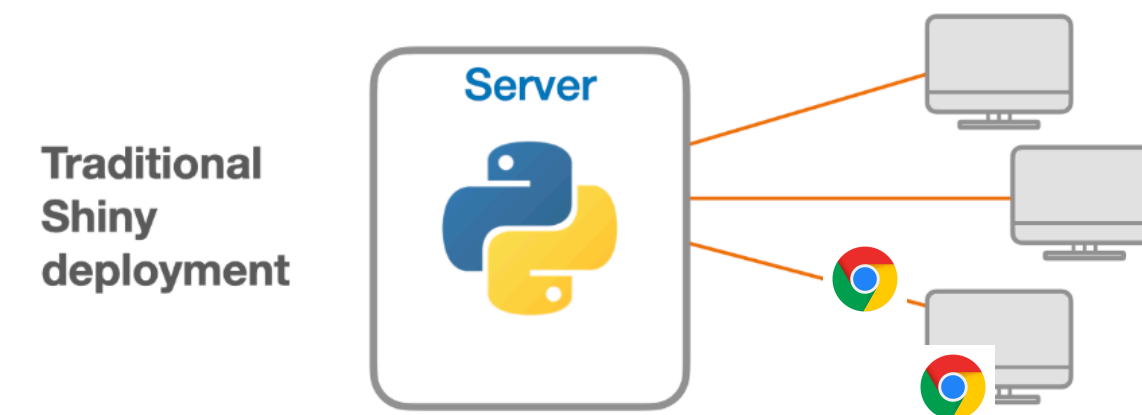
webR의 원리



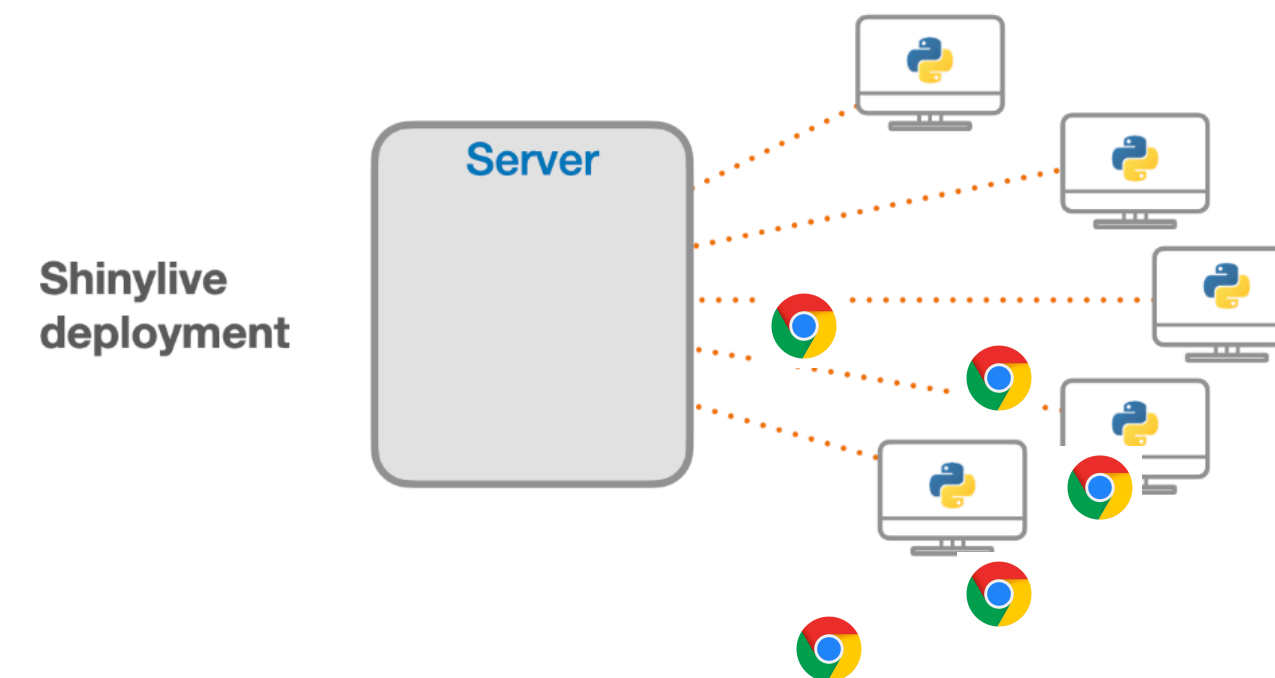
Shinylive: Shiny + WebAssembly

Shiny for Python has an experimental mode called *Shinylive*: this is where applications can run entirely in a web browser, without the need for a separate server running Python.

The traditional way of deploying Shiny involves in a separate server and client: the server runs Python and Shiny, and clients connect via the web browser. Each client keeps an open websocket connection as long as they are using the application.



When an application is deployed with Shinylive, Python and Shiny *run in the web browser*: the browser is effectively both the client and server for the application. There is a web server that serves files, but it does not run Python or Shiny—it can be a “dumb” static web server.



코드/리소스만 (정적페이지로) 제공

정적 페이지

서버에서 동적으로 생성 / 처리되는 내용 없이 고정된 내용을 제공하는 웹페이지



23.8.4

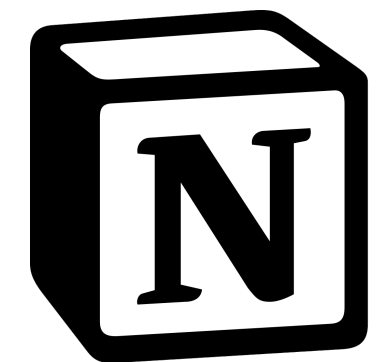


23.6.28

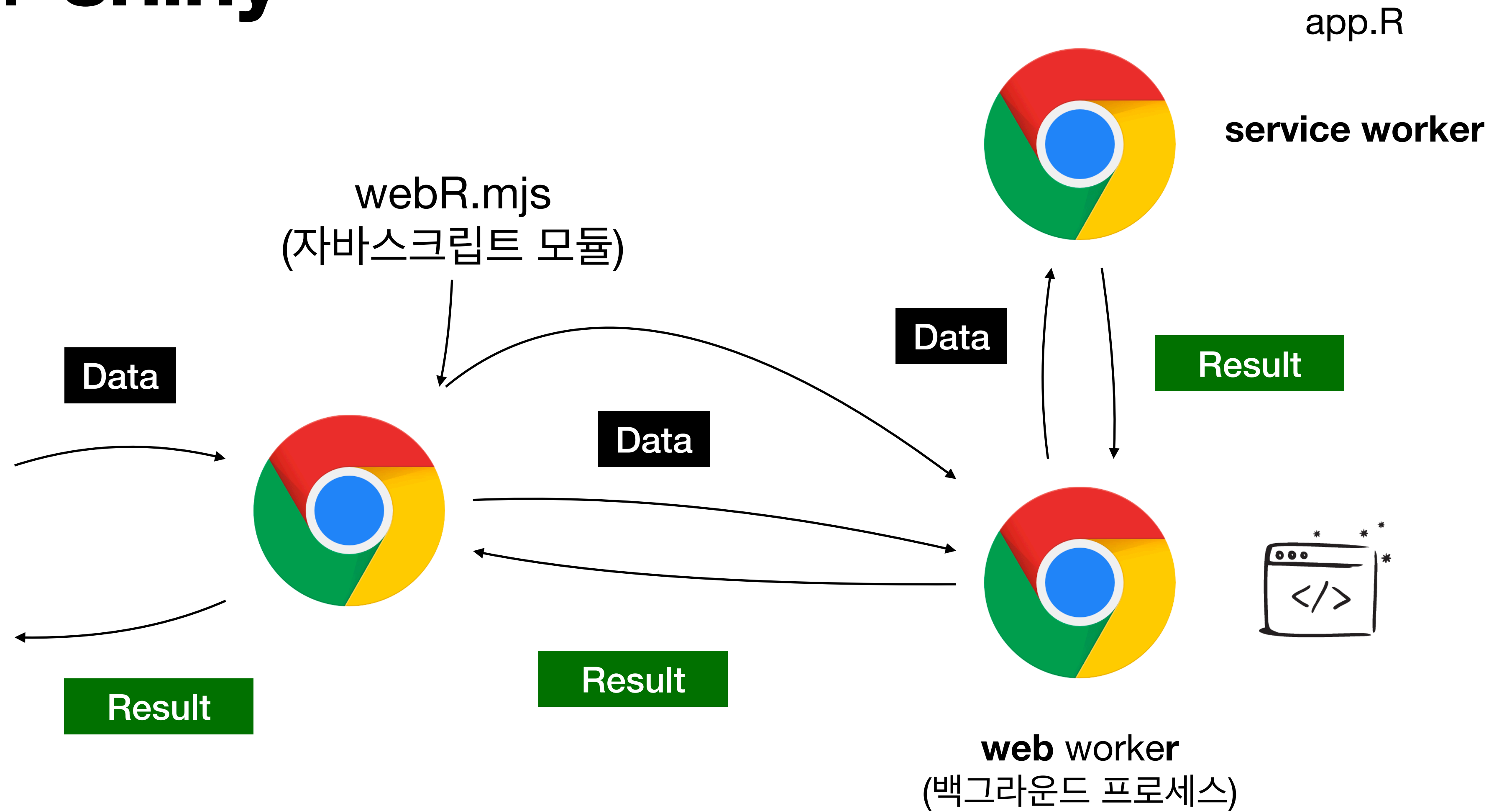


23.7.25

?



정적 페이지 + shiny



장단점

- 정적 웹사이트에 "간단한" shinyapp 추가 가능.
- 비용 / 스케일 / 퍼포먼스 걱정 X. 크롬 + 사용자 메모리 + 네트워크 연결
- 다른 웹 프레임워크 (react / vue tailwind 등) 확장 가능
- 배포 걱정 없음 (github가 대신 해줌)
 - webR은 아직 초초초초초창기 단계 (레퍼런스 / 개발자 / 사용 가능한 패키지 등이 적음)
 - 초기 설정 시간 (webR init) 필요
 - why shiny? (webR + web framework)
 - **알수 없는 이유로 (webR ?/ client ?) 가끔 로딩이 엄청 오래 걸림**

실제 구현

- 필요한 것은 3 (+ 1)가지
 - html (status button + iframe) -> Quarto로 빌드
 - shiny code (app.R) -> 있는 것 사용
 - service worker.js -> 있는 것 사용
- github page에서 service worker를 돌릴 수 있게 하는 설정 (**HARD**)

quarto 만들기

index.qmd -> index.html

yaml 설정

````{=html}

내부 코드

```

<_quarto.yml>

project:

type: website

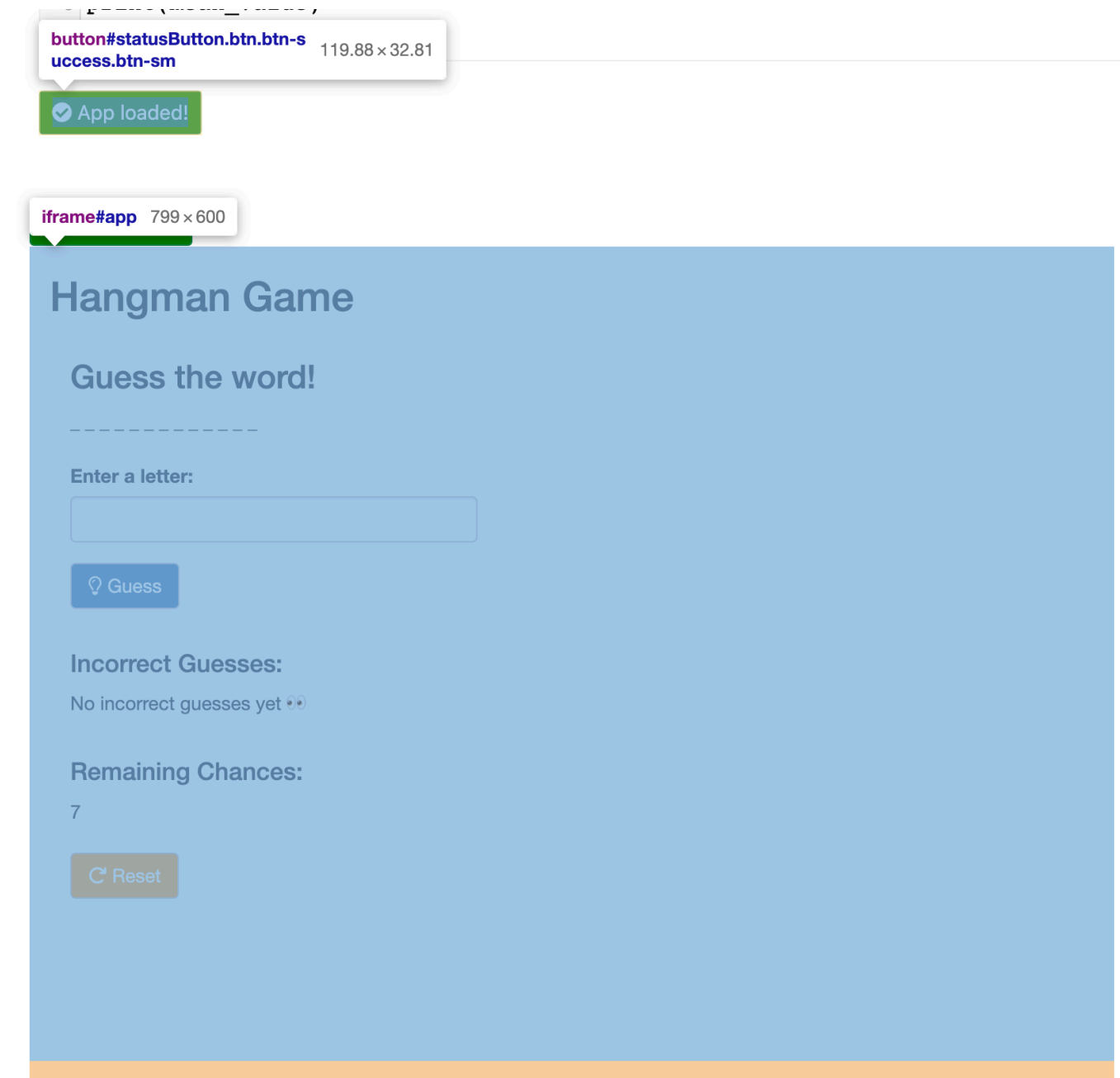
output-dir: docs

... 안됨 (실패함)

무조건 /root에 배포

html

- status를 보여주는 button (not Essential)
- shiny app을 심어주는 iframe (Essential)
- service worker를 실행시키는 javascript (Essential)



button + iframe

```
<button class="btn btn-success btn-sm" type="button" style="background-color: dodgerblue" id="statusButton">
```

```
  <i class="fas fa-spinner fa-spin"></i> Loading webR...
```

```
</button>
```

```
<div id="iframeContainer"></div>
```


service worker script

```
<script type="module">
```

```
...
```

```
loadShiny();
```

```
</script>
```

이후 step

- render index.qmd -> index.html (index 아니어도 상관없음)
 - github page 설정
 - 끝?
-
- github page에서 service worker를 돌릴 수 있게 하는 설정 (**HARD**)

CORB (COOP and COEP)

도메인 외부 파일을 함부로 읽지 못하게 하는 보안 정책

| | |
|---|--|
| ⚠ Error with Permissions-Policy header: Origin trial controlled feature not enabled: 'interest-cohort'. | |
| ⚠ 교차 출처 읽기 차단(CORB)으로 MIME 유형이 text/plain인 교차 출처 응답(https://raw.githubusercontent.com/georgestagg/shiny-standalone-webr-demo/main/httpuv-serviceworker.js)이 차단되었습니다. 자세한 내용은 https://www.chromestatus.com/feature/5629709824032768 을 참고하세요. | |
| ▶ [Violation] Added non-passive event listener to a scroll-blocking 'touchstart' event. Consider marking event handler as 'passive' to make the page more responsive. See https://www.chromestatus.com/feature/5745543795965952 | codemirror.min.js:1 |
| ▶ [Violation] Added non-passive event listener to a scroll-blocking 'touchmove' event. Consider marking event handler as 'passive' to make the page more responsive. See https://www.chromestatus.com/feature/5745543795965952 | codemirror.min.js:1 |
| ▶ [Violation] Added non-passive event listener to a scroll-blocking 'mousewheel' event. Consider marking event handler as 'passive' to make the page more responsive. See https://www.chromestatus.com/feature/5745543795965952 | codemirror.min.js:1 |
| ⚠ DevTools failed to load source map: Could not load content for https://jkh0530.github.io/webR/index2_files/libs/bootstrap/397ef2e...css.map : HTTP 오류: 상태 코드 404, net::ERR_HTTP_RESPONSE_CODE_FAILURE | |
| ⚠ DevTools failed to load source map: Could not load content for https://jkh0530.github.io/webR/index2_files/libs/quarto-html/debc5d5...css.map : HTTP 오류: 상태 코드 404, net::ERR_HTTP_RESPONSE_CODE_FAILURE | |
| webR service worker installed | webr-serviceworker.js:91 |
| webR service worker activating | webr-serviceworker.js:95 |
| > | |

✅ "enable-threads.js"를 html 페이지에 포함시키는 것으로 해결

❌ quarto 헤더에 <meta>로 설정하는 것은 도움이 되지 않음. (chatGPT)

```
<meta http-equiv="Cross-Origin-Opener-Policy" content="same-origin">
<meta http-equiv="Cross-Origin-Embedder-Policy" content="require-corp">
```

<https://github.com/josephrocca/clip-image-sorter/blob/main/enable-threads.js>

CORB with quarto

```
---  
title: "asdf"  
include-in-header:  
  text: |  
    <script type="application/javascript" src = 'enable-threads.js'></script>
```

로컬 파일이 아닌 외부 url로 하면 오류 남

정작 app.R은 외부 url로 다운로드 해서 작동 가능 (?)

httpuv-serviceworker.js

- 반드시 있어야 함. 없으면 webR은 되는데 serviceworker랑 커뮤니케이션이 불가

✖ Something went wrong...

```
index2.html:189
✖ A bad HTTP response code (404) was received when
  fetching the script.
✖ ▶ Service worker registration error: index2.html:214
  TypeError: Failed to register a ServiceWorker for scope
  ('https://jhk0530.github.io/webR/') with script ('http
  s://jhk0530.github.io/webR/httpuv-serviceworker.js'): A
  bad HTTP response code (404) was received when fetching
  the script.
```

webR = github repo 이름

- + registration 수정 필요

```
// Register service worker: note that this script needs to be uploaded to your site
const registration = await navigator.serviceWorker.register('/webR/httpuv-serviceworker.js', { scope: '/webR/' })
  .catch((error) => {
    console.error('Service worker registration error:', error);
  });
```

- 소스 코드 제외하고, quarto 에서는 수정 X

localhost

- 보통 quarto 렌더한 결과를 확인할때 localhost에서 확인.
- 이거는 localhost에서 아무런 결과가 나오지 않는 것이 정상 (!)

Hi Kim,

Thanks for your message and awesome to hear that you you're getting started with it! 😊

Are you sure you followed the steps in the first article as well? While that was aimed at WordPress, you do need to make sure you have correct policies (COOP and COEP) in place to run all of it. You can read about that here: <https://hypebright.nl/index.php/2023/06/18/embedding-r-in-wordpress-using-webr/> . Alternatively, since you're running it on an undeployed thing (probably you just run it on localhost), you might not get any of it working. You might try to deploy it first and see if it works when deployed or if it outputs anything different there.

Hope that helps! If not, let me know what else you see on the console (any JS errors or anything).

Good luck!

Veerle van Leemput
Director of Hypebright B.V.

app.R

- /app.R로는 접근 불가능.
- app.R의 raw url 주소를 통해 접근 및 다운로드
- private repository여도 접근 가능
- url = "https://raw.githubusercontent.com/jhk0530/webR/main/app.R?token=GHS~~"

최종결과물

- repo (webR)
 - index.qmd
 - index.html
 - /index_files
 - enable-thread.js
 - httpuv-serviceworker.js
 - readme.md
 - ...

[https://github.com/josephrocca/
clip-image-sorter/blob/main/enable-threads.js](https://github.com/josephrocca/clip-image-sorter/blob/main/enable-threads.js)

[https://github.com/georgestagg/
shiny-standalone-webr-demo/blob/main/httpuv-serviceworker.js](https://github.com/georgestagg/shiny-standalone-webr-demo/blob/main/httpuv-serviceworker.js)

3단계로 만들어보기

- 준비물: app.R

만들어보기


- repo 생성
 - gitignore
 - add readme
- 이후 clone

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Required fields are marked with an asterisk (*).

Owner *

 jhk0530 ▾

Repository name *

wasnR

✔ wasnR is available.

Great repository names are short and memorable. Need inspiration? How about [effective-tribble](#) ?

Description (optional)

☐

Public

Anyone on the internet can see this repository. You choose who can commit.

☒

Private

You choose who can see and commit to this repository.

Initialize this repository with:

☒ Add a README file

This is where you can write a long description for your project. [Learn more about READMEs](#).

Add .gitignore

.gitignore template: R ▾

Choose which files not to track from a list of templates. [Learn more about ignoring files](#).

Choose a license

License: None ▾

A license tells others what they can and can't do with your code. [Learn more about licenses](#).

This will set `main` as the default branch. Change the default name in your [settings](#).

 You are creating a private repository in your personal account.

Create repository

만들어보기 2

- 디렉토리로 이동 후 index.qmd 생성 (파일명은 상관없지만)
- 이후 이 내용 qmd에 입력 및 render (<https://github.com/jhk0530/wasmR/blob/main/index.qmd>)
 - 수정 1: app.R (https://raw.githubusercontent.com/rstudio/shiny/main/inst/examples/01_hello/app.R)
 - 수정 2: repo Name (wasmR)

[https://github.com/josephrocca/
clip-image-sorter/blob/main/enable-threads.js](https://github.com/josephrocca/clip-image-sorter/blob/main/enable-threads.js)

- 2개 js 파일 다운로드

[https://github.com/georgestagg/
shiny-standalone-webr-demo/blob/main/httpuv-serviceworker.js](https://github.com/georgestagg/shiny-standalone-webr-demo/blob/main/httpuv-serviceworker.js)

만들어보기 3

- 커밋
- github page 설정
 - main /(root)
- 기다리기
- 끝

다른 방법

- golem 을 써서 webR + shiny 가 가능하다고는 함 (페이지 로딩에 천년 만년 걸림)

<https://github.com/DivadNojnarg/golemWebR>

<https://github.com/RinteRface/webR4Shiny>

<https://golem-webr.rinterface.com/>

- quarto에서 webR 을 바로 쓸 수 있게 하는 template가 있음 ! 훨씬 더 편함 (shiny는 안 됨)

<https://github.com/coatless/quarto-webr>

개선이 가능할 지도?

될지 안될지 모름

- module script 별도 js로 빼서 작업 -> quarto에는 iframe만 !
- app.R 을 repo에서 바로 불러서 쓸 수 있게 작업 -> 수정 1이 없어짐
- /docs로 빼서 작업 -> quarto blog에도 shiny 가능
- 간단한 shiny의 기준은 어디까지인가. -> shinyapps.io를 대체 가능
 - 파일 업로드 / 다운로드 안됨

Q&A

- 감사합니다