

Reproducible Research with RStudio and Docker

Amy Kendig
R-Ladies Gainesville
November 12, 2019

<https://github.com/aekendig/docker-rstudio-tutorial>



Special thanks to:



Online tutorials:

- https://github.com/sanjanasudarshan/container_camp_workshop_2019/blob/master/docker/dockerintro.rst
- <http://www.derekmpowell.com/posts/2018/02/docker-tutorial-2/>
- <https://github.com/BillMills/Rocker-tutorial>

UF Flory Lab

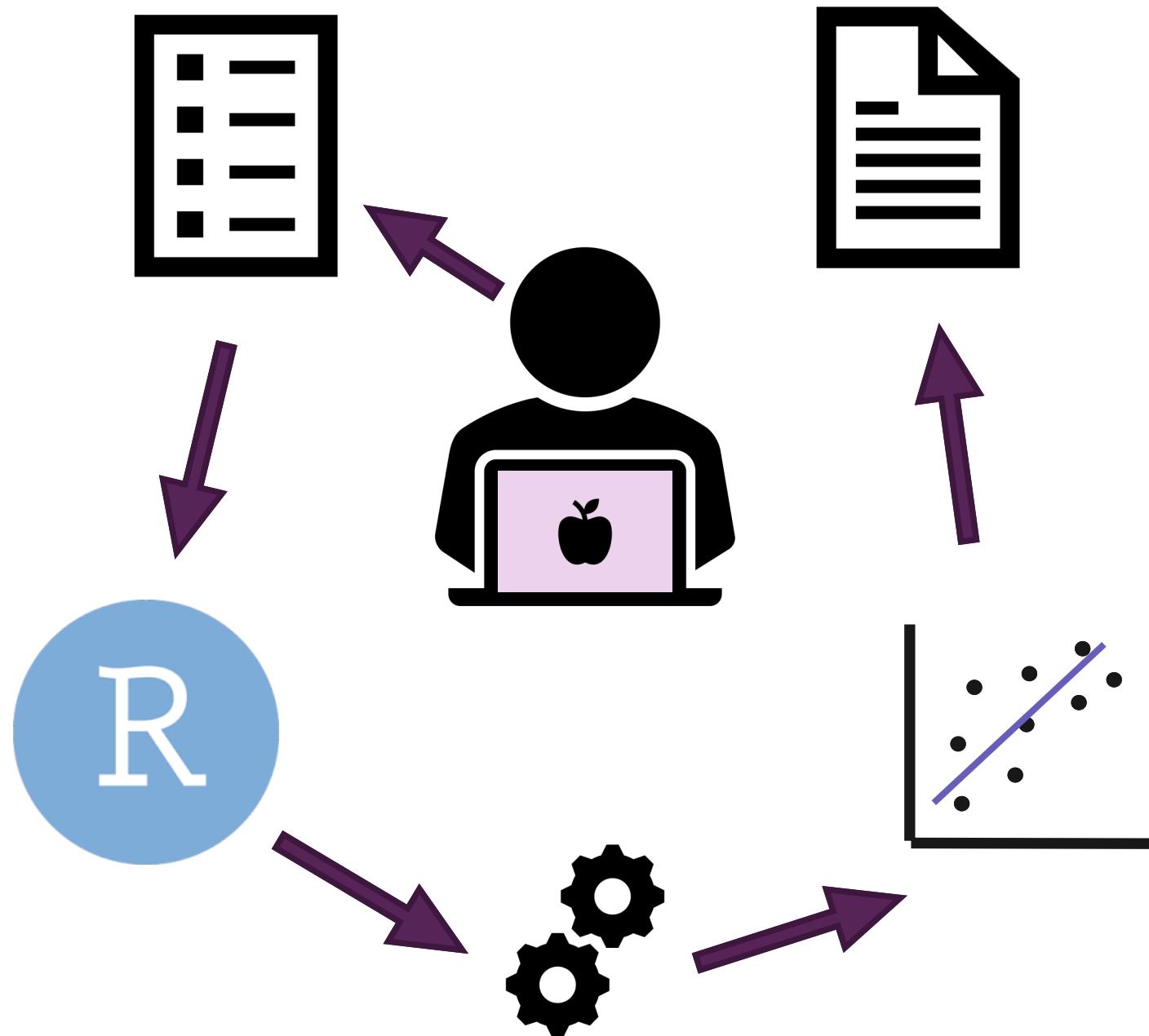
IS THERE A REPRODUCIBILITY CRISIS?

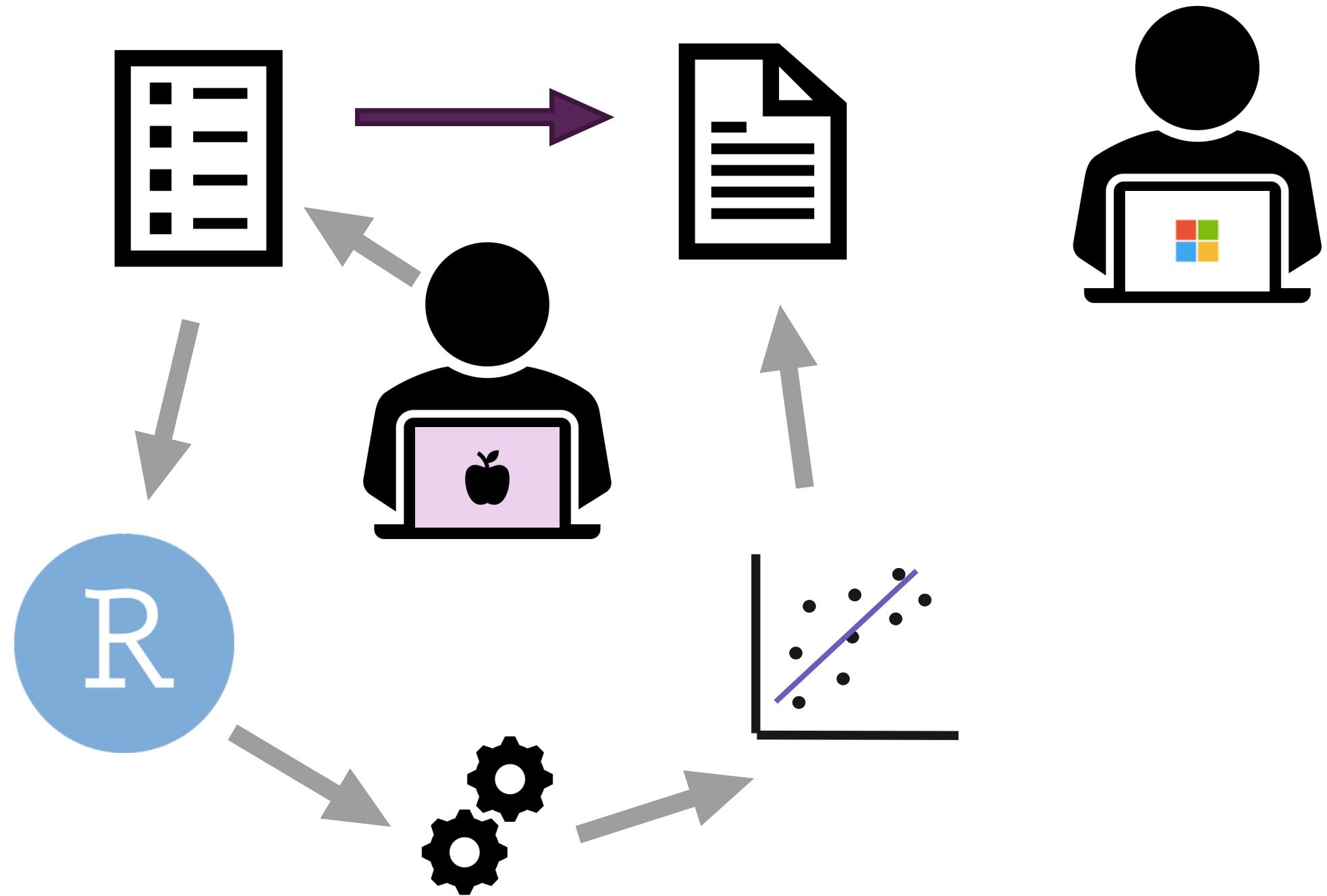


WHAT FACTORS COULD BOOST REPRODUCIBILITY?

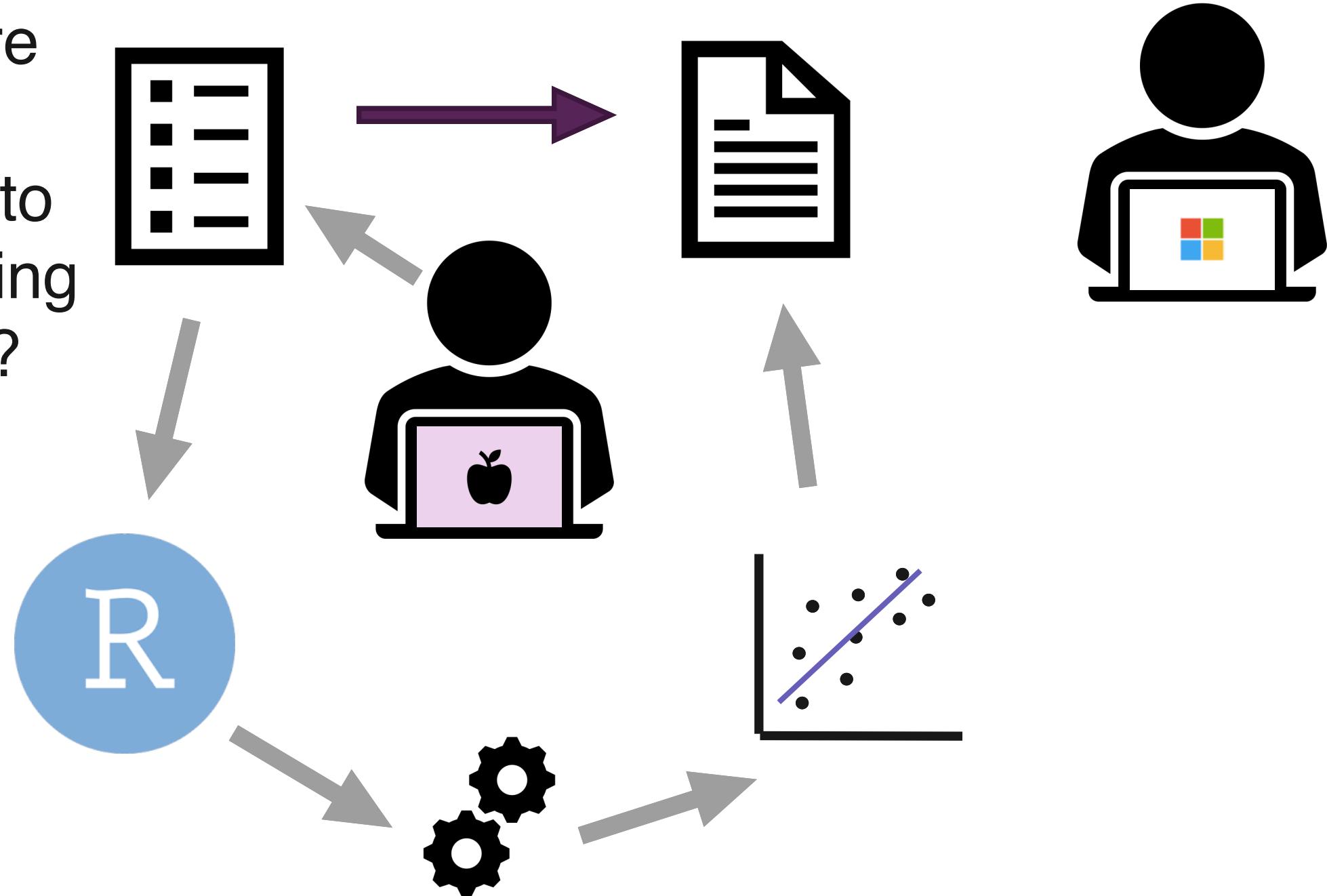
Respondents were positive about most proposed improvements but emphasized training in particular.

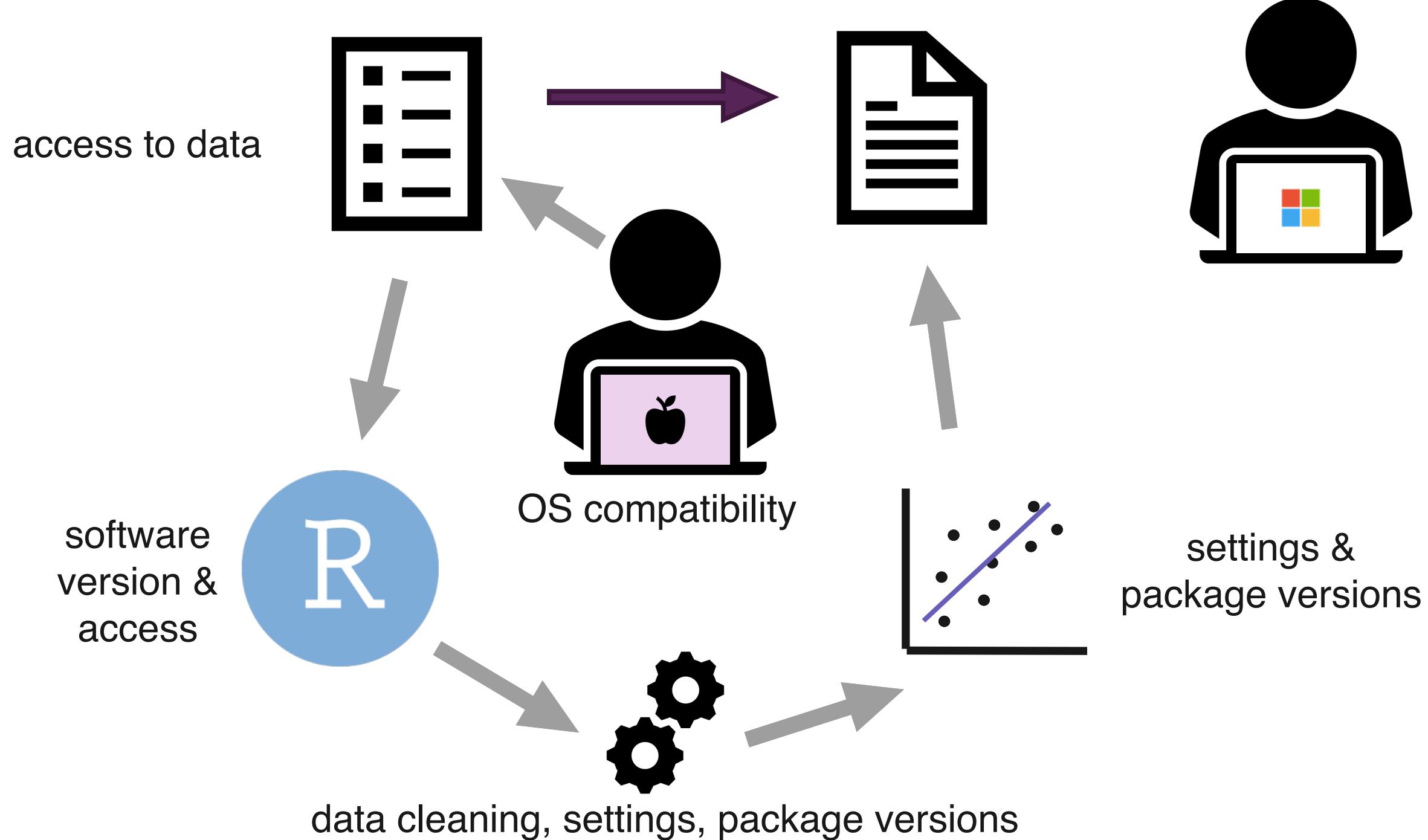






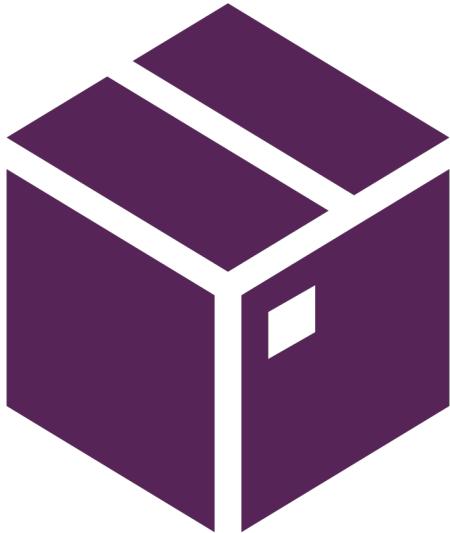
What are
some
barriers to
reproducing
results?



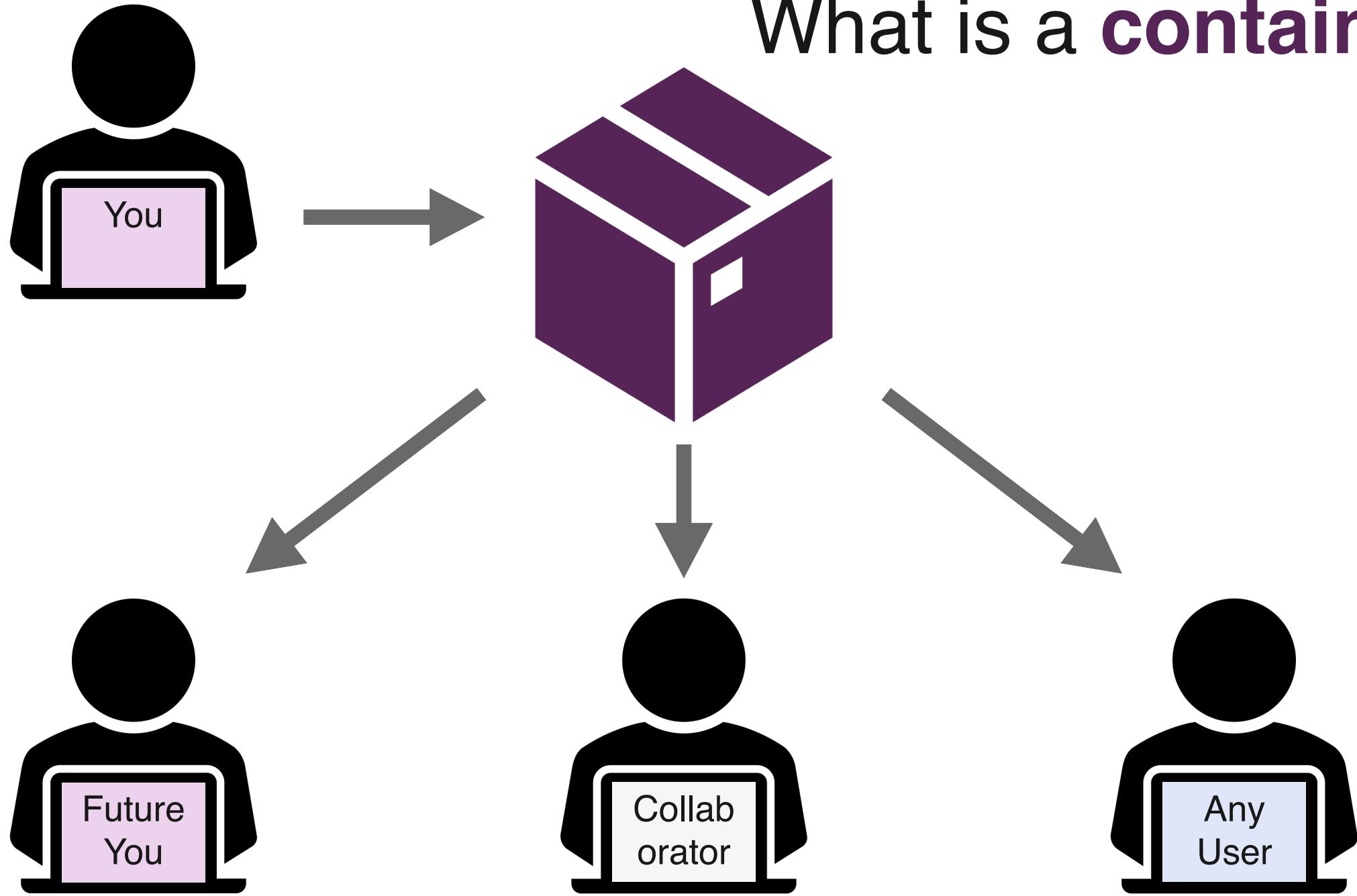


One piece
of the
solution:

a **container**



What is a container?

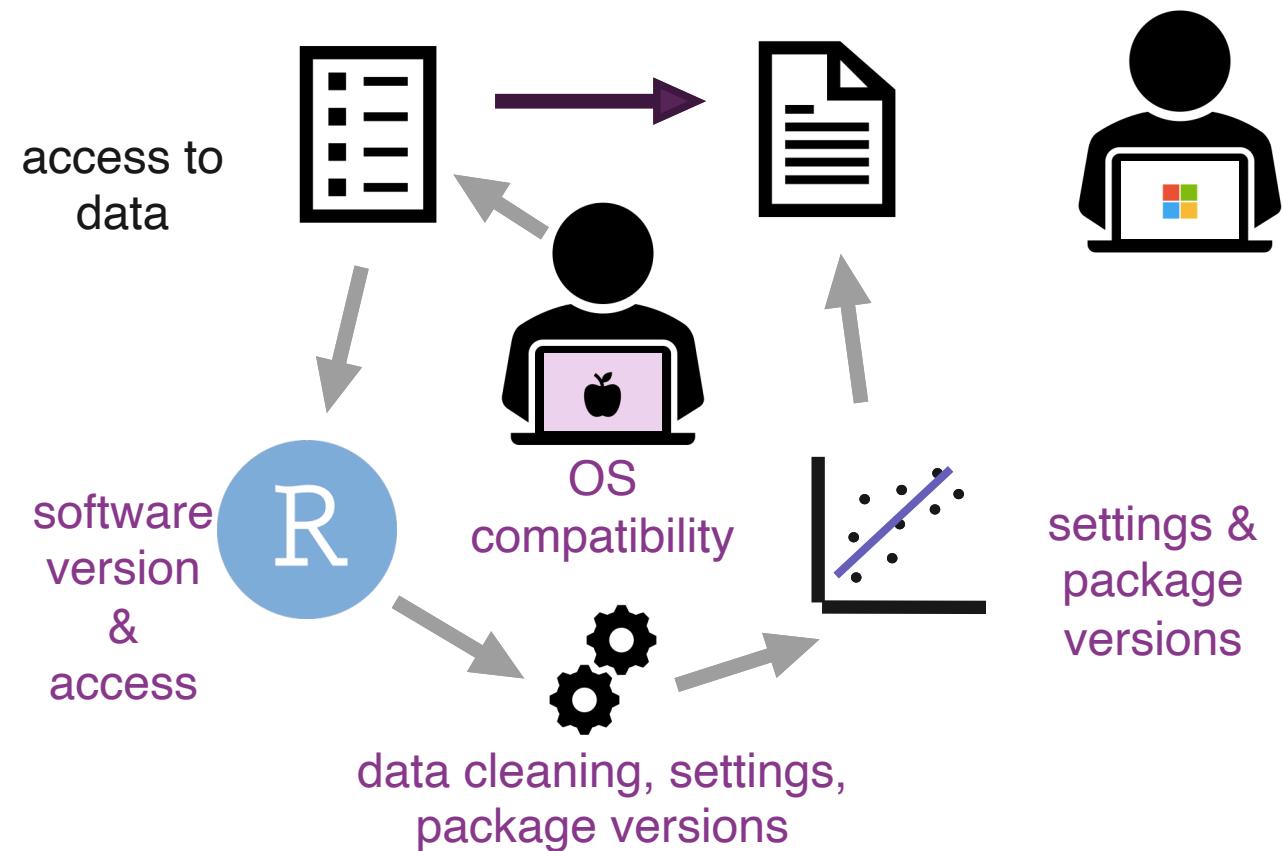


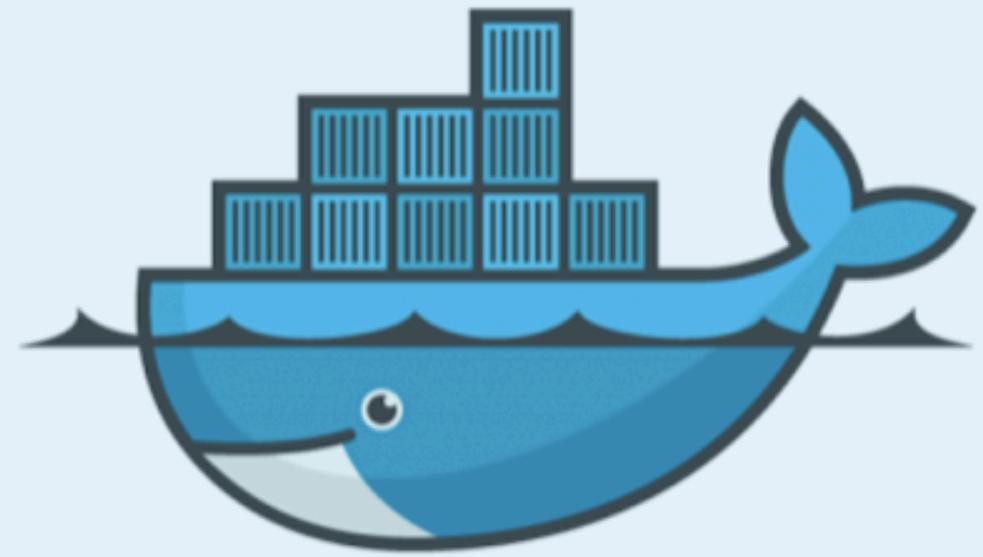
Using containers for reproducible research

1. Write instructions
2. Compile the pieces
3. Make it public
4. Others can reproduce
your analysis

Using containers for reproducible research

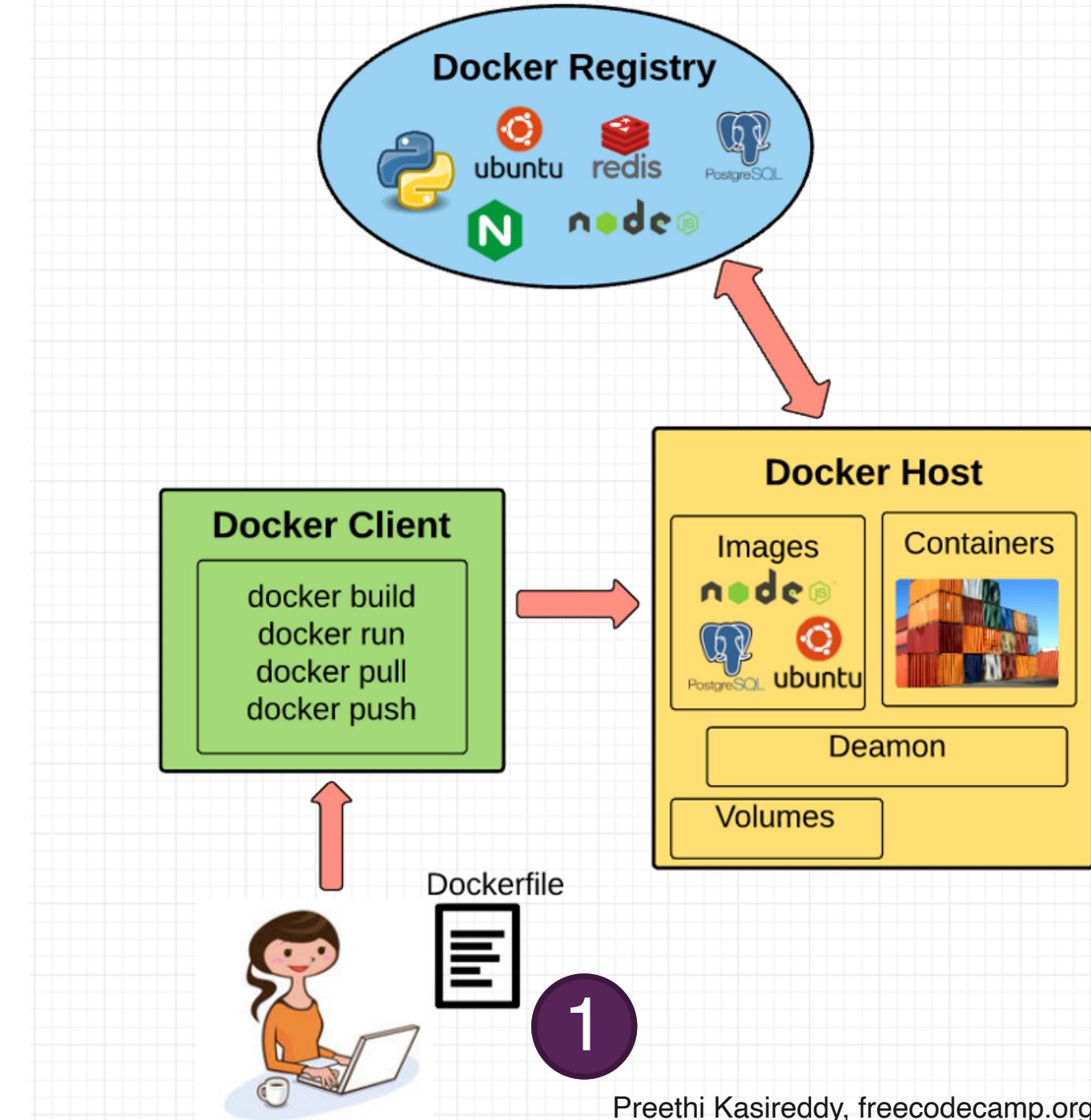
1. Write instructions
2. Compile the pieces
3. Make it public
4. Others can reproduce your analysis



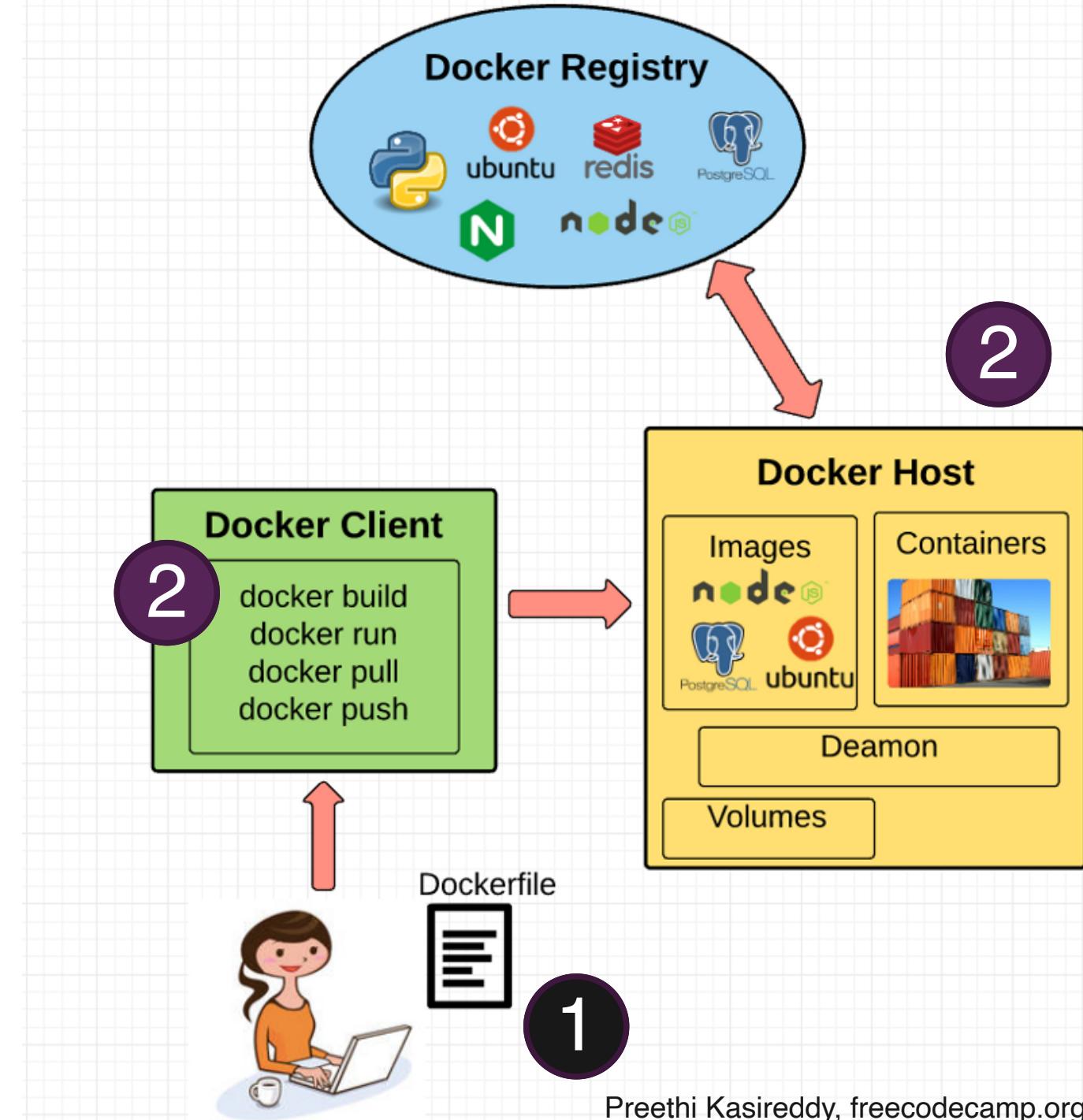


docker

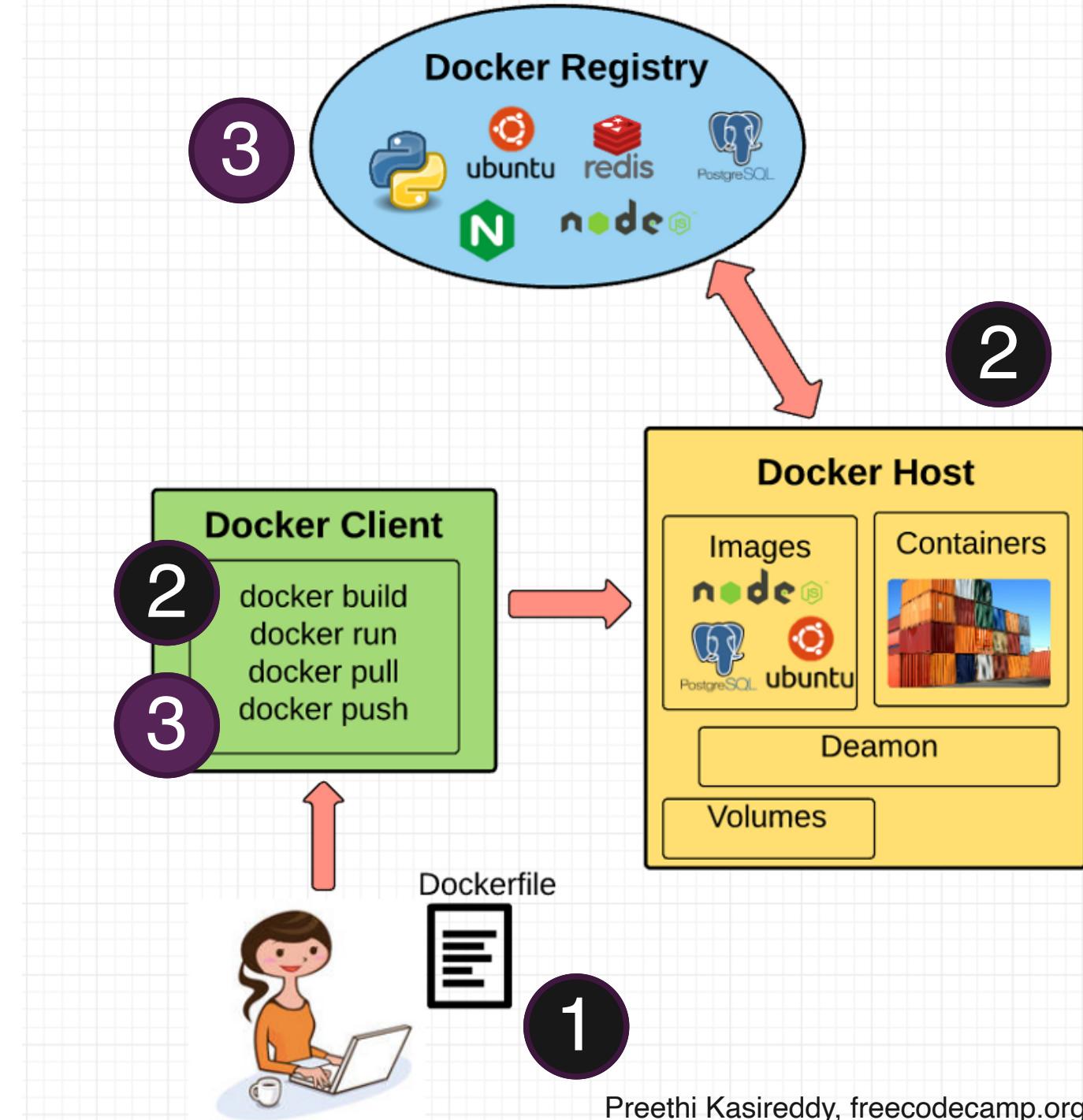
1. Write instructions
2. Compile the pieces
3. Make it public
4. Others can reproduce your analysis



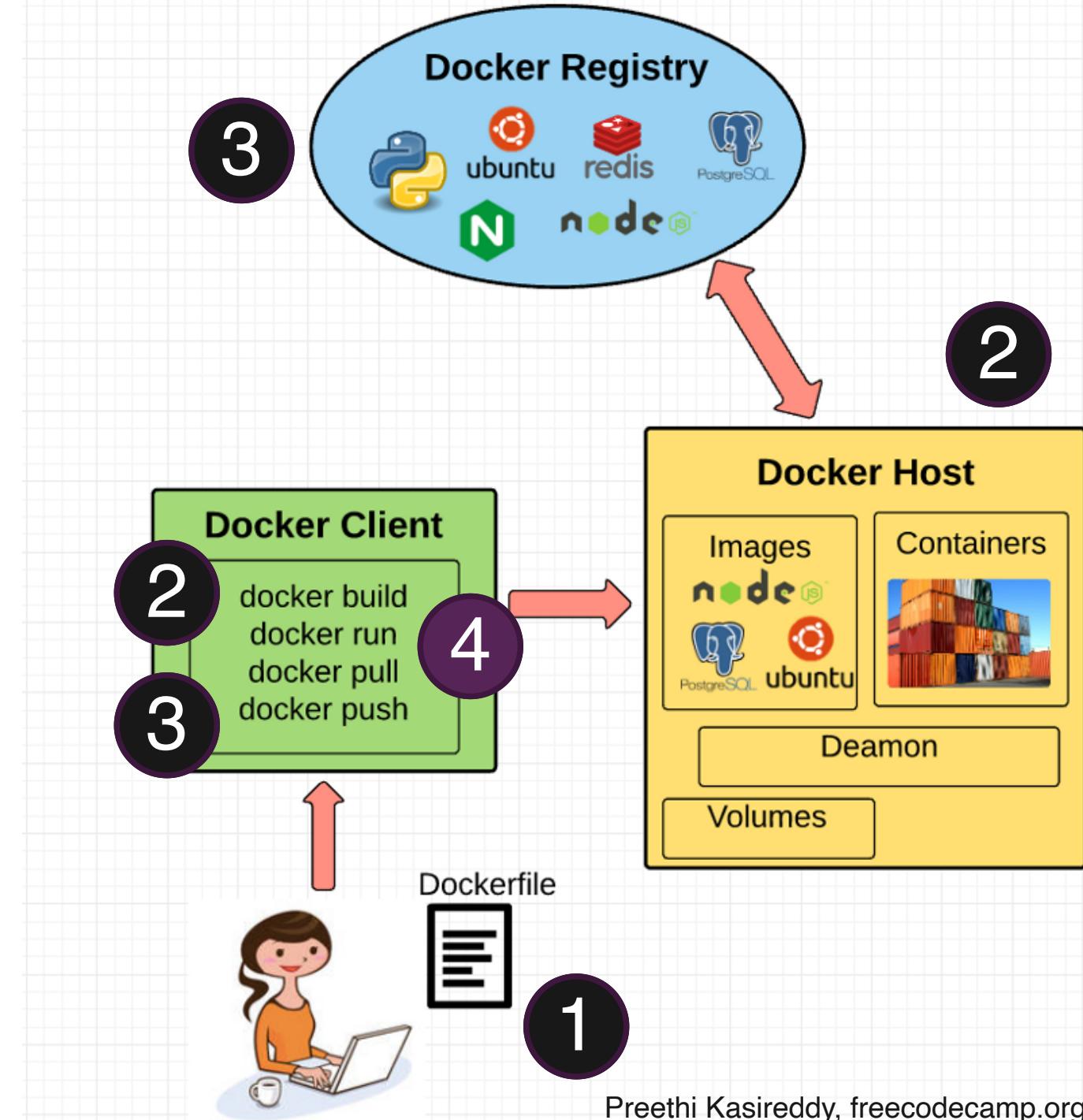
1. Write instructions
2. **Compile the pieces**
3. Make it public
4. Others can reproduce your analysis



1. Write instructions
2. Compile the pieces
- 3. Make it public**
4. Others can reproduce your analysis



1. Write instructions
2. Compile the pieces
3. Make it public
4. **Others can reproduce your analysis**



Today's steps to reproducibility:

1. Run data analysis locally
2. Run template Docker container
3. Build your own Docker container

Today's steps to reproducibility:

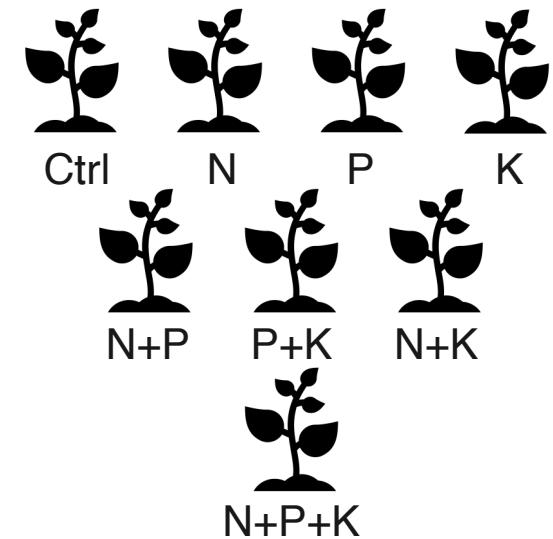
1. Run data analysis locally
2. Run template Docker container
3. Build your own Docker container

<https://github.com/aekendig/docker-rstudio-tutorial>
tutorial-instructions.md

1

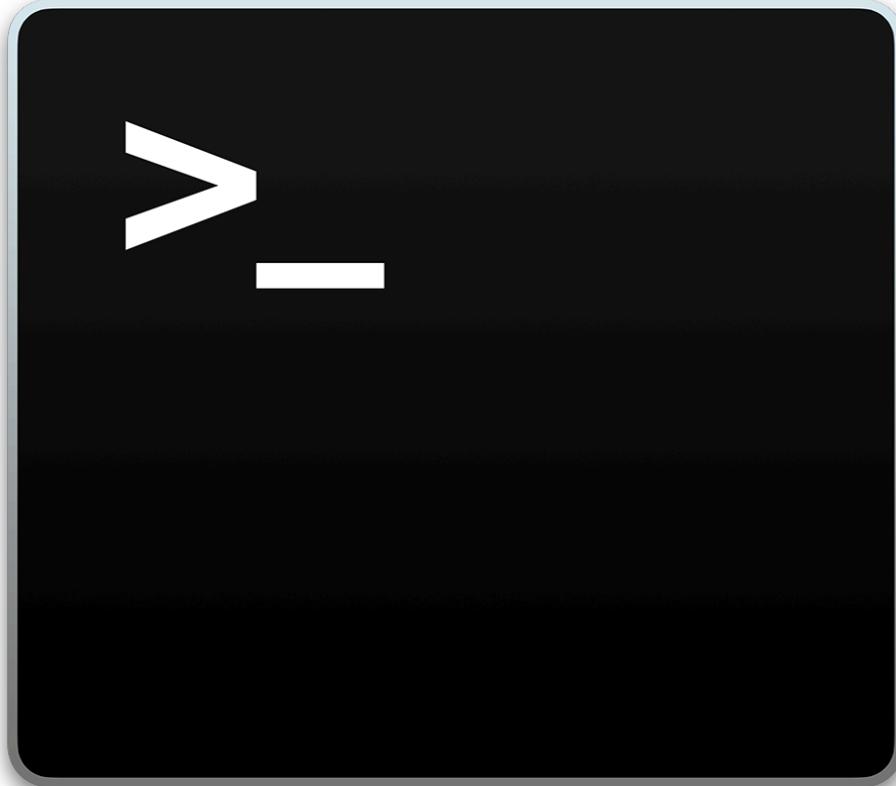
Run data analysis locally

1. Create a new folder (i.e., directory) “docker-rstudio-tutorial” on your desktop*
2. Navigate to: <https://raw.githubusercontent.com/ae kendig/docker-rstudio-tutorial/master/myScript.R>
3. Save the script in “docker-rstudio-tutorial” as myScript.R
 - a. Copy and paste it into a new R script in RStudio
 - b. Browser menu: File → Save Page As...
4. Open the script in RStudio
5. Run the script and examine the output



*I don't encourage saving things on your desktop, but it will make the rest of the tutorial easier if you do

We're going to use command line 😬



We're going to use command line 😬

Mac:

- Finder
- Applications
- Utilities
- Terminal

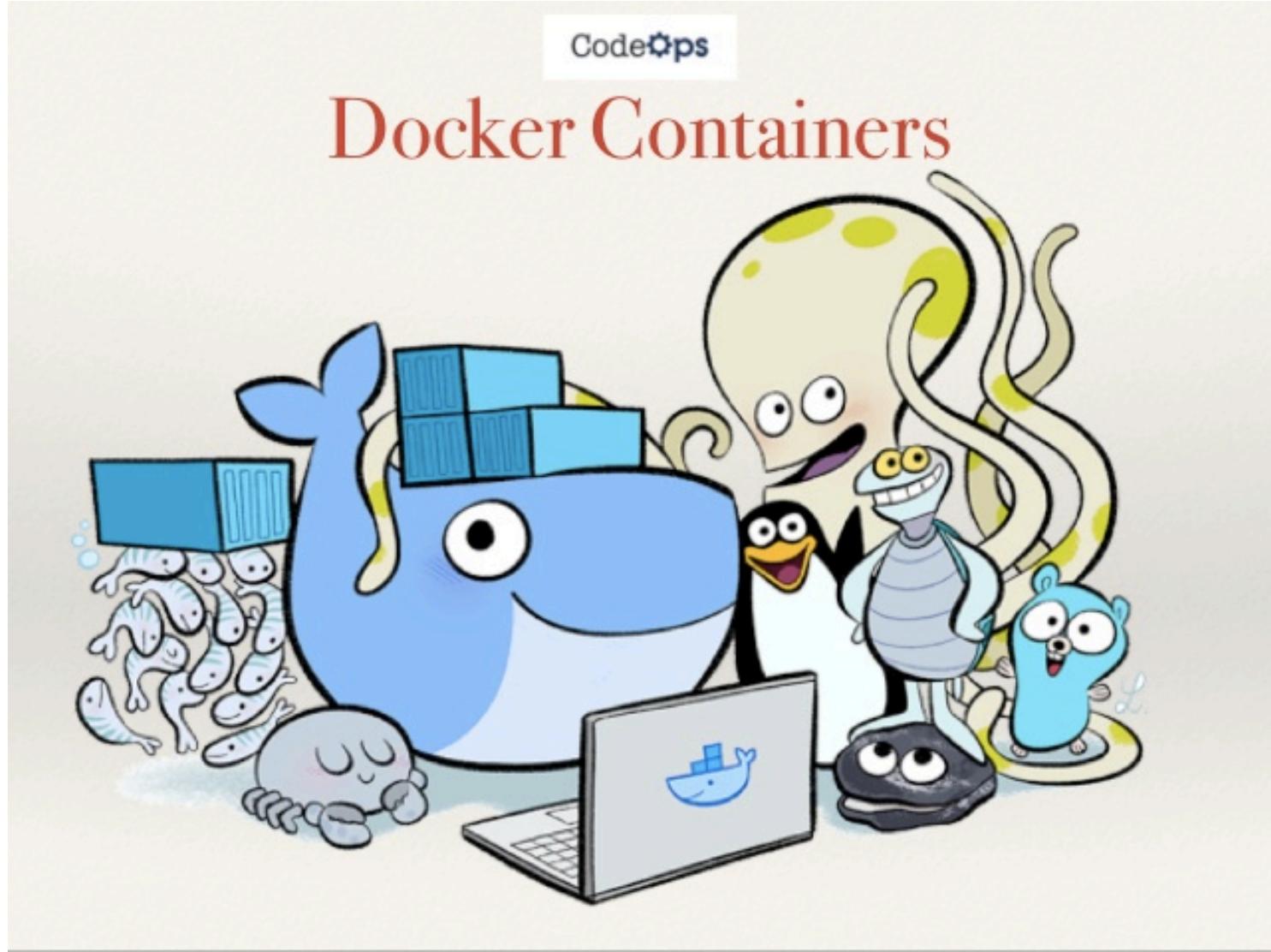
Windows:

- Windows key
- “cmd”
- Right-click and choose “run as administrator”



2

Run template Docker container



Test the Docker installation

```
> docker run hello-world
```

Test the Docker installation

```
> docker run hello-world
```

Hello from Docker!

This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:

1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
(amd64)
3. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it to your terminal.

To try something more ambitious, you can run an Ubuntu container with:

```
$ docker run -it ubuntu bash
```

Share images, automate workflows, and more with a free Docker ID:

```
https://hub.docker.com/
```

For more examples and ideas, visit:

```
https://docs.docker.com/get-started/
```

Save and check three variables*

```
> rstudio_username=<enter-your-new-RStudio-username>
> echo $rstudio_username
> rstudio_password=<enter-your-new-RStudio-password>
> echo $rstudio_password
> docker_username=<enter-your-Docker-Hub-username>
> echo $docker_username
```

*Alternative code for Windows on the tutorial website

Run a Docker container

```
> docker run -d -e USER=$rstudio_username  
           -e PASSWORD=$rstudio_password  
           -p 8787:8787 rocker/tidyverse
```



all
one
line

Windows: substitute \$text for %text%

Run a Docker container

```
> docker run -d -e USER=$rstudio_username  
           -e PASSWORD=$rstudio_password  
           -p 8787:8787 rocker/tidyverse
```



all
one
line

All Docker commands start with “docker”
Here, we’re using the “run” command

Run a Docker container

```
> docker run -d -e USER=$rstudio_username  
           -e PASSWORD=$rstudio_password  
           -p 8787:8787 rocker/tidyverse
```



all
one
line

“-[letter(s)]” is a way to specify options

-d = detached

The container will run in the background

Run a Docker container

```
> docker run -d -e USER=$rstudio_username  
           -e PASSWORD=$rstudio_password  
           -p 8787:8787 rocker/tidyverse
```

} all
one
line

-e = environment variable

Set the username and password

Run a Docker container

```
> docker run -d -e USER=$rstudio_username  
           -e PASSWORD=$rstudio_password  
           -p 8787:8787 rocker/tidyverse
```

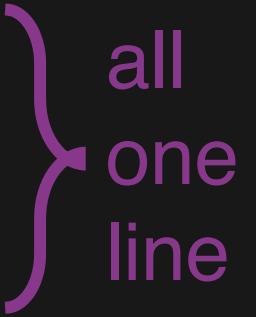


all
one
line

Maps a port from inside of the Docker container to your computer, which you'll access through a web browser

Run a Docker container

```
> docker run -d -e USER=$rstudio_username  
           -e PASSWORD=$rstudio_password  
           -p 8787:8787 rocker/tidyverse
```



all
one
line

The image that we're using to run the container

The Rocker Project maintains R images for Docker

Others include: rocker/rstudio, rocker/geospatial, etc.

Run a Docker container

```
> docker run -d -e USER=$rstudio_username  
           -e PASSWORD=$rstudio_password  
           -p 8787:8787 rocker/tidyverse
```

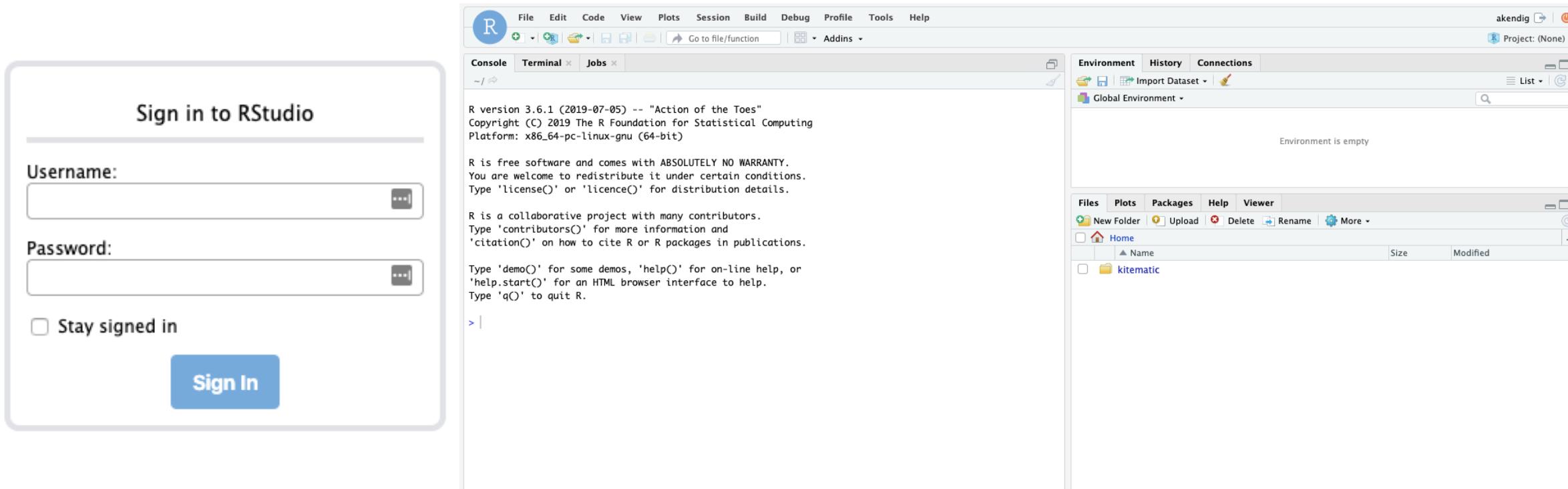


all
one
line

Now that it's done, type “localhost:8787” into your browser

Run a Docker container

Now that it's done, type “localhost:8787” into your browser



Stop the Docker container

```
> docker ps
```

Get a list of running docker containers

Copy the container ID (letters & numbers)

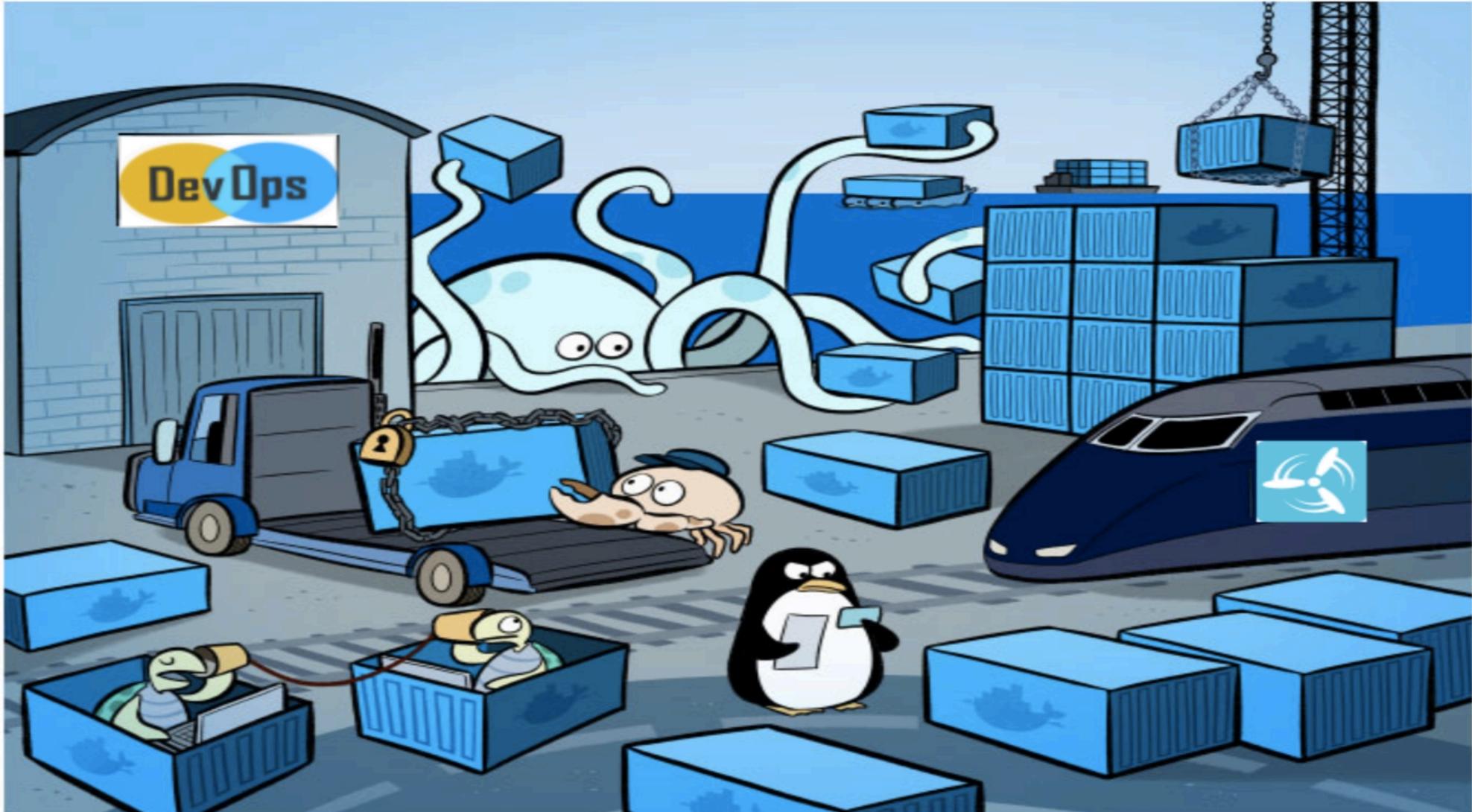
Stop the Docker container

```
> docker stop <paste-container-ID>
```

[Stop the container](#)

[Paste the container ID](#)

3 Build your own Docker container



Navigate to the folder we created

```
> cd Desktop/docker-rstudio-tutorial
```

Not in your home directory? Find out where you are.

Mac: pwd

Windows: echo %cd%

Open a text editor and paste the Dockerfile text

Mac:

```
> vi Dockerfile
```

paste text

[esc]

:wq

[enter]

Windows:

```
> copy con Dockerfile
```

paste text

[ctrl] + z

[enter]

Dockerfile text

```
FROM rocker/tidyverse:3.6.1
```

```
RUN R --no-restore --no-save
```

```
-e 'devtools::install_version("nlme", version="3.1-137")'  
-e 'devtools::install_version("ggplot2", version="3.1.1")'
```

```
ADD myScript.R /home/rstudio/
```

} all one
line

Dockerfile text

```
FROM rocker/tidyverse:3.6.1
```

```
RUN R --no-restore --no-save
```

```
-e 'devtools::install_version("nlme", version="3.1-137")'
```

```
-e 'devtools::install_version("ggplot2", version="3.1.1")'
```

} all one
line

```
ADD myScript.R /home/rstudio/
```

Not recommended for large objects. Other options:

- Add object to GitHub or a data repository
- Include Dockerfile instructions to clone the repository/download the object
- Run container with a volume to a local directory containing the object

Build the container

```
> docker build -t $docker_username/docker- } all  
rstudio-tutorial:1.0 . } one  
line
```

“:1.0” adds the tag 1.0 in case you want to update it with newer versions (e.g., 2.0)

Run the container

```
> docker run -d -e USER=$rstudio_username -e  
PASSWORD=$rstudio_password -p 8787:8787  
$docker_username/docker-rstudio-tutorial:1.0
```

all
one
line

This should look familiar – the structure is the same as when we ran the template container

Push the image

```
> docker login  
> docker push $docker_username/docker-  
rstudio-tutorial:1.0
```



all
one
line

Navigate to Docker Hub (docker.com)
to see your image!

Stop the Docker container

```
> Docker ps  
> docker stop <paste-container-ID>
```

Try out your neighbor's container

```
> docker pull <insert-their-docker-username>/docker-rstudio-tutorial:1.0 } all one line
```

```
> docker run -d -e  
USER=$rstudio_username -e  
PASSWORD=$rstudio_password -p  
8787:8787 <insert-their-docker-username>/docker-rstudio-tutorial:1.0 } all one line
```

Example volume

```
> docker run -d -e USER=$rstudio_username  
-e PASSWORD=$rstudio_password -p 8787:8787  
-v ~/Desktop/docker-rstudio-tutorial:/home/rstudio  
<Docker username>/docker-rstudio-tutorial:1.0
```

Example containers

<https://github.com/benmarwick/1989-excavation-report-Madjedbebe/blob/master/Dockerfile>

<https://github.com/benmarwick/researchcompendium/blob/master/Dockerfile>

Note:

mran.microsoft.com