Cheatsheet: fast container builds for R 🚀

docker command line

Build and push from Dockerfile

```
docker build -t namespace/repo:tag .
docker push
```

Run code inside container

```
docker run --rm namespace/repo:tag \
Rscript script.R
```

Run rocker/rstudio container

```
docker run --rm -it -p 8787:8787 \
-e PASSWORD=mypassword rocker/rstudio
# find it at http://localhost:8787
```

Convenient R commands

Dockerfile from DESCRIPTION file

```
dockerfiler::dock_from_desc()
```

Package system requirements

```
pak::pkg_system_requirements(
    "DT", "ubuntu", "20.04")
```

Binary or source installation?
For most applications, a *binary installation is recommended*. It's faster and it's easier to handle dependencies.

Acknowledgements
Rocker and r2u are developed by Carl Boettiger and Dirk Eddelbuettel. This cheatsheet was created by Geert van Geest

Dockerfile examples

apt-get
interacts with the system package manager, that often includes R packages

```
FROM ubuntu

RUN apt-get update \
    && apt-get install -y \
    --no-install-recommends \
    r-cran-rmarkdown \
    r-cran-dt \
    r-bioc-biostrings
```

```
FROM r-base:latest

RUN apt-get update \
    && apt-get install -y \
    --no-install-recommends \
    r-cran-rmarkdown \
    r-cran-dt \
    r-bioc-biostrings
```

install2.r script
is a helper for installing R packages from the littler package. Find all options here.

```
FROM eddelbuettel/r2u:jammy

RUN install2.r \
    rmarkdown \
    DT \
    Biostrings
```

```
FROM rocker/rstudio:4

RUN install2.r \
    rmarkdown \
    DT \
    Biostrings
```

Choosing your base image

fastest! 🚀

The base images below have many different characteristics. Refer to their docs for a full overview.

	rocker				r2u
	Versioned	Base	rocker/r-ubuntu	rocker/r-bspm	eddelbuettel/r2u
apt-get install	No	Yes	Yes	Yes	Yes
repo install2.r or install.packages()	RSPM (binary)	CRAN (source)	CRAN (source)	package manager (binary)	r2u (binary)
base image	ubuntu	debian	ubuntu	debian, ubuntu, fedora, opensuse	ubuntu