DSApps @ TAU 2020: Running HW Assignments

2020 Sem. B, Mon. 8-11

Giora Simchoni

gsimchoni@gmail.com and add #dsapps in subject

> **Please note: This is NOT the Installations Instructions file.**
>
> **For "Playground mode" you may proceed.**
>
> **For "Submission mode" you need to have a Github account, be added to the DSApps Github organization, and have the course admin distribute you your own private copy of a specific HW Assignment repo. You also need Git installed, and either Docker or a local installation of R + RStudio and Python + Jupyter. The "Submission mode" instructions assume you have all these.**

# Contents

1. Playground mode

1.1. The goal of Playground mode

I want you to start playing with data and code. And I don't want you to worry about installations. Using hosting services such as RStudio Cloud, Binder, Google Colab, Kaggle, Observable allow you to ramp up a R/Python/Whatever environment in the press of a button, and that is cool.

1.2. Running a R Assignment in Playground mode in RStudio Cloud

a. Press the suitable RStudio Cloud link in the Assignment repo README.

b. If you haven't signed into RStudio Cloud it will ask you to sign in. If it's your first-time logging into RC, best login with your Github account.

c. RC will load the Assignment project, you can open the appropriate Rmd file, all necessary packages are installed, you can play.

d. However, once you've exited RC, your changes will not be saved.
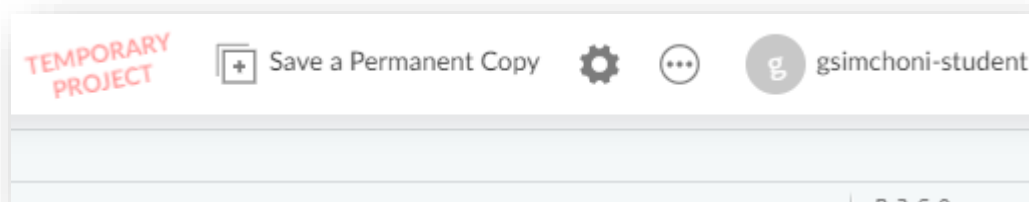
You can instead press "Save a Permanent Copy" to keep a copy of the assignment in your own Workspace, which will persist any changes. This copy is by default private, for your eyes and course admin only.

1.3. Running a Python Assignment in Playground mode in Binder

a. Press the suitable Binder link in the Assignment repo README.

b. Binder will load the Assignment notebook, no need to login.

c. If I did my job correctly the relevant Jupyter notebook should open, if not you'll be in the Jupyter main page and open it up yourself, you can play.

d. However, once you've exited Binder, your changes will not be saved. You *could* technically fork the assignment public repo and launch your own Binder image and save your changes through a git push action inside a Binder terminal. A potential issue with this is that a Binder image is **always public** and so is your public Github repo. Anyone with a link, including other students, can see your work and *that* is a serious violation of university rules. So, don't do it, this is Playground mode anyway. See Submission mode for working with your personal private Assignment repo distributed to you by the course admin.

1.4. Running an Assignment in Playground mode locally

a. Alternatively you can just clone the assignment repo locally (see example below in 2.2.b)

b. Play with the repo with a local installation of R/Python, or with Docker (see below, section 3)

2. Submission mode

2.1. The main idea behind Submission mode

Github is public. If we use free public repos, anyone can see anyone's work and copy it even without the person who wrote the code's knowing. We therefore use a Github "organization", allowing the admin to generate for each student a **private** repository for each assignment. Only you and the admin(s) can view your repo and what you do with it.

2.2. Submission flow

a. Once the admin gets your Github account (via the HW0 survey), and an assignment is released, the admin will create for you your own private repo for this assignment. For example, the assignment repo inside the class organization is "HW1", and your private repo would be called "HW1-john_smith", where "john_smith" is your Github user.

b. You will then clone the repo locally. For example:

```
git clone https://github.com/DSApps-2020/HW1-john_smith.git
```

c. You will work on the notebook (R or Python, local installation or Docker, see below), committing your work as you go:

```
git add .
git commit –m "Solved part 1, this is too easy"
```

d. Once your notebook is ready (R or Python), save it as html (in R this is essentially knitting the notebook, in Python you do "Save as HTML"), be sure to add and commit this html file.

d. Submit your work by pushing it to the repo:

```
git push –u origin master
```

e. When the due date of the assignment comes (usually night before class) the admin will clone all assignment repositories and grade them. Depending on time and no. of students the admin may also give feedback in the form of a Pull Request or Github issue. TBD.

2.3. Warnings

a. Do NOT clone the public assignment to your personal account and work from there. Like any university exercise no one should see your work, so work only with the private repository.

b. For R assignments you can work from RStudio Cloud by opening a RC project or copying the RC assignment project, but you cannot change the access of the project from private to public!

c. For Python assignments you CANNOT work with Binder, as Binder projects are public by default.

d. At any case your private repo is the repo which will be cloned and graded, no other.

3. Working locally on an assignment using Docker

Once you have Docker and a terminal installed (see installations instructions), please consider using the Docker image for the assignment for all the reasons we talked about, mainly reproducibility.

a. Change directory to your local copy of the repo.

b. Your repo should have a Docker folder, in which the Dockerfile is stored. We need to build this Docker image and give it a name. For example for HW0 there are two Dockerfiles: "Dockerfile-R" and "Dockerfile-Python"

Building the R Docker image, naming it "dsapps-r-hw0":

```
docker build -t dsapps-r-hw0 -f docker/Dockerfile-R .
```

Building the Python Docker:

```
docker build -t dsapps-python-hw0 -f docker/Dockerfile-Python .
```

c. See that the image is there:

```
docker image ls
```

d. Run the R Docker image:

```
docker run -it -p 8787:8787 -v ${PWD}:/home/rstudio -e
PASSWORD=password dsapps-r-hw0
```

You need to go to the local port in your browser: http://localhost:8787/

RStudio should load with all the necessary packages for the assignment. If it prompts you for a username and password, give it "rstudio" and "password" (or the one you used in the docker run command above).

Run the Python Docker image:

```
docker run -it -p 8888:8888 -v ${PWD}:/home/jovyan/work/ dsapps-
python-hw0
```

Copy-paste the link given to you to your browser and Jupyter will load (should load in the "work" directory, you need to open to see your local mounted repo).