reinstalling-all-packages

- Organizations will, hopefully, centralize their data science practice
- If your organization is a security conscious one, which I'd hope it is, there is a very good chance you have manually installed your packages into a custom directory and have changed your `.libPath` to use that
- As you move to a centralized server environment you'll likely want to continue to use those libraries since they are what you are comfortable with.
- however, if you are using a tool like RStudio Connect (RSC), this will not be tenable in the long run.
- This is because RSC tries to recreate the package environment that you used and will try to install packages from the same location as where you installed them. If this is a private directory, RSC will likely fail to do so. A longer term solution would to be to have a shared centralized package environment with something like RStudio Package Manager.

To rectify this package environment we can reinstall all of the packages you have in your local installation using RSPM.

To do this we'll first set your default repository to be package manager. We'll use the RStudio public package manager in this case with the RStudio CRAN mirror as the secondary repository.

```r
options(
  "repos" = c("RSPM" = "https://packagemanager.rstudio.com/all/__linux__/bionic/latest",
              "CRAN" = "https://cran.rstudio.com/")
)
```

Next let's identify all of your packages that are installed.

```r
pkgs <- tibble::as_tibble(installed.packages())

head(pkgs)
```

Now to install these packages from RSPM we can just iterateover all of these packages and their respective versions and install using `remotes::install_version()`. First ensure that `remotes` is installed as this will be necessary to revert to a specific version.

The following one line will install each package based on the version you have now.

```r
map2(pkgs$Package, pkgs$Version, remotes::install_version)
```

Alternatively, you can install of the packages to their most recent version using `install.packages(pkgs$Package)`. This will be much faster but will not return to the version you have currently.