4. Options for presenting code to users

Download *.R script/*.Rmd report with `downloadButton`

Use `buildScriptBundle` or `buildRmdBundle` to dynamically generate .zip bundles

```
buildRmdBundle(
  report_template = "report.Rmd",
  include_files = list("data.csv" = downloads_data),
  vars = list(pkgname = input$package, code = code),
  output_zip_path = out
)
```

Using shinymeta (recap)

1. You (the app author) **identify the domain logic in your app code** so we can separate it from the reactive structure

2. Within that domain logic, you **identify references to reactive values and reactive expressions** that need to be replaced with static values and static code, respectively

3. At runtime, **choose which pieces** of domain logic to export, and in what order

4. **Present the code** to the user (in a window, as a downloadable script or report, etc.)