- How do you write a lesson using R Markdown and `{sandpaper}`?

- Explain how to use markdown with the new lesson template
- Demonstrate how to include pieces of code, figures, and nested challenge blocks

Introduction

This is a lesson created via The Carpentries Workbench. It is written in Pandoc-flavored Markdown for static files and R Markdown for dynamic files that can render code into output. Please refer to the Introduction to The Carpentries Workbench for full documentation.

What you need to know is that there are three sections required for a valid Carpentries lesson template:

1. `questions` are displayed at the beginning of the episode to prime the learner for the content.
2. `objectives` are the learning objectives for an episode displayed with the questions.
3. `keypoints` are displayed at the end of the episode to reinforce the objectives.

Inline instructor notes can help inform instructors of timing challenges associated with the lessons. They appear in the “Instructor View”

Challenge 1: Can you do it?

What is the output of this command?

```r
paste("This", "new", "lesson", "looks", "good")
```

```
[1] "This new lesson looks good"
```

Challenge 2: how do you nest solutions within challenge blocks?

You can add a line with at least three colons and a `solution` tag.

Figures

You can also include figures generated from R Markdown:

```r
pie(
  c(Sky = 78, "Sunny side of pyramid" = 17, "Shady side of pyramid" =
      5),
  init.angle = 315,
  col = c("deepskyblue", "yellow", "yellow3"),
  border = FALSE
)
```

Sun arise each and every morning

Or you can use standard markdown for static figures with the following syntax:

Keypoints

- Use `.md` files for episodes when you want static content
- Use `.Rmd` files for episodes when you need to generate output
- Run `sandpaper::check_lesson()` to identify any issues with your lesson
- Run `sandpaper::build_lesson()` to preview your lesson locally