Curriculum outline

with purpose and objectives

teach-shiny.rbind.io

Mine Çetinkaya-Rundel

@minebocek

mine-cetinkaya-rundel

mine@rstudio.com

Goal

Develop the curriculum outline
for a Shiny workshop you're likely to teach
(keeping in mind the likely length and the likely audience)

Your turn

- Go to rstd.io/shiny-wsds-18 and scan the content
- Who is the intended audience? What is their background? How can you tell?

Think 3m 00s

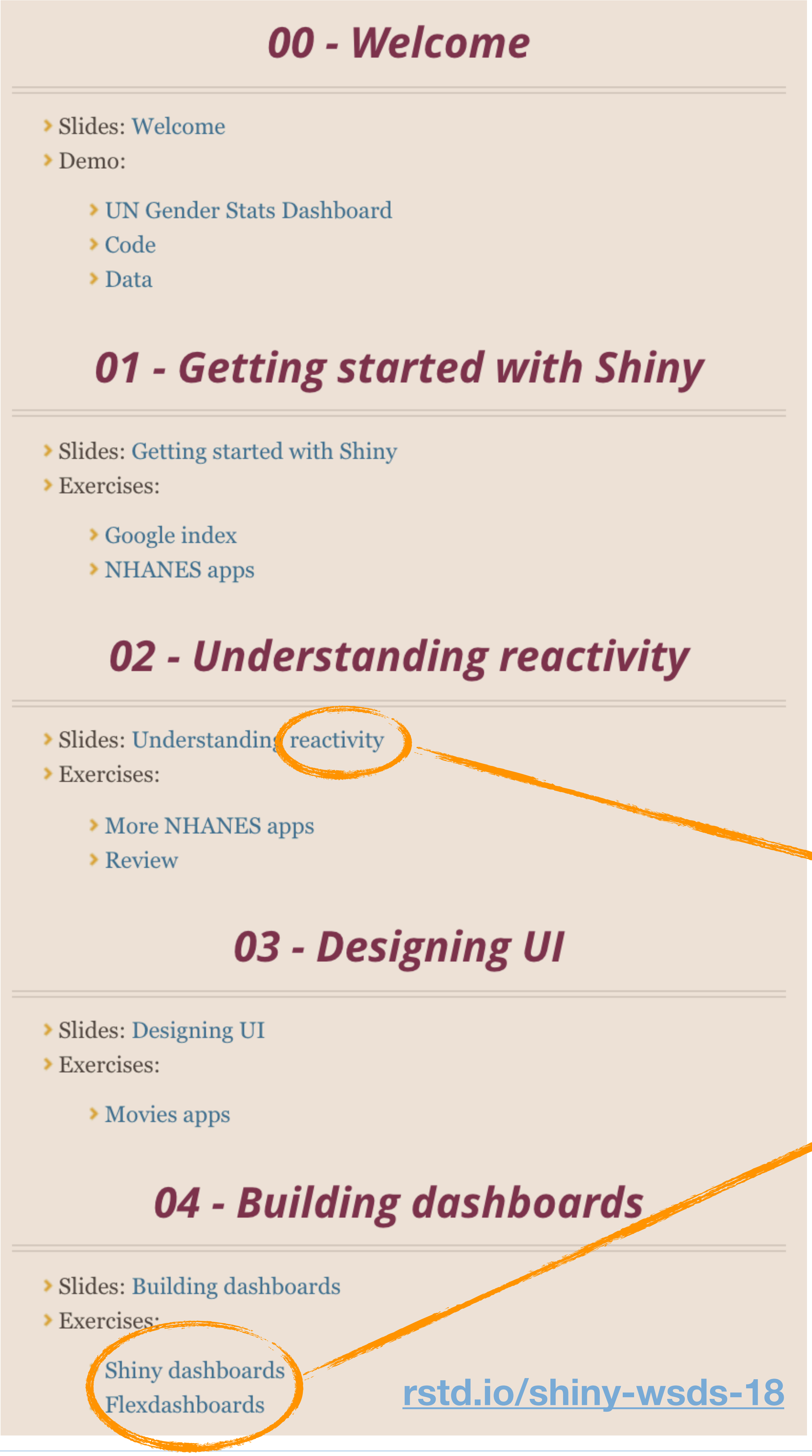
Pair 2m 00s

Your turn

- Go to rstd.io/shiny-wsds-18 and scan the content
- Who is the intended audience? What is their background? How can you tell?

Think 3m 00s

Pair 2m 00s



00 - Welcome

- Slides: Welcome
- Demo:
  - UN Gender Stats Dashboard
  - Code
  - Data

01 - Getting started with Shiny

- Slides: Getting started with Shiny
- Exercises:
  - Google index
  - NHANES apps

02 - Understanding reactivity

- Slides: Understanding reactivity
- Exercises:
  - More NHANES apps
  - Review

03 - Designing UI

- Slides: Designing UI
- Exercises:
  - Movies apps

04 - Building dashboards

- Slides: Building dashboards
- Exercises:
  - Shiny dashboards
  - Flexdashboards

rstd.io/shiny-wsds-18

These words will make sense to a student with some experience with Shiny, but may not be informative for a complete novice to gauge the level of the class.

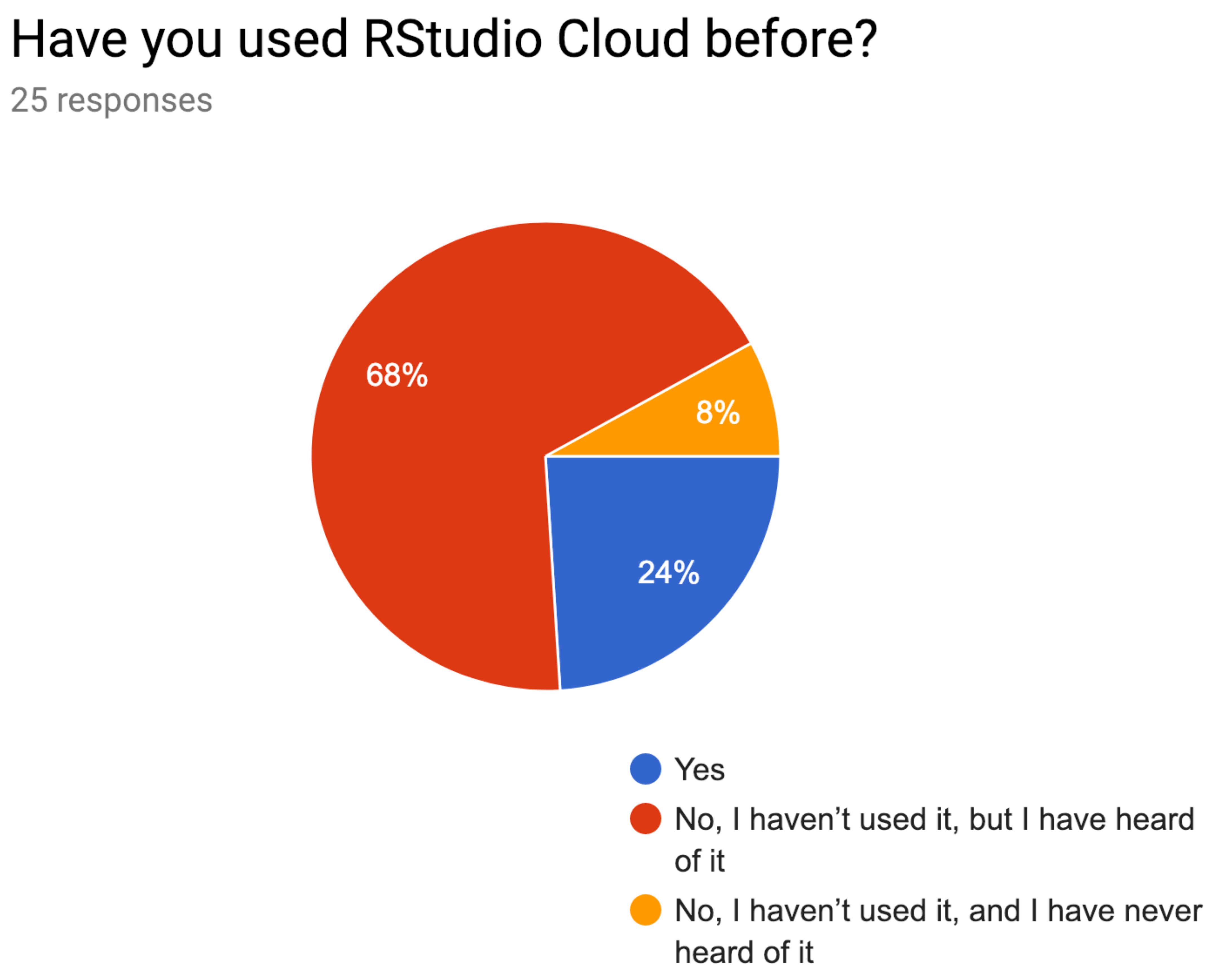
define your audience

Discussion

What are some ways you can get to know your audience prior to your workshop?

Pre-workshop survey

- A pre-workshop survey is helpful, but may not always be feasible
- It's also not helpful if you're planning a curriculum first, and then pitching it for interest



Have you used RStudio Cloud before?

25 responses

Learner personas

Creating learner personals is a technique borrowed from UI designers, who create short profiles (personas) of typical users to help them think about their audience:

1. background
2. prior knowledge
3. motivations or goal(s)
4. how the course will help them
5. special needs

Meet Jorge

A learner persona for a weekend introduction to programming aimed at college students might be:

1. **Background:** Jorge has just moved from Costa Rica to Canada to study agricultural engineering. He has joined the college soccer team, and is looking forward to learning how to play ice hockey.
2. **Prior knowledge:** Other than using Excel, Word, and the Internet, Jorge's most significant previous experience with computers is helping his sister build a WordPress site for the family business back home in Costa Rica.
3. **Motivations or goal(s):** Jorge needs to measure properties of soil from nearby farms using a handheld device that sends logs in a text format to his computer. Right now, Jorge has to open each file in Excel, crop the first and last points, and calculate an average.
4. **How the course will help them:** This workshop will show Jorge how to write a little Python program to read the data, select the right values from each file, and calculate the required statistics.
5. **Special needs:** Jorge can read English well, but still struggles sometimes to keep up with spoken conversation (especially if it involves a lot of new jargon).

Source: Ten Quick Tips for Creating an Effective Lesson

Your turn

- Write three short learner personas for
  - A Shiny novice
  - A Shiny expert
  - A student you expect to encounter at a Shiny workshop you teach

10m 00s

Shiny novice

1. background
2. prior knowledge
3. motivations or goal(s)
4. how the course will help them
5. special needs

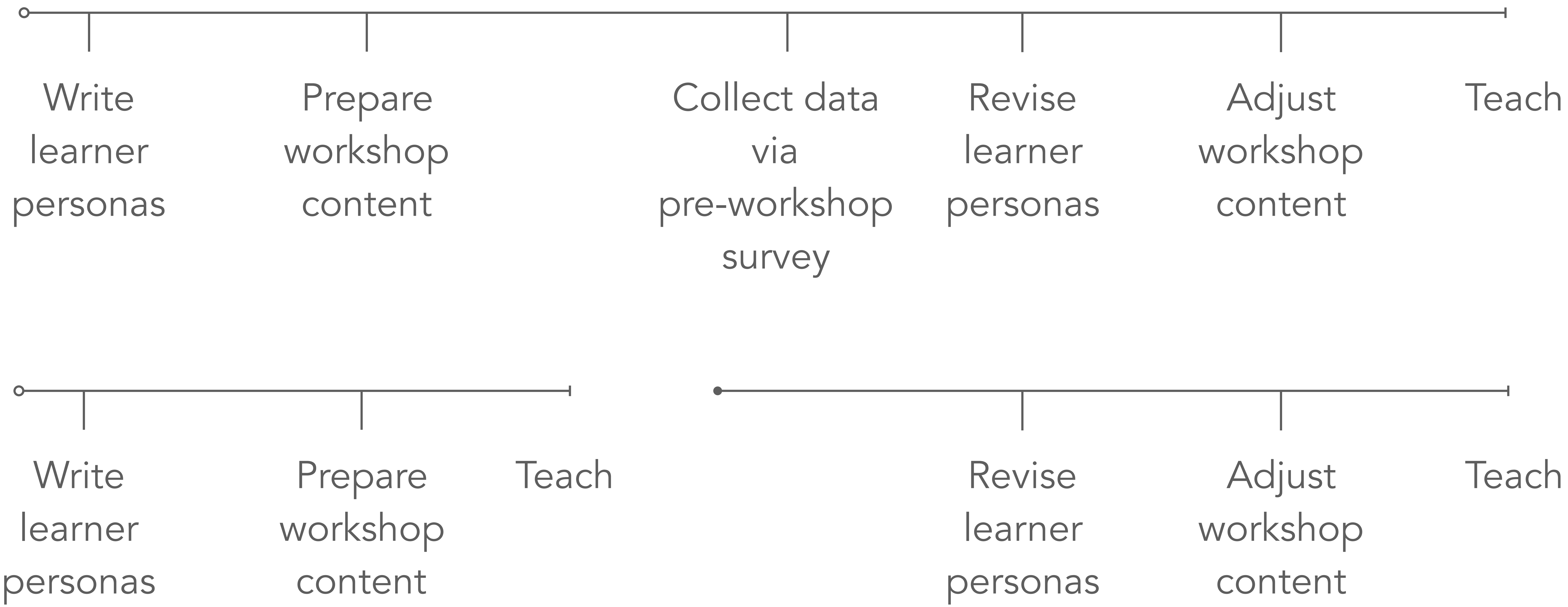
Shiny expert

1. background
2. prior knowledge
3. motivations or goal(s)
4. how the course will help them
5. special needs

Your student

1. background
2. prior knowledge
3. motivations or goal(s)
4. how the course will help them
5. special needs

Planning



pre-workshop survey feasible

Write learner personas

Prepare workshop content

Collect data via pre-workshop survey

Revise learner personas

Adjust workshop content

Teach

pre-workshop survey not feasible

Write learner personas

Prepare workshop content

Teach

Revise learner personas

Adjust workshop content

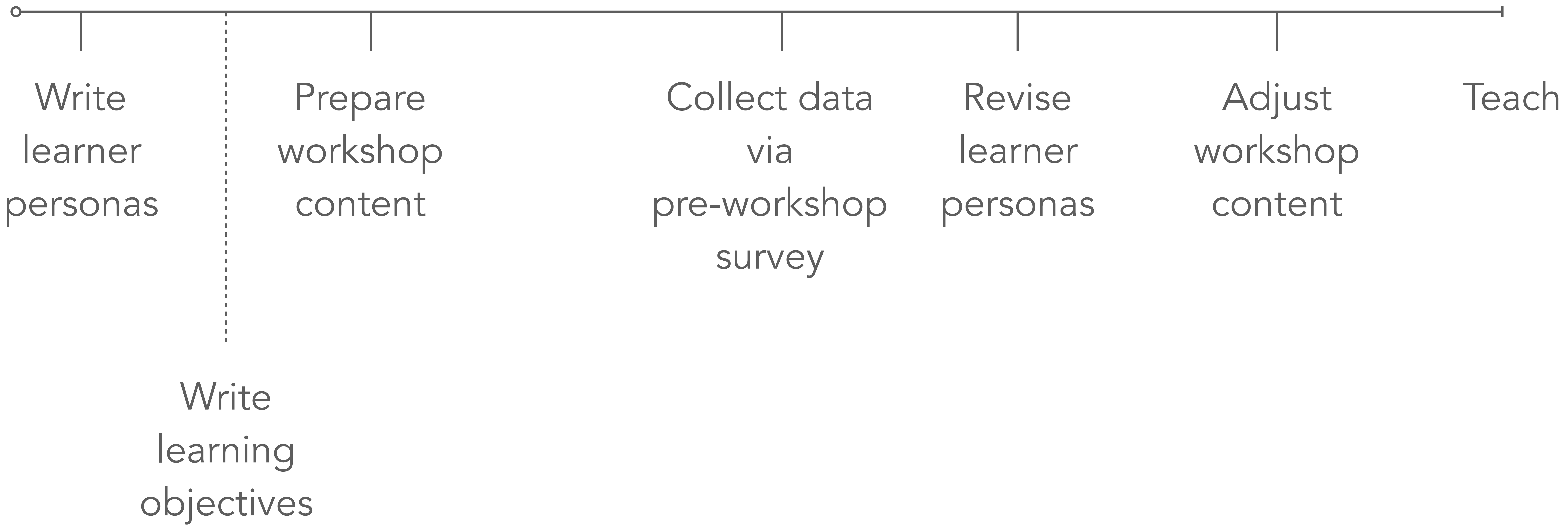
Teach

▶ write learning objectives

Learning objectives

- A learning objective is a goal of a course or lesson stated in a way that is
  - clear: a student should be able to understand
  - observable: a student (and you) should be able to easily check whether or not they have attained the objective
- A learning outcome is what a course actually achieves

Planning



Write learner personas

Write learning objectives

Prepare workshop content

Collect data via pre-workshop survey

Revise learner personas

Adjust workshop content

Teach

For whom the LO tolls

- Students:
  - Those who have some experience with Shiny can better understand the content / coverage of the course
  - Those who are completely new to it can use them as a learning checklist throughout / at the end
- Other instructors: So they can understand / evaluate your course content / coverage at a glance
- Yourself: Keeps you focused and organized

Quiz

Order the following verbs from Bloom’s taxonomy from bottom to top of the pyramid*:

analyze

apply

be aware of

create

evaluate

learn

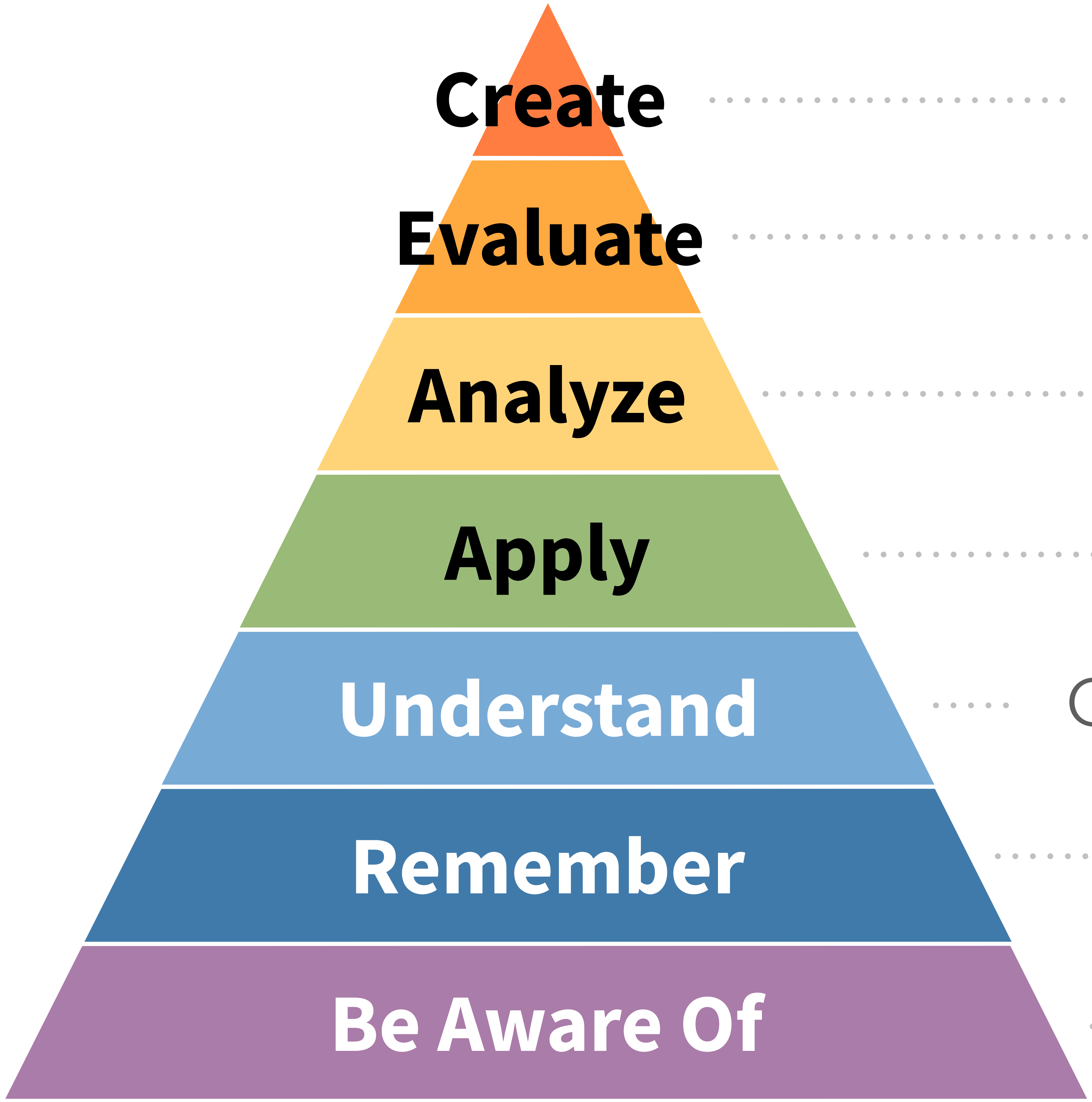
remember

understand

1m 00s

* I don’t know why it’s called a pyramid, it’s a triangle.

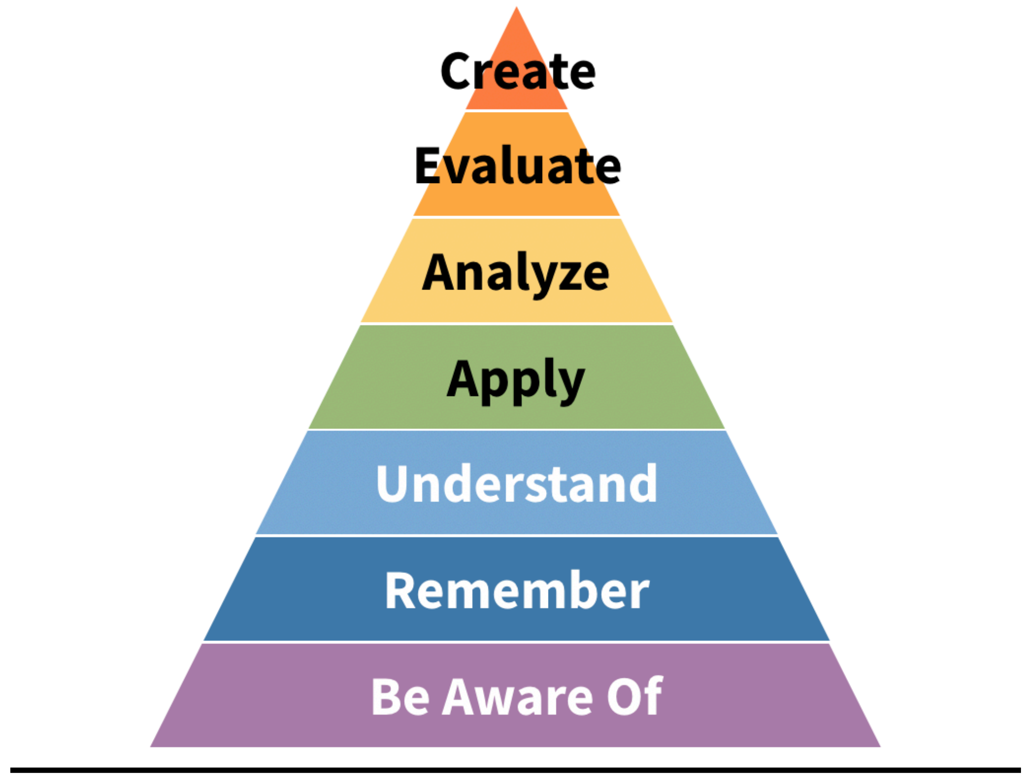
Bloom’s taxonomy



Create	Make new or original work
Evaluate	Justify opinions, decisions
Analyze	Identify components and their relationships
Apply	Use in novel situations
Understand	Comprehend, explain ideas
Remember	Recall facts
Be Aware Of	Know what to search for

Anatomy of a LO

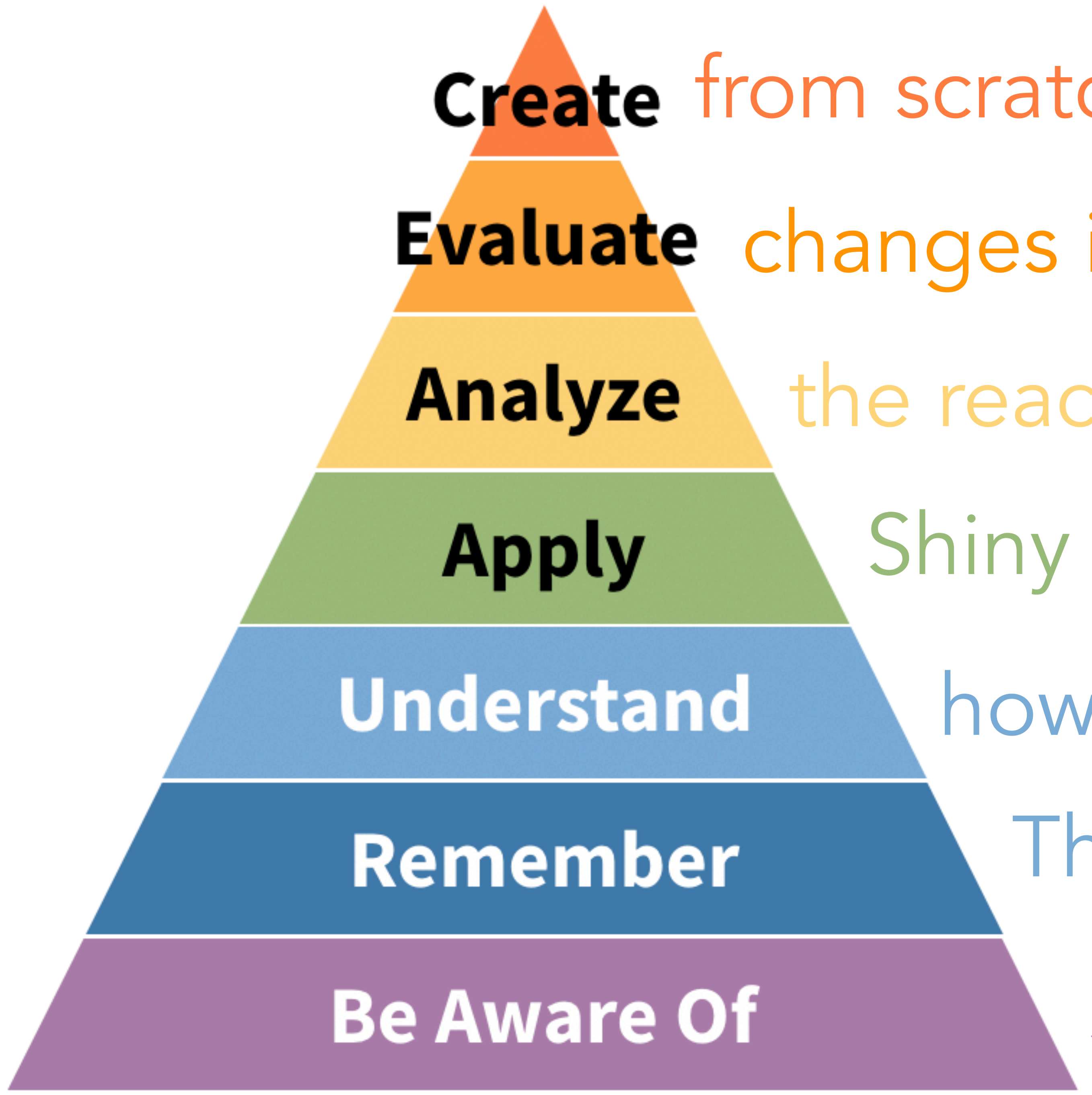
A traditional approach to writing learning objectives follows the form:



something...

while (IMHO) this is not a hill worth dying on...

it's important to realize that there are **BIG** differences between



Create from scratch Shiny app with a complex reactive structure using **modules**

Evaluate changes in the react log after refactoring an app with **modules**

Analyze the react log to identify how your app can code be improved with **modules**

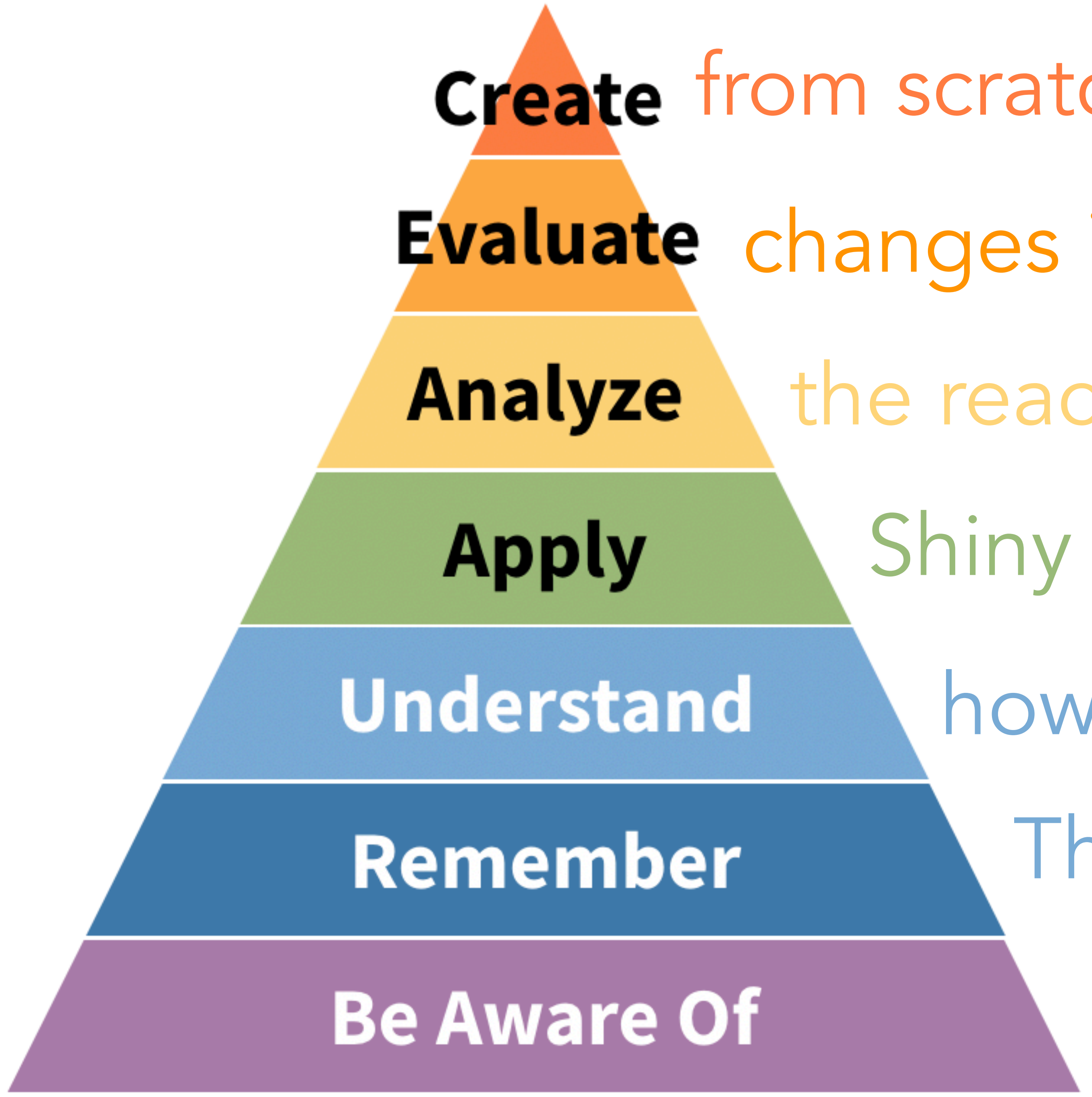
Apply Shiny **modules** to refactor code for an app with a complex reactive structure

Understand how Shiny **modules** make use of namespaces

Remember That use of modules requires

Be Aware Of Shiny **modules** for managing complex Shiny apps

and an even bigger difference between



Create from scratch Shiny app with a complex reactive structure using **modules**

Evaluate changes in the react log after refactoring an app with **modules**

Analyze the react log to identify how your app can code be improved with **modules**

Apply Shiny **modules** to refactor code for an app with a complex reactive structure

Understand how Shiny **modules** make use of namespaces

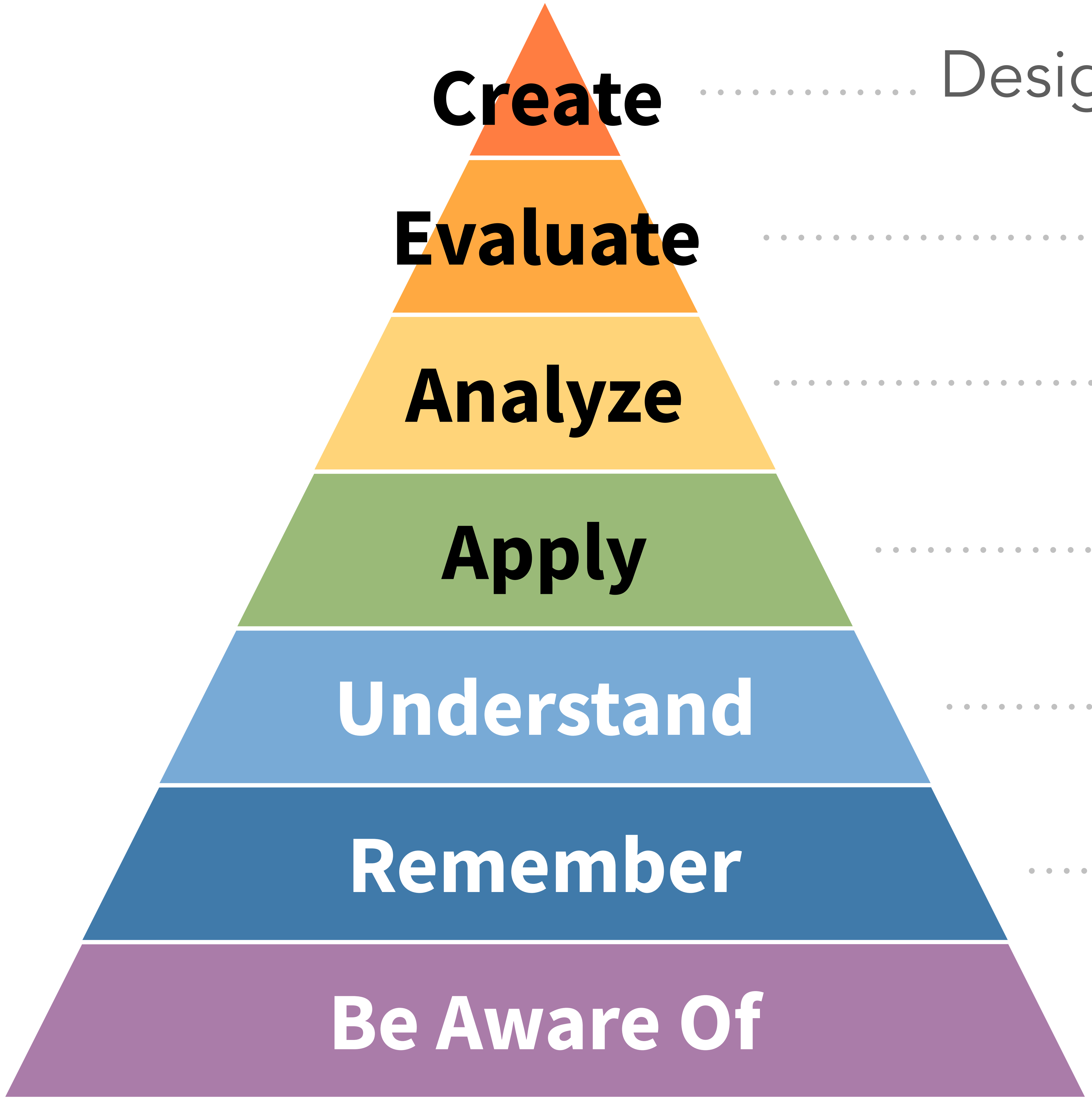
Remember That use of modules requires

Be Aware Of Shiny **modules** for managing complex Shiny apps

vs.

Learn how to use Shiny modules

Bloom's taxonomy ++



Create .... Design, Construct, Improve, Adapt, Make, Refactor, Program

Evaluate .... Check, Choose, Critique, Prove, Rate, Review

Analyze .... Compare, Contrast, Simplify, Debug, Change

Apply .... Find, Write, Compute, Use, Plan, Return

Understand .... Summarize, Predict, Explain, Comment, Complete

Remember .... Recognize, List, Describe, Name, Find the function

Be Aware Of Search for, Know where to look, Look up, Find help, Ask

Your turn

- Improve the following not-so-ideally written learning objectives for the Shiny workshop we considered earlier:
  - Design a Shiny app from scratch
  - Understand the essentials of reactive programming in Shiny
  - Customize reactive objects in your app for better performance
  - Customize the user interface of your app
  - Build interactive dashboards
  - Explore deployment options for sharing your app
- Note that the workshop assumes experience with R, but not Shiny

Think 5m 00s

Pair 3m 00s

Your turn

- Improve the following not-so-ideally written learning objectives for the Shiny workshop we considered earlier:
  - Design a Shiny app from scratch
  - Understand the essentials of reactive programming in Shiny
  - Customize reactive objects in your app for better performance
  - Customize the user interface of your app
  - Build interactive dashboards
  - Explore deployment options for sharing your app
- Note that the workshop assumes experience with R, but not Shiny

Think 5m 00s

Pair 3m 00s

plan

▶ your

time

Discussion

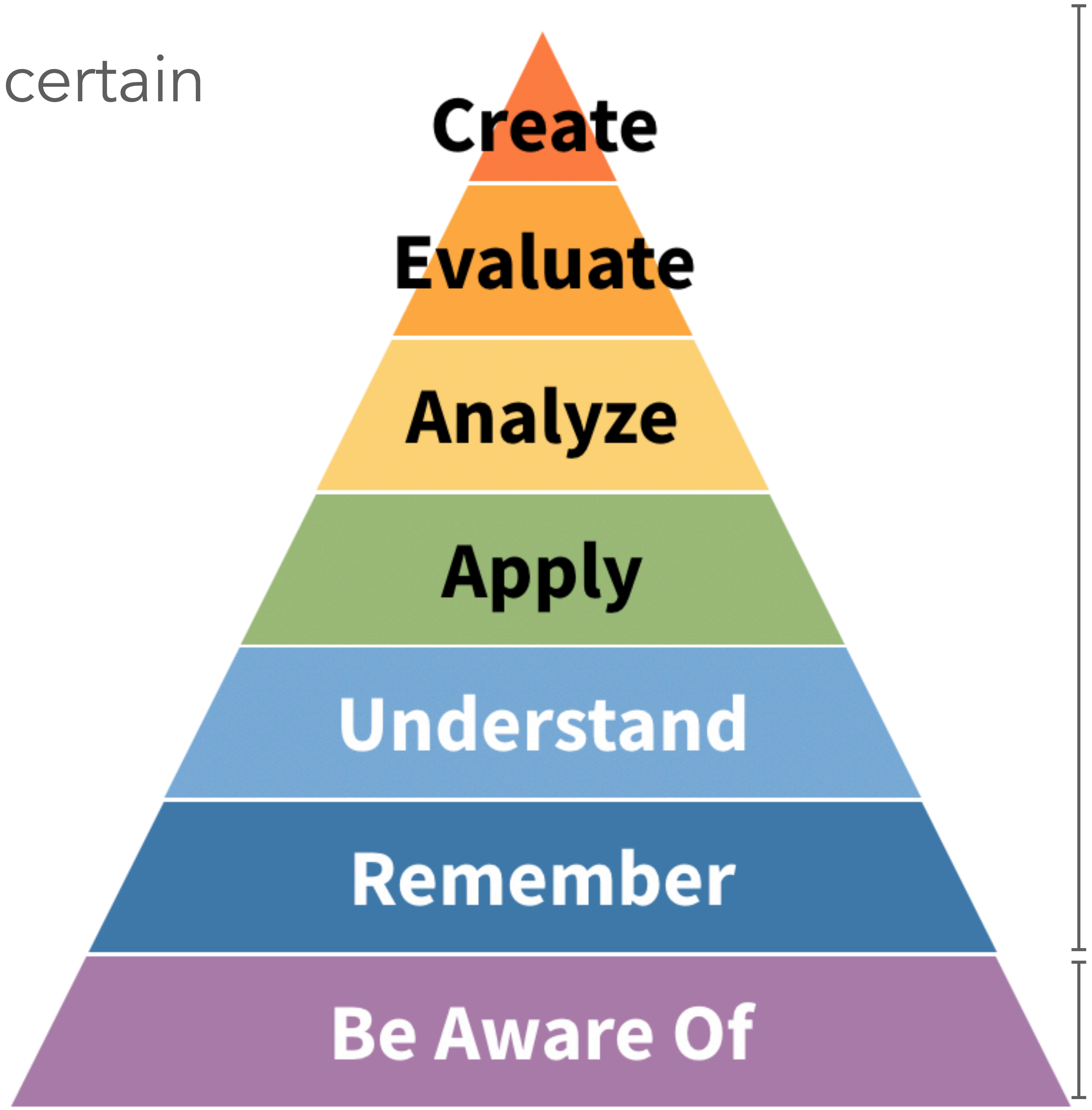
Evaluate whether it's feasible to turn these learning objectives into learning outcomes in a half-day (4 hr, with breaks) workshop.

How does one even
start answering this question?

Method of delivery

How long it takes to deliver certain material depends on

- Topics covered
- Level of desired mastery



Create

Evaluate

Analyze

Apply

Understand

Remember

Be Aware Of

Active learning

more on this later…

Lecture

Your turn

- Write learning objectives for a Shiny workshop for beginners (assume familiarity with R, but not Shiny) for one of the following time periods
  - 1 hour
  - 2 hours
  - half day
  - full day
  - 2 days
- Remember to be clear and observable — include topics and functions/packages
- Compare your outlines to your teammates

Think 5m 00s

Pair 3m 00s

Your turn

- Write learning objectives for a Shiny workshop for beginners (assume familiarity with R, but not Shiny) for one of the following time periods
  - 1 hour
  - 2 hours
  - half day
  - full day
  - 2 days
- Remember to be clear and observable — include topics and functions/packages
- Compare your outlines to your teammates

Think 5m 00s

Pair 3m 00s

Running out of time

- Design your curriculum in modules, and skip as needed
- For each module, have a back up plan for self-study
- Take a minute to describe how the self-study would work, instead of rushing through the module content

Your turn

- Suppose for an intermediate Shiny workshop you will teach the following functions:
  - `eventReactive()`
  - `invalidateLater()`
  - `isolate()`
  - `observeEvent()`
  - `reactiveFileReader()`
  - `reactivePoll()`
- Work in teams to sort this content into modules
- Tip: You should be able to describe quickly what each module is about

3m 00s

Running out of material

- Open-ended, long activity:
  - e.g. build an app from scratch
  - Can fill as little or as much time as you have
  - Multi-step:
    - Step 1: Students who absorbed ~50% of what was taught should be able to complete
    - Step 2: Students who absorbed everything taught should be able to complete
- Office hours: Use pre/mid workshop survey to collect info on Shiny hurdles your students have come across and use live coding to go through a sample of them

Sample curricula

Add link to repo with curricula / materials

- define your audience
- write learning objectives
- plan your time

Something to ponder

🤔 What does a curriculum for Shiny for people with no background in R look like?