

Code Sleuth

Instructions

This exercise challenges you to figure out how R accesses elements within vectors.

- First, you'll see tables with code examples and their outputs. Use *these to infer how R accesses elements within vectors*.
- Then, use your *inferences to predict what the outputs will be* for the subsequent code examples.

For this exercise, assume the following code has run first:

```
survey_years <- seq(2008, 2024, by = 2)
coral_cover_pct <- c(34.53, 22.26, 11.66, 9.25, 14.05, 22.30, 30.35,
16.35, 5.74)
n_observations <- c(Acropora = 13882, Pocillopora = 13988, Porites =
3192)
```

Indexing with positive integers

Inference

Code example	Output
<code>survey_years[1]</code>	<code>[1] 2008</code>
<code>survey_years[3:5]</code>	<code>[1] 2012 2014 2016</code>
<code>survey_years[c(2, 4, 6)]</code>	<code>[1] 2010 2014 2018</code>

What do square brackets `[]` do?

Index into a data structure

Describe how R accesses elements in a vector when you use *positive integers* as the **index**.

Access data by position

Prediction

What will be the output of the following expressions?

Code example	Output
<code>coral_cover_pct[8:9]</code>	
<code>coral_cover_pct[c(1, length(coral_cover_pct))]</code>	

Indexing with negative integers

Inference

Code example	Output
<code>survey_years[-9]</code>	<code>[1] 2008 2010 2012 2014 2016 2018 2020 2022</code>
<code>survey_years[-(1:3)]</code>	<code>[1] 2014 2016 2018 2020 2022 2024</code>
<code>survey_years[c(-3, -5, -7)]</code>	<code>[1] 2008 2010 2014 2018 2022 2024</code>

Describe how R accesses elements in a vector when you use *negative integers* as the **index**.

Drop elements by position

Prediction

Using the type coercion rules you inferred above, what will be the output of the following expressions?

Code example	Output
<code>coral_cover_pct[-c(1, 3, 5, 7, 9)]</code>	
<code>coral_cover_pct[-(2:length(coral_cover_pct))]</code> <code>]</code>	

Indexing with names

Inference

Code example	Output
<code>n_observations["Acropora"]</code>	Acropora 13882
<code>n_observations[c("Pocillopora", "Porites")]</code>	Pocillopora Porites 13988 3192

Recall the expression:

```
n_observations <- c(Acropora = 13882, Pocillopora = 13988, Porites = 3192)
```

Where do you think the *names* of the elements in `n_observations` are defined?

See highlighted boxes

Describe how R accesses elements in a vector when you use *names* as the **index**.

Accesses elements with matching names

Prediction

Using the type coercion rules you inferred above, what will be the output of the following expressions?

Code example	Output
<code>n_observations["Porites"]</code>	
<code>coral_name <- "Pocillopora"</code> <code>n_observations[coral_name]</code>	

Indexing with logic

Inference

Code example	Output
<code>coral_cover_pct >= 25</code>	<pre>[1] TRUE FALSE FALSE FALSE FALSE [6] FALSE TRUE FALSE FALSE</pre>
<code>coral_cover_pct[coral_cover_pct >= 25]</code>	<pre>[1] 34.53 30.35</pre>
<code>survey_years[coral_cover_pct >= 25]</code>	<pre>[1] 2008 2020</pre>

Describe how R accesses elements in a vector when you use *logic* as the **index**.

Keeps elements where the index is TRUE

Prediction

Using the type coercion rules you inferred above, what will be the output of the following expressions?

Code example	Output
<pre>cyclone_year <- 2010 coral_cover_pct[survey_years == cyclone_year]</pre>	
<pre>risk_threshold <- 10 coral_cover_pct[coral_cover_pct <= risk_threshold]</pre>	