Comprehension Check due May 18, 2021 14:59 +03

## Introduction: Questions 1-3

Load the following web page, which contains information about Major League Baseball payrolls, into R: https://web.archive.org/web/20181024132313/http://www.stevetheump.com/Payrolls.htm

```
library(rvest)
url <- "https://web.archive.org/web/20181024132313/http://w
h <- read_html(url)
```

We learned that tables in html are associated with the `table` node. Use the `html_nodes()` function and the `table` node type to extract the first table. Store it in an object `nodes`:

```
nodes <- html_nodes(h, "table")
```

The `html_nodes()` function returns a list of objects of class `xml_node`. We can see the content of each one using, for example, the `html_text()` function. You can see the content for an arbitrarily picked component like this:

```
html_text(nodes[[8]])
```

If the content of this object is an html table, we can use the `html_table()` function to convert it to a data frame:

```
html_table(nodes[[8]])
```

You will analyze the tables from this HTML page over questions 1-3.

---

## Question 1

2.5/2.5 points (graded)

Many tables on this page are team payroll tables, with columns for rank, team, and one or more money values.

Convert the first four tables in `nodes` to data frames and inspect them.

Which of the first four `nodes` are tables of team payroll?
Check all correct answers. Look at table content, not column names.

- [ ] None of the above
- [ ] Table 1
- [x] Table 2
- [x] Table 3
- [x] Table 4

✔

**Answer code**

```
sapply(nodes[1:4], html_table)    # 2, 3, 4 give tables with payroll info
```

Submit You have used 1 of 2 attempts

Answers are displayed within the problem

## Question 2

2/2 points (graded)
For the last 3 components of `nodes`, which of the following are true? (Check all correct answers.)
Check all correct answers.

- [x] All three entries are tables.

- [ ] All three entries are tables of payroll per team.
- [x] The last entry shows the average across all teams through time, not payroll per team.
- [ ] None of the three entries are tables of payroll per team.

✔

**Answer code**

```
            html_table(nodes[[length(nodes)-2]])
html_table(nodes[[length(nodes)-1]])
html_table(nodes[[length(nodes)]])
```

Submit

You have used 1 of 2 attempts

Answers are displayed within the problem

## Question 3

1/1 point (graded)

Create a table called `tab_1` using entry 10 of `nodes`. Create a table called `tab_2` using entry 19 of `nodes`.

Note that the column names should be `c("Team", "Payroll", "Average")`. You can see that these column names are actually in the first data row of each table, and that `tab_1` has an extra first column `No.` that should be removed so that the column names for both tables match.

Remove the extra column in `tab_1`, remove the first row of each dataset, and change the column names for each table to `c("Team", "Payroll", "Average")`. Use a `full_join()` by the `Team` to combine these two tables.

How many rows are in the joined data table?

58

✔ **Answer:** 58

58

**Answer code**

```
           tab_1 <- html_table(nodes[[10]])
tab_2 <- html_table(nodes[[19]])
col_names <- c("Team", "Payroll", "Average")
tab_1 <- tab_1[-1, -1]
tab_2 <- tab_2[-1,]
names(tab_2) <- col_names
names(tab_1) <- col_names
full_join(tab_1,tab_2, by = "Team")
```

Submit

You have used 1 of 10 attempts

Answers are displayed within the problem

**Introduction: Questions 4 and 5**

The Wikipedia page on opinion polling for the Brexit referendum, in which the United Kingdom voted to leave the European Union in June 2016, contains several tables. One table contains the results of all polls regarding the referendum over 2016:

**2016** [ edit ]

| Date(s) conducted | Remain | Leave | Undecided | Lead | Sample | Conducted by | Polling type | Notes |
|---|---|---|---|---|---|---|---|---|
| *23 June 2016* | *48.1%* | ***51.9%*** | *N/A* | *3.8%* | *33,577,342* | *Results of the United Kingdom European Union membership referendum, 2016* | *UK-wide referendum* | |
| 23 June | **52%** | 48% | N/A | 4% | 4,772 | YouGov | Online | On the day poll |
| 22 June | **55%** | 45% | N/A | 10% | 4,700 | Populus | Online | |
| 20–22 June | **51%** | 49% | N/A | 2% | 3,766 | YouGov | Online | Includes Northern Ireland (turnout weighted) |
| 20–22 June | **49%** | 46% | 1% | 3% | 1,592 | Ipsos MORI | Telephone | |
| 20–22 June | 44% | **45%** | 9% | 1% | 3,011 | Opinium | Online | |
| 17–22 June | **54%** | 46% | N/A | 8% | 1,032 | ComRes | Telephone | Those expressing a voting intention (turnout weighted) |
| | **48%** | 42% | 11% | 6% | | | | All UK adults (turnout weighted) |
| 16–22 June | 41% | **43%** | 16% | 2% | 2,320 | TNS | Online | |
| 20 June | **45%** | 44% | 11% | 1% | 1,003 | Survation/IG Group | Telephone | |
| 18–19 June | 42% | **44%** | 13% | 2% | 1,652 | YouGov | Online | |
| 16–19 June | **53%** | 46% | 2% | 7% | 800 | ORB/Telegraph | Telephone | Definite voters only |
| 17–18 June | **45%** | 42% | 13% | 3% | 1,004 | Survation | Telephone | |

Use the **rvest** library to read the HTML from this Wikipedia page (make sure to copy both lines of the URL):

```
library(rvest)
library(tidyverse)
url <- "https://en.wikipedia.org/w/index.php?title=Opinion_
```

---

## Question 4

1/1 point (graded)

Assign `tab` to be the html nodes of the "table" class.

How many tables are in this Wikipedia page?

39 ✔ **Answer:** 39

39

### Answer code

```
tab <- read_html(url) %>% html_nodes("table")
length(tab)
```

Submit You have used 1 of 10 attempts

Answers are displayed within the problem

## Question 5

1/1 point (graded)

Inspect the first several html tables using `html_table()` with the argument `fill=TRUE` (you can read about this argument in the documentation). Find the first table that has 9 columns with the first column named "Date(s) conducted".

What is the first table number to have 9 columns where the first column is named "Date(s) conducted"?

5

✔ **Answer:** 5

5

**Answer code**

Inspect the column names of table 5 with this code (you can substitute other integers for 5 to confirm this is correct):

```
tab[[5]] %>% html_table(fill = TRUE) %>% names()    # inspect column names
```

Submit You have used 1 of 10 attempts

Answers are displayed within the problem