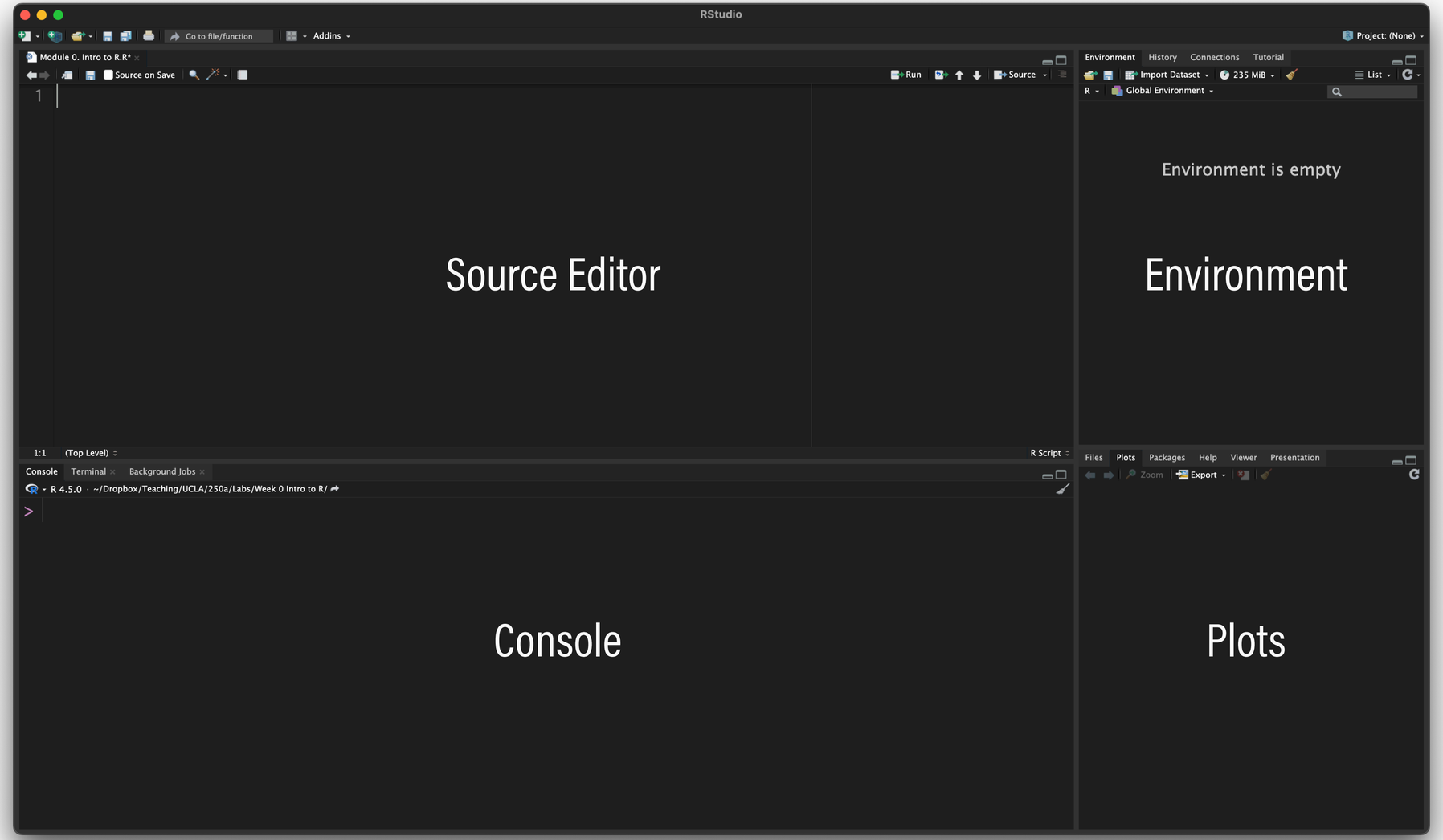
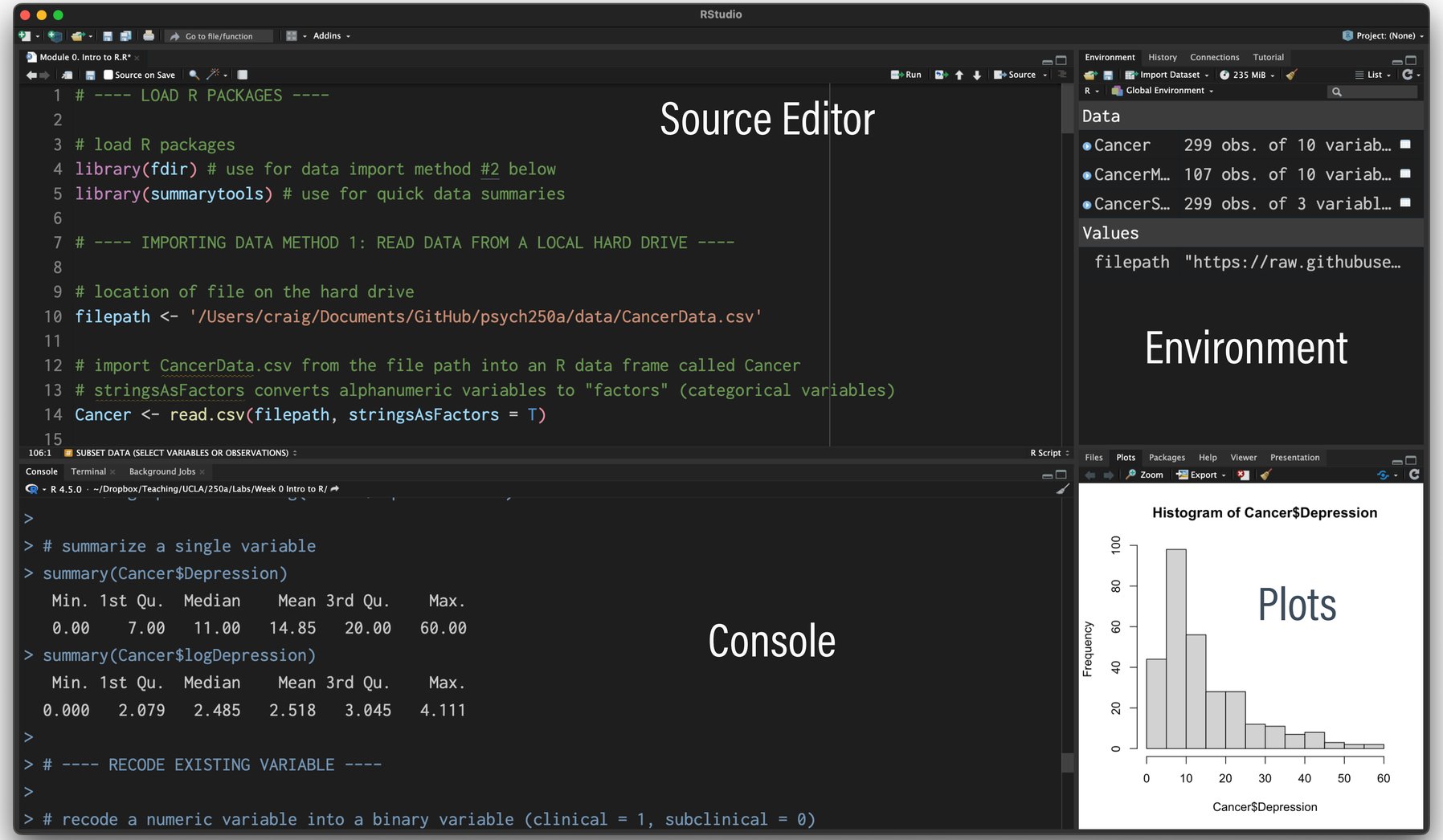
LAB WEEK 0

INTRODUCTION TO R

RSTUDIO INTERFACE



RSTUDIO INTERFACE

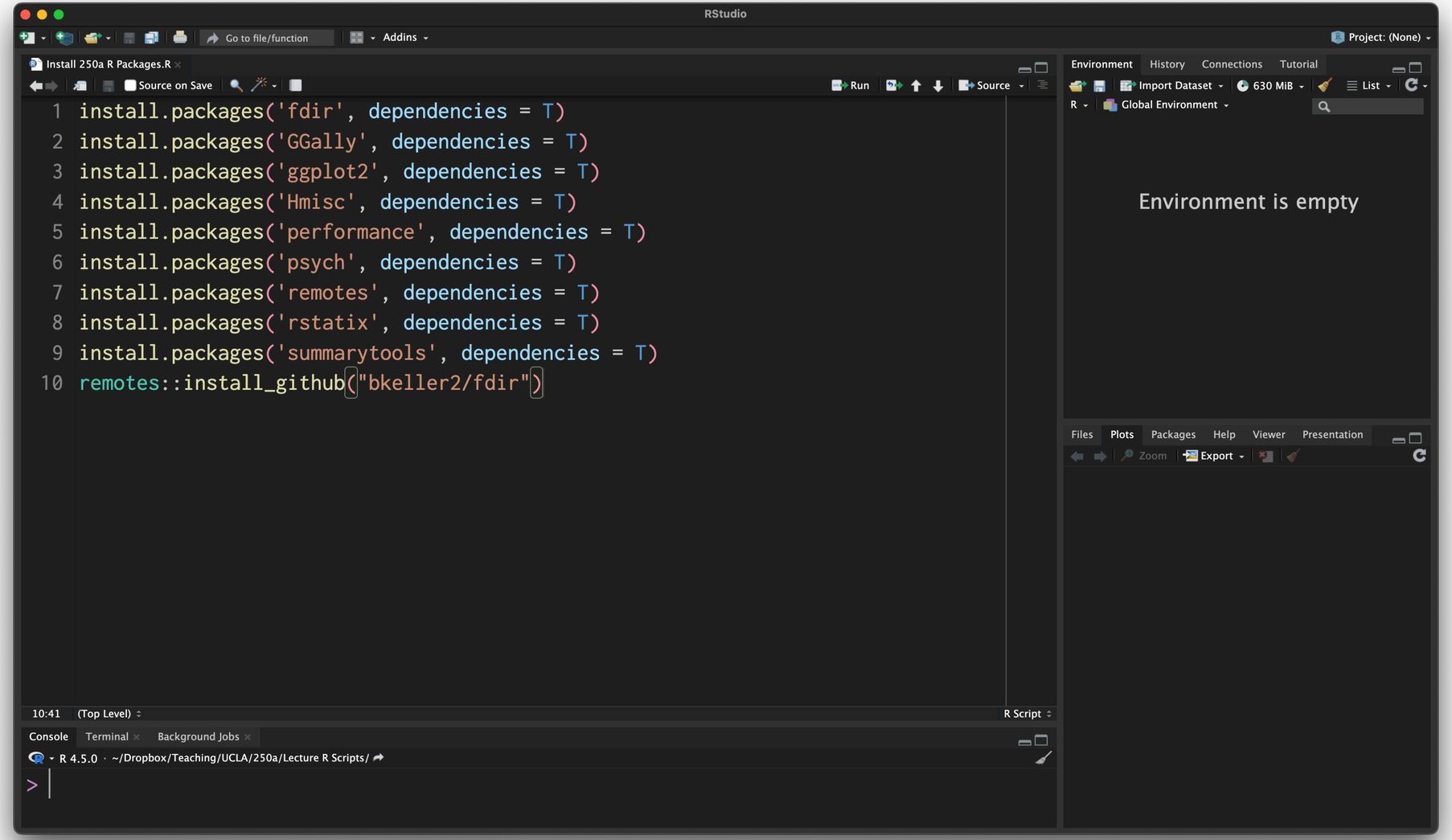


INSTALLING PACKAGES

- A package is a collection of functions bundled together to extend R's capabilities beyond its basic capabilities. Packages need to be installed once (and usually again whenever you update to a newer version of R). Open "Install 250a R Packages.R" to run the code below.

```
install.packages('fdir', dependencies = T)
install.packages('GGally', dependencies = T)
install.packages('ggplot2', dependencies = T)
install.packages('Hmisc', dependencies = T)
install.packages('performance', dependencies = T)
install.packages('psych', dependencies = T)
install.packages('remotes', dependencies = T)
install.packages('rstatix', dependencies = T)
install.packages('summarytools', dependencies = T)
remotes::install_github("bkeller2/fdir")
```

RSTUDIO INSTALLATION SCRIPT



R BASICS: ASSIGNMENT OPERATOR

- The assignment operator <- defines a quantity (or object) on the left by an expression or quantity on the right

```
# ASSIGNMENT OPERATOR ----

# define a new object called Constant that equals 1
Constant <- 1

# copy a data frame called DataSet1 into a new data frame called DataSet2
DataSet2 <- DataSet1
```

R BASICS: VECTORS

- A vector c(...) is a container that can hold a set of values of the same type (e.g., all numeric, all alphanumeric)

```
# VECTOR OF VALUES ----

# define a new object called vars2analyze with the names of three variables
vars2analyze <- c('Depression', 'Anxiety', 'Interference')

# define a new object called axis_limits with two numeric values
axis_limits <- c(0,10)
```

R BASICS: COMMENT LINES

- The # is a special comment symbol that tells R to ignore any text that follows on the same line (e.g., used to insert notes explaining code)

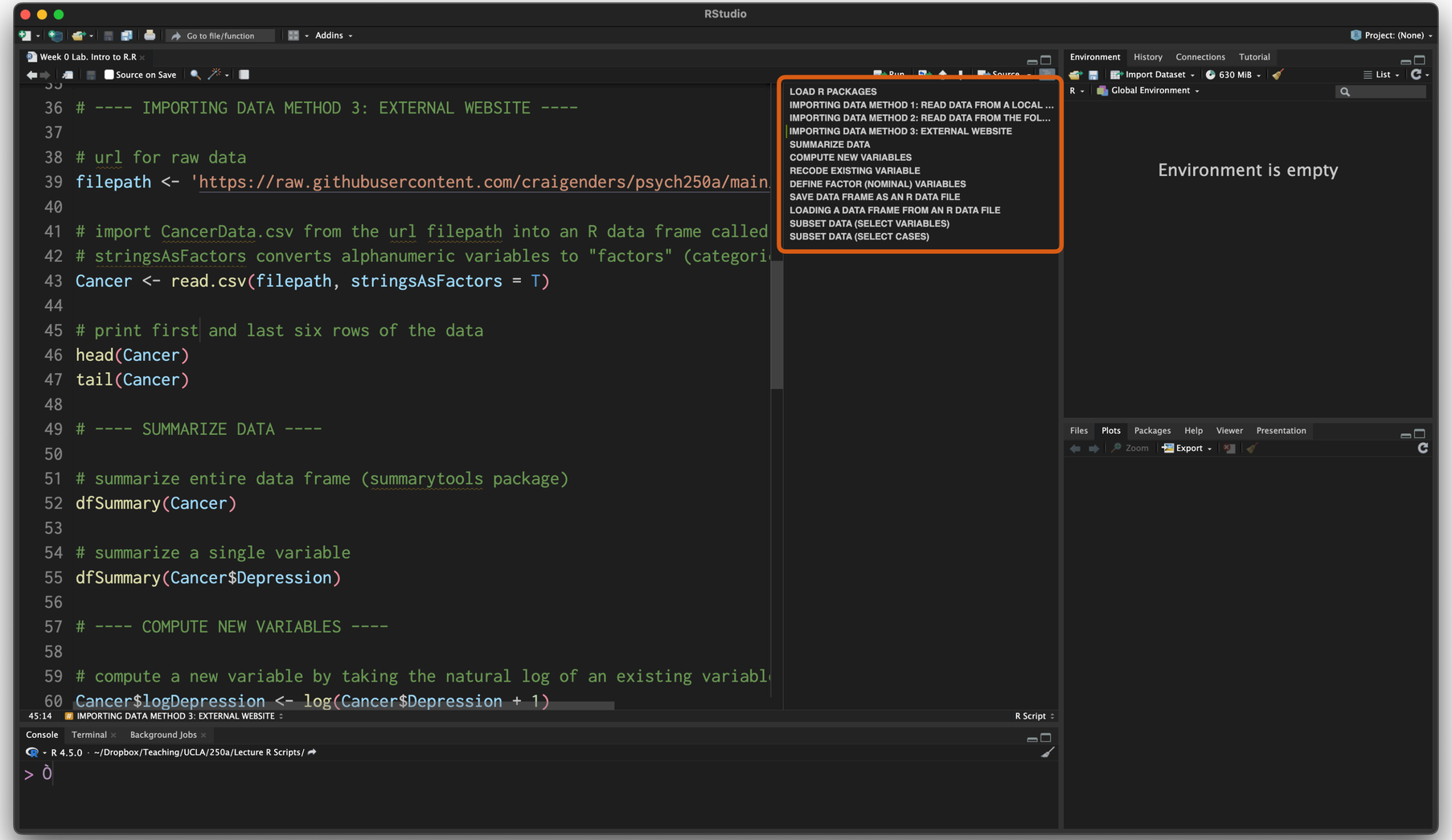
```
# this is a comment statement that will be ignored

# define a new object called Constant that equals 1
Constant <- 1

# copy a data frame called DataSet1 into a new data frame called DataSet2
DataSet2 <- DataSet1
```

R BASICS: SECTION HEADERS

- The string # TEXT ---- is called a section header or code section marker, and it's used in R scripts to help organize your code (RStudio displays the headers in a navigation pane).

```
# DEMO ASSIGNMENT OPERATOR ----

# define a new object called Constant that equals 1
Constant <- 1

# DEMO DATA SET COPY ----

# copy a data frame called DataSet1 into a new data frame called DataSet2
DataSet2 <- DataSet1
```

RSTUDIO INTERFACE



UVEAL MELANOMA AND DEPRESSION

Uveal melanoma, a rare eye cancer, presents potential vision loss and life threat. This prospective, longitudinal study interrogated the predictive utility of visual impairment, as moderated by optimism/pessimism, on depressive symptoms in 299 adults undergoing diagnostic evaluation.

Annette Stanton

James MacDonald

MacDonald, J.J., Jorge-Miller, A., Enders, C.K., McCannel, T., Beran, T., & Stanton, A.L. (2021). Perceived and objective visual impairment predicting depressive symptoms across one year in uveal melanoma diagnostic biopsy: Optimism and pessimism as moderators. *Health Psychology*, *40*, 408-417.

PRELIMINARIES: LOADING PACKAGES

- A package is a collection of functions bundled together to extend R's capabilities beyond its basic capabilities. The library function loads

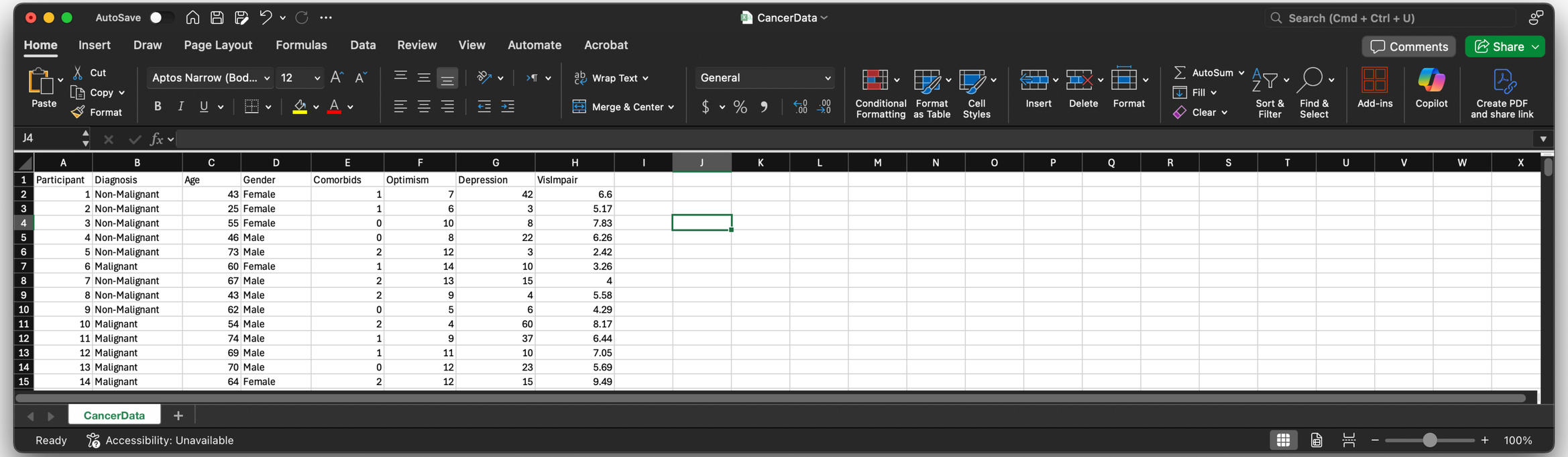
```
# LOAD R PACKAGES ----

# load R packages
library(fdir)  # use for data import method #2
library(summarytools)  # use for data summaries
```

.CSV DATA FILES

- .csv is a plain text format with variables separated by commas, often with the variable names in the first row

OPENING .CSV FILES

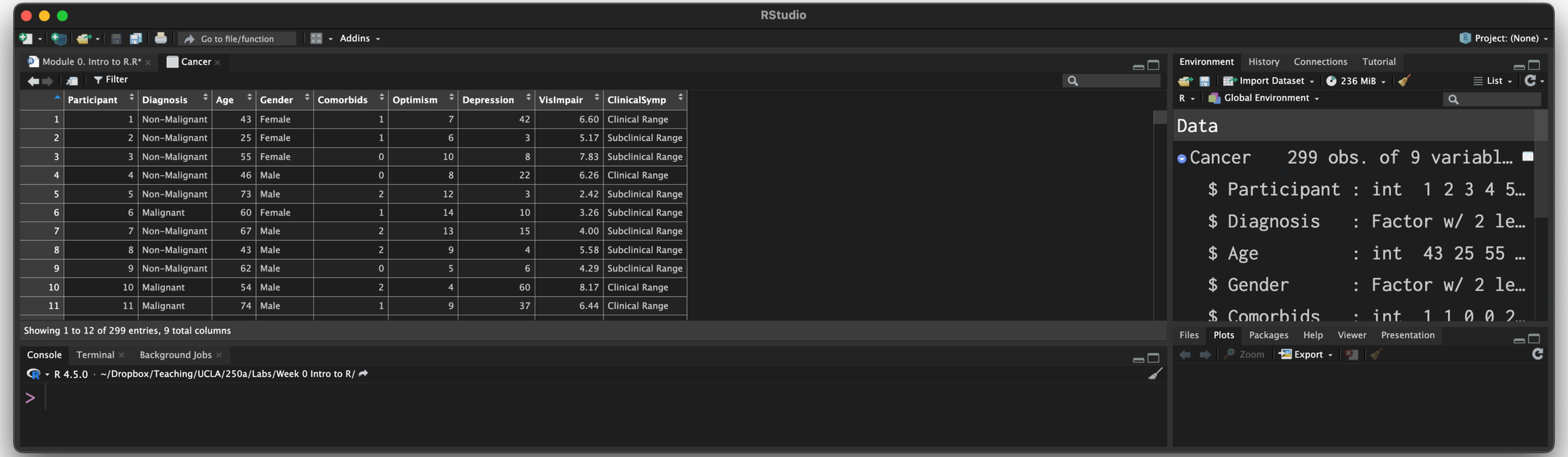
- Opening a .csv file parses the comma-separated values into columns of variables, often with one row of data per person



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
1	Participant	Diagnosis	Age	Gender	Comorbids	Optimism	Depression	VisImpair																
2	1	Non-Malignant	43	Female	1	7	42	6.6																
3	2	Non-Malignant	25	Female	1	6	3	5.17																
4	3	Non-Malignant	55	Female	0	10	8	7.83																
5	4	Non-Malignant	46	Male	0	8	22	6.26																
6	5	Non-Malignant	73	Male	2	12	3	2.42																
7	6	Malignant	60	Female	1	14	10	3.26																
8	7	Non-Malignant	67	Male	2	13	15	4																
9	8	Non-Malignant	43	Male	2	9	4	5.58																
10	9	Non-Malignant	62	Male	0	5	6	4.29																
11	10	Malignant	54	Male	2	4	60	8.17																
12	11	Malignant	74	Male	1	9	37	6.44																
13	12	Malignant	69	Male	1	11	10	7.05																
14	13	Malignant	70	Male	0	12	23	5.69																
15	14	Malignant	64	Female	2	12	15	9.49																

R DATA FRAMES

- A data frame in R is a special container that stores the data imported from a .csv file, arranging it into rows (cases) and columns (variables)



IMPORTING DATA

= data frame name
= variable name
= raw data file name

- The lab R script demonstrates three ways to import raw data from a .csv file—we will primarily read data from an external website

```
# IMPORTING DATA METHOD 3: EXTERNAL WEBSITE ----

# url for raw data
filepath <-
  'https://raw.githubusercontent.com/craigenders/psych250a/main/data/CancerData.csv'

# import CancerData.csv from the url filepath into an R data frame called Cancer
# stringsAsFactors converts alphanumeric variables to "factors" (categorical variables)
Cancer <- read.csv(filepath, stringsAsFactors = T)

# print first and last six rows of the data
head(Cancer)
tail(Cancer)
```

R OUTPUT

```
> head(Cancer)
  Participant     Diagnosis Age Gender Comorbids Optimism Depression VisImpair
1           1 Non-Malignant  43 Female         1        7         42      6.60
2           2 Non-Malignant  25 Female         1        6          3      5.17
3           3 Non-Malignant  55 Female         0       10          8      7.83
4           4 Non-Malignant  46   Male         0        8         22      6.26
5           5 Non-Malignant  73   Male         2       12          3      2.42
6           6     Malignant  60 Female         1       14         10      3.26

> tail(Cancer)
    Participant     Diagnosis Age Gender Comorbids Optimism Depression VisImpair
294         294 Non-Malignant  44   Male         1        9          9      3.31
295         295 Non-Malignant  65 Female         1       10         10      2.59
296         296 Non-Malignant  58 Female         2        9         14      5.99
297         297 Non-Malignant  67   Male         1        7          6      2.42
298         298     Malignant  67   Male         0       11          7      2.65
299         299     Malignant  76 Female         2        9          8      3.71
```

SUMMARIZING DATA

= data frame name

= variable name

- Use the dfSummary function with every analysis to get numeric and visual summaries of a data frame's variables

```
# SUMMARIZE DATA ----

# summarize entire data frame (summarytools package)
dfSummary(Cancer)

# summarize a single variable
dfSummary(Cancer$Depression)
```

R OUTPUT

```
Data Frame Summary
Cancer
Dimensions: 299 x 1
Duplicates: 249

-------------------------------------------------------------------------------------------------------
No   Variable     Stats / Values             Freqs (% of Valid)   Graph                Valid      Missing
---- ------------ -------------------------- -------------------- ---------------- ---------- ---------
1    Depression   Mean (sd) : 14.9 (11.4)    50 distinct values     :                299        0
     [integer]    min < med < max:                                  :                (100.0%)   (0.0%)
                  0 < 11 < 60                                     : : .
                  IQR (CV) : 13 (0.8)                             : : : .
                                                                  : : : : . . . .
-------------------------------------------------------------------------------------------------------
```

COMPUTING NEW VARIABLES

= data frame name
= variable name

- Variables are referenced by naming a data frame followed by a $ and the variables name (i.e., dataname$varname)

```
# COMPUTE NEW VARIABLES ----

# compute a new variable by taking the natural log of an existing variable
Cancer$logDepression <- log(Cancer$Depression + 1)

# summarize a single variable
dfSummary(Cancer$logDepression)
```

R OUTPUT

```
Data Frame Summary
Cancer
Dimensions: 299 x 1
Duplicates: 249

-------------------------------------------------------------------------------------------------------
No   Variable        Stats / Values          Freqs (% of Valid)   Graph              Valid      Missing
---- -------------- ----------------------- -------------------- ------------------ ---------- ---------
1    logDepression   Mean (sd) : 2.5 (0.7)   50 distinct values           :         299        0
     [numeric]       min < med < max:                                     : .       (100.0%)   (0.0%)
                     0 < 2.5 < 4.1                                        : : :
                     IQR (CV) : 1 (0.3)                                 : : : : .
                                                                      : : : : : :
-------------------------------------------------------------------------------------------------------
```

RECODING EXISTING VARIABLES

= data frame name
= variable name

- R has numerous built in functions (e.g., ifelse) for recoding and modifying variables

```
# RECODE EXISTING VARIABLE ----

# recode a numeric variable into a binary variable (clinical = 1, subclinical = 0)
Cancer$ClinicalSymp <- ifelse(Cancer$Depression >= 16, 1, 0)

# summarize a single variable
dfSummary(Cancer$ClinicalSymp)
```

R OUTPUT

```
Data Frame Summary
Cancer
Dimensions: 299 x 1
Duplicates: 297

-------------------------------------------------------------------------------------------
No   Variable       Stats / Values   Freqs (% of Valid)   Graph              Valid      Missing
---- -------------- ---------------- -------------------- ------------------ ---------- ---------
1    ClinicalSymp   Min  : 0         0 : 198 (66.2%)      IIIIIIIIIIII       299        0
     [numeric]      Mean : 0.3       1 : 101 (33.8%)      IIIIII             (100.0%)   (0.0%)
                    Max  : 1
-------------------------------------------------------------------------------------------
```

DEFINE FACTOR (NOMINAL) VARIABLES

= data frame name

= variable name

- A factor is a special designation for a nominal or alphanumeric (i.e., categorical) variable

```
# DEFINE FACTOR (NOMINAL) VARIABLES ----

# recode a numeric variable into a binary variable (clinical = 1, subclinical = 0)
Cancer$ClinicalSymp <- factor(
  Cancer$ClinicalSymp,
  levels = c(0, 1),
  labels = c('Subclinical Range', 'Clinical Range')
)

# summarize a single variable
dfSummary(Cancer$ClinicalSymp)
```

R OUTPUT

Data Frame Summary
Cancer
Dimensions: 299 x 1
Duplicates: 297

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Valid	Missing
1	ClinicalSymp [factor]	1. Subclinical Range 2. Clinical Range	198 (66.2%) 101 (33.8%)	IIIIIIIIIIII IIIIII	299 (100.0%)	0 (0.0%)

SAVING AN R DATA FRAME

= data frame name
= variable name
= R data file name

- An updated data frame containing the new variables and modifications can be saved as an R-formatted file (.RData)

```
# SAVE DATA FRAME AS AN R DATA FILE  ----

# save Cancer data frame to the desktop
save(Cancer, file = '~/Desktop/Cancer.RData')
```

LOADING AN R DATA FRAME

= data frame name
= variable name
= R data file name

- Loading an .RData file restores the saved data frame into your R session using its original name, which may be different from the .RData file name itself

```
# LOAD R DATA FRAME  ----

# load Cancer data frame from the desktop
load('~/Desktop/Cancer.RData')
```

SUBSETTING: SELECTING VARIABLES

= data frame name
= variable name

- Elements of a data frame are accessed with the format DataFrame[row indices, column indices], where the first position specifies which rows to select and the second position specifies which columns to select

```
# SUBSET DATA (SELECT VARIABLES) ----

# create new data frame with a subset of variables from the original
# a subset of variable names appears in the column index after the comma
CancerSubset <- Cancer[,c('Optimism','Depression','VisImpair')]

# print dimensions of original and subset data
dim(Cancer)
dim(CancerSubset)
```

R OUTPUT

```
> dim(Cancer)
[1] 299  10

> dim(CancerSubset)
[1] 299   3
```

SUBSETTING: SELECTING CASES

= data frame name
= variable name

- Elements of a data frame are accessed with the format DataFrame[row indices, column indices], where the first position specifies which rows to select and the second position specifies which columns to select

```
# SUBSET DATA (SELECT CASES) ----

# create new data frame with only participants with Diagnosis = Malignant
# a selection criterion appears in the row index before the comma
CancerMalig <- Cancer[Cancer$Diagnosis == 'Malignant',]

# print dimensions of original and subset data
dim(Cancer)
dim(CancerMalig)
```

R OUTPUT

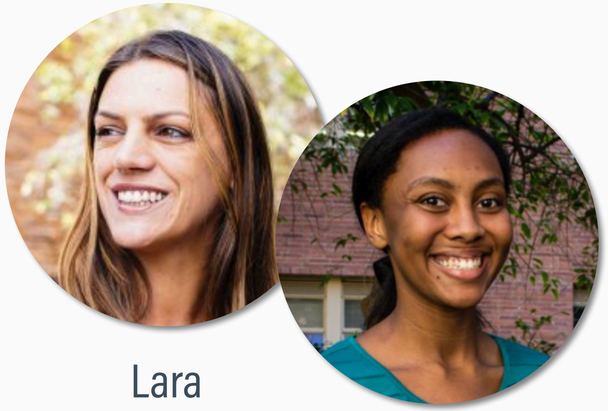
```
> dim(Cancer)
[1] 299  10

> dim(CancerMalig)
[1] 107  10
```

SMALL GROUP EXERCISE

Download two files from Bruin Learn: “Week 0 Lab. R Intro.R” and “Week 0 Small Group Exercise.R”. The Lab script contains the R code we just discussed. The Exercise script contains only the URL for a different data set, ClinicalTrialData.csv. In groups of two or three, you will complete a series of R tasks that provide practice for the next assignment. There is no need to write code from scratch; instead, you can copy and paste code chunks from the Lab file into your Exercise script, modifying the data and variable names as needed. The ClinicalTrialData.csv file for this exercise contains data from a clinical trial investigating the impact of two medication regimens on smoking and drinking behavior.

SMOKING AND DRINKING CESSATION TRIAL

Pharmacological treatments that can concomitantly address cigarette smoking and heavy drinking stand to improve health care delivery for these highly prevalent co-occurring conditions. This superiority trial compared the combination of varenicline and naltrexone against varenicline alone for smoking cessation and drinking reduction among heavy-drinking smokers.



Lara Ray

ReJoyce Green

Ray, L.A., Green, R., Enders, C., et al. (2021). Efficacy of combining varenicline and naltrexone for smoking cessation and drinking reduction: A randomized clinical trial. *American Journal of Psychiatry*, *178*, 818–828.

SMALL GROUP EXERCISE TASK 1

- Use the provided URL to import the ClinicalTrialData.csv file into an R data frame (import method #3 from the Week 0 lab script).
- Print the first six rows of the data frame.

SMALL GROUP EXERCISE TASK 2

- Use the dfSummary function to get numeric and visual summaries of the data frame's variables.

SMALL GROUP EXERCISE TASK 3

- Add a new variable to the data frame called DrinksPerDay that captures each person's average number of drinks per day. Compute the variable by dividing the DrinksWeek8 variable by seven.

- Use the dfSummary function to a get numeric and visual summary of just the new variable.

SMALL GROUP EXERCISE TASK 4

- Add a new factor variable to the data frame called DailyDrinker that captures whether somebody drank at least one drink per day, on average. First, create a new binary variable that equals 1 if DrinksPerDay >= 1 and 0 otherwise. Second, convert the binary variable to a factor and assign labels to each category.

- Use the dfSummary function to a get numeric and visual summary of just the new variable.

SMALL GROUP EXERCISE TASK 5

- Save the data frame to the desktop as an R data file called Lab0.RData. The file path is most likely “~/Desktop/Lab0.RData”.
- Shut down and restart RStudio. Use the load command to load the data frame back into memory.