Basic skills in R: part II

Learning objectives

- **Importing data from excel into the R environment**

- **Basic data management**
  - Management of numerical variables
  - Manipulation of categorical variables

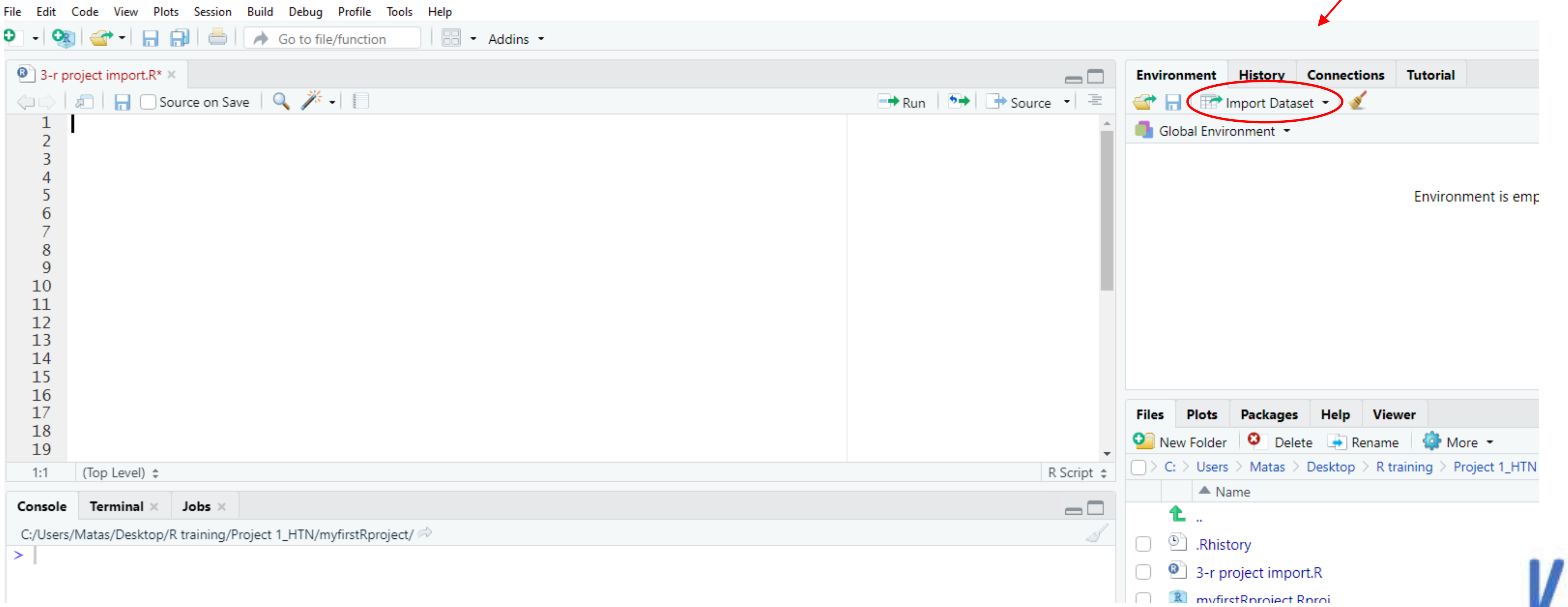
Scenario

- Imagine that your colleague sends you a patient dataset in excel format and asks you to perform exploratory data analysis.

- Let’s go through how to upload a file in RStudio

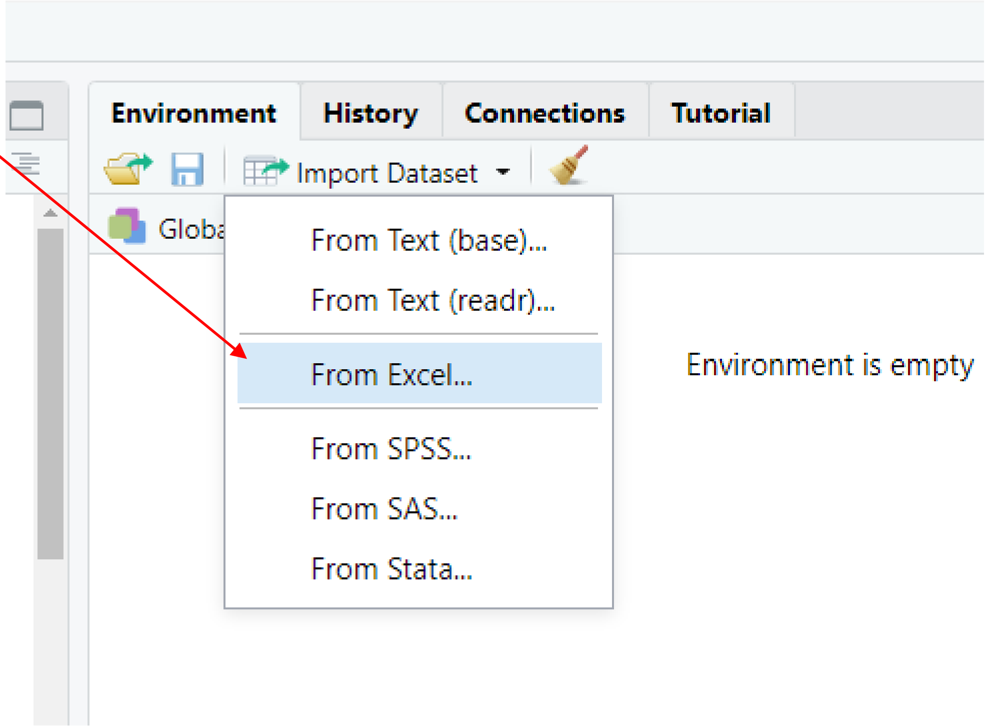
How to upload an excel file in R

The easiest way is to click “import dataset”

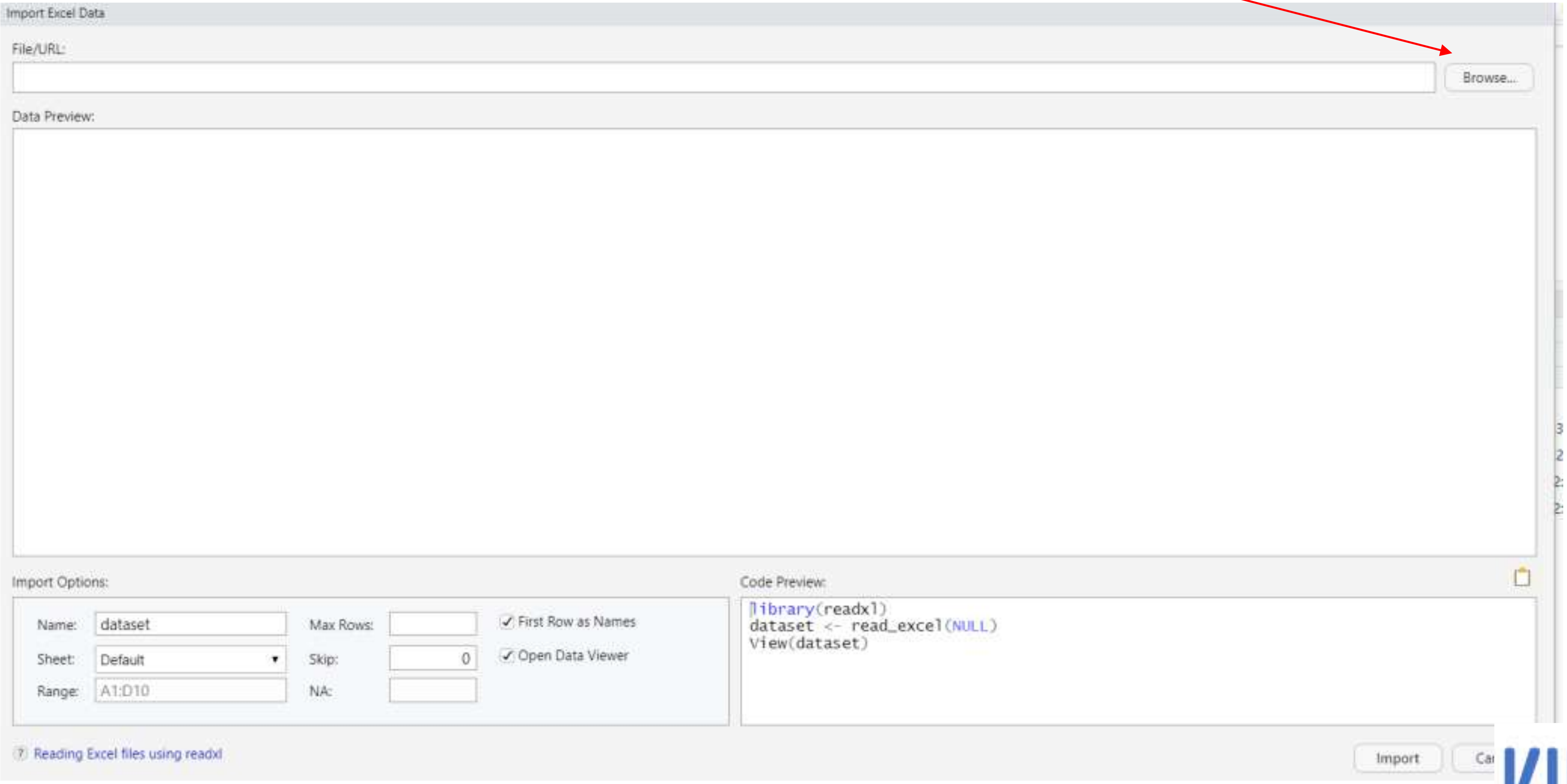


```
C:/Users/Matas/Desktop/R training/Project 1_HTN/myfirstRproject/
>
```

Select “from Excel”



Select “browse” and search for your excel file. Make sure it’s saved on your computer!



Name: dataset

Sheet: Default

Range: A1:D10

Max Rows:

Skip: 0

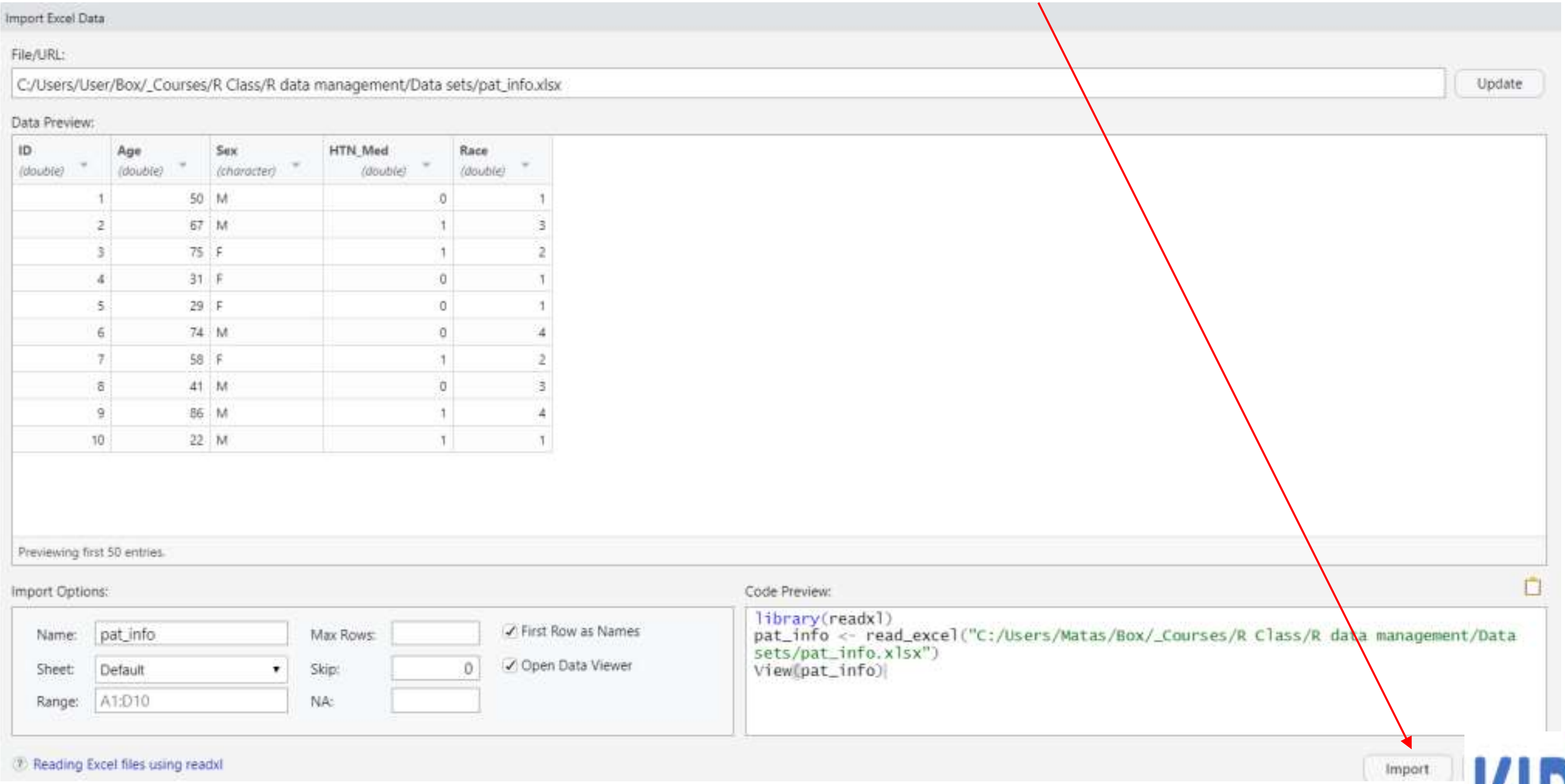
NA:

☑ First Row as Names

☑ Open Data Viewer

```
library(readxl)
dataset <- read_excel(NULL)
View(dataset)
```

Once you select your excel file, you will see the preview of the data. Next, select “import”.



Import Excel Data

File/URL:

C:/Users/User/Box/_Courses/R Class/R data management/Data sets/pat_info.xlsx

Data Preview:

ID (double)	Age (double)	Sex (character)	HTN_Med (double)	Race (double)
1	50	M	0	1
2	67	M	1	3
3	75	F	1	2
4	31	F	0	1
5	29	F	0	1
6	74	M	0	4
7	58	F	1	2
8	41	M	0	3
9	86	M	1	4
10	22	M	1	1

Previewing first 50 entries.

Import Options:

Name: pat_info

Sheet: Default

Range: A1:D10

Max Rows:

Skip: 0

NA:

☑ First Row as Names

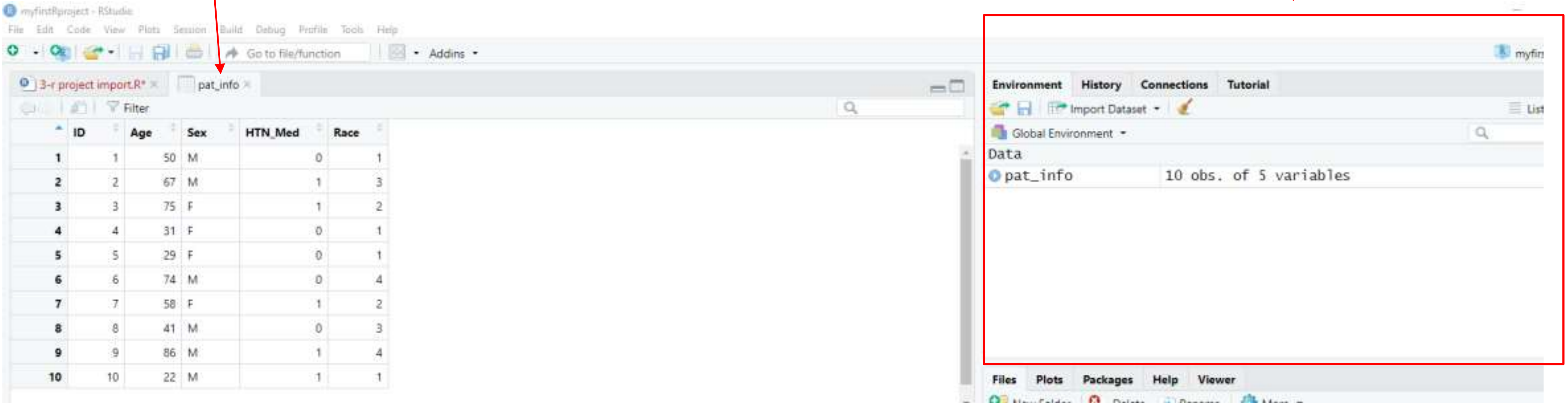
☑ Open Data Viewer

Code Preview:

```
library(readxl)
pat_info <- read_excel("C:/Users/Matas/Box/_Courses/R Class/R data management/Data
sets/pat_info.xlsx")
View(pat_info)
```

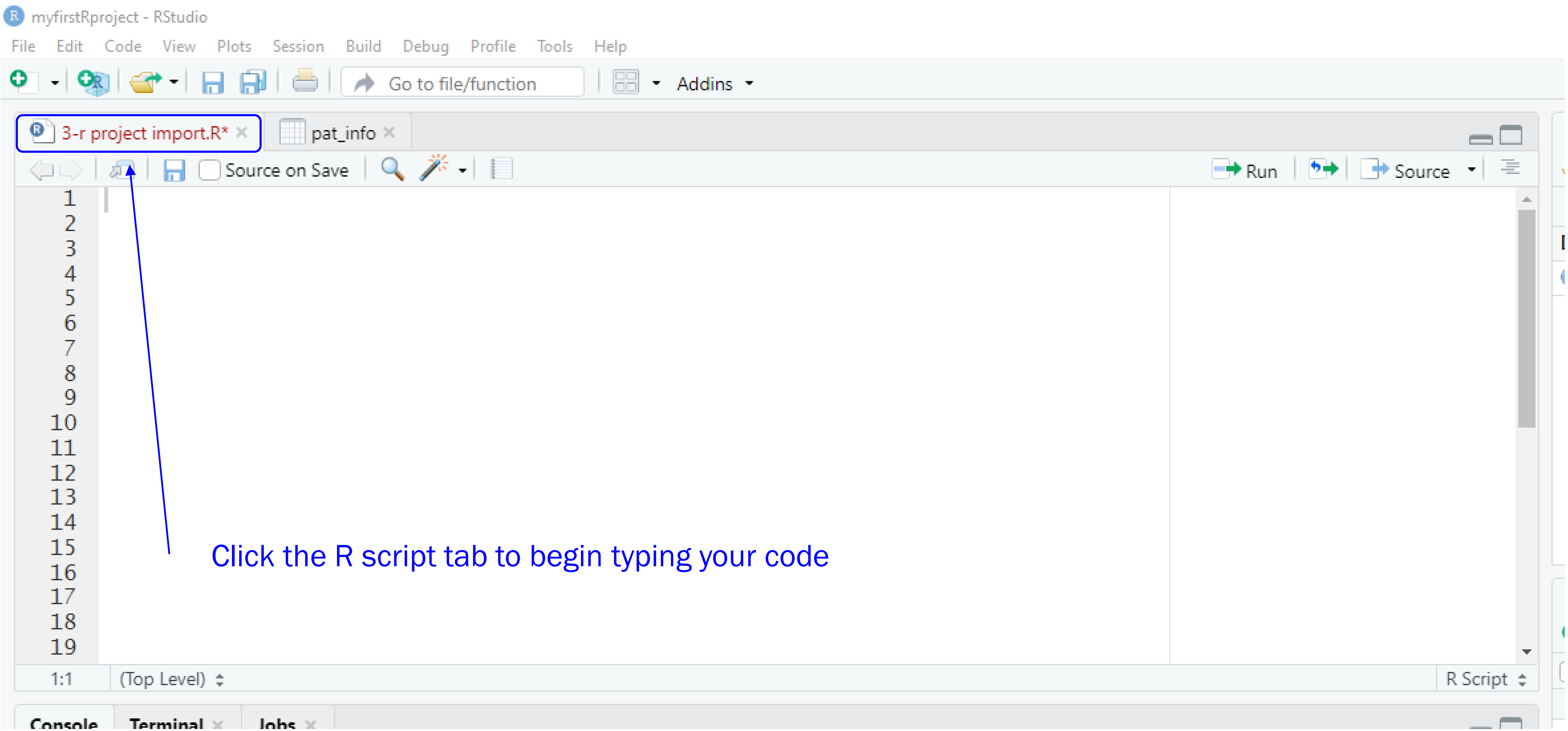
You see a new tab that has opened your dataset in R.

The dataset is officially in the R environment as shown here. Name of dataset is “pat_info”


myfirstRproject - RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
3-r project import.R*
pat_info
Filter
Environment History Connections Tutorial
Import Dataset
List
Global Environment
myfirs
Files Plots Packages Help Viewer

	ID	Age	Sex	HTN_Med	Race
1	1	50	M	0	1
2	2	67	M	1	3
3	3	75	F	1	2
4	4	31	F	0	1
5	5	29	F	0	1
6	6	74	M	0	4
7	7	58	F	1	2
8	8	41	M	0	3
9	9	86	M	1	4
10	10	22	M	1	1

```
Data
pat_info          10 obs. of 5 variables
```



Click the R script tab to begin typing your code

Basic data exploration

NUMERICAL VARIABLES

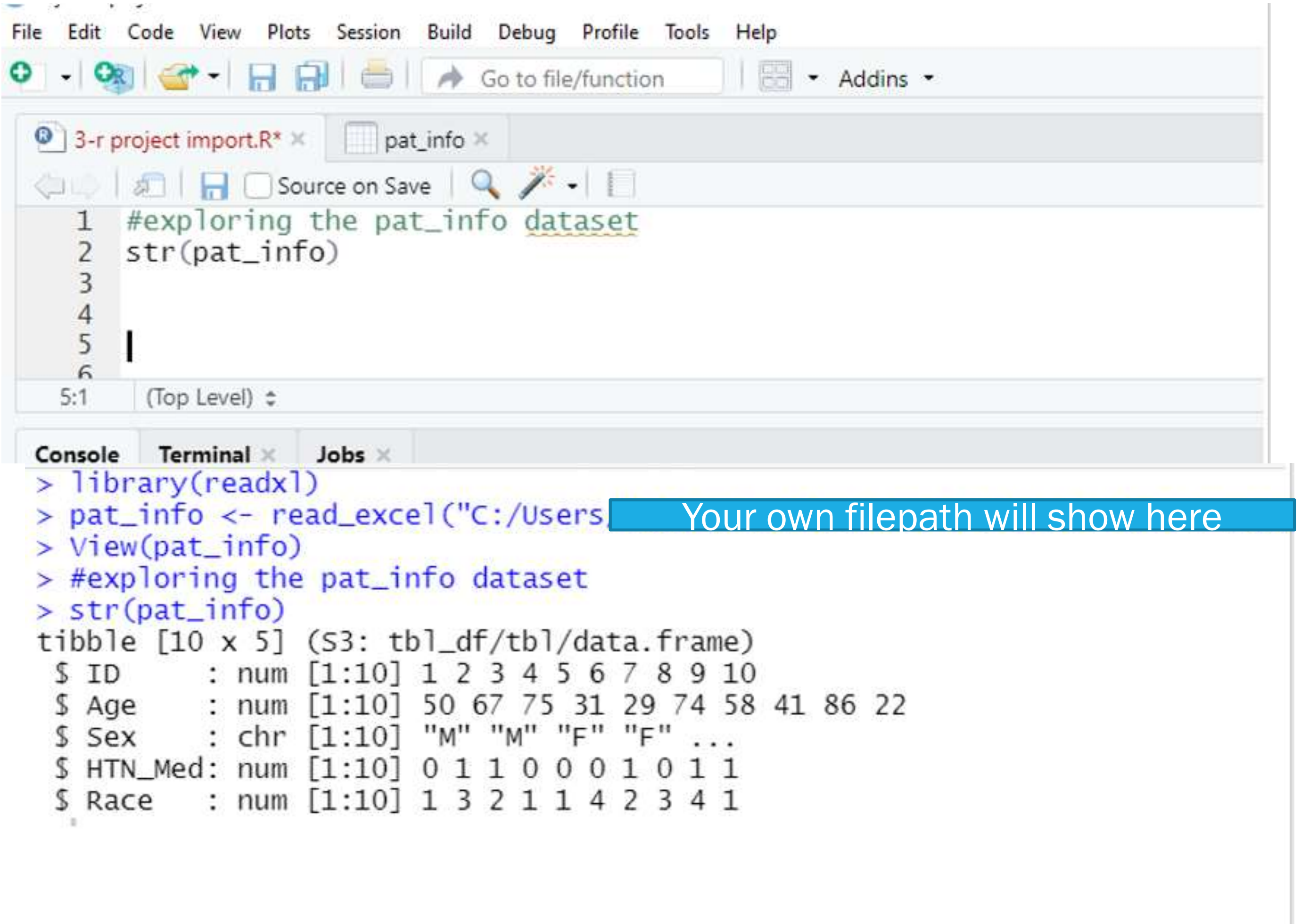
Exploring the dataset: str function

str function: displays the structure of a R object (in this case a data frame).

Output is displayed here

Note: everything in R is case-sensitive

Down here we see that the data frame weights contains 5 variables: 4 numerical and 1 character



```
#exploring the pat_info dataset
str(pat_info)
```

```
> library(readxl)
> pat_info <- read_excel("C:/Users
> View(pat_info)
> #exploring the pat_info dataset
> str(pat_info)
tibble [10 x 5] (S3: tbl_df/tbl/data.frame)
 $ ID     : num [1:10] 1 2 3 4 5 6 7 8 9 10
 $ Age    : num [1:10] 50 67 75 31 29 74 58 41 86 22
 $ Sex    : chr [1:10] "M" "M" "F" "F" ...
 $ HTN_Med: num [1:10] 0 1 1 0 0 0 1 0 1 1
 $ Race   : num [1:10] 1 3 2 1 1 4 2 3 4 1
```

Your own filepath will show here

The head function in R environment displays the first observations of a dataframe or variable. By specifying the “n” option, you control how many observations will be displayed.

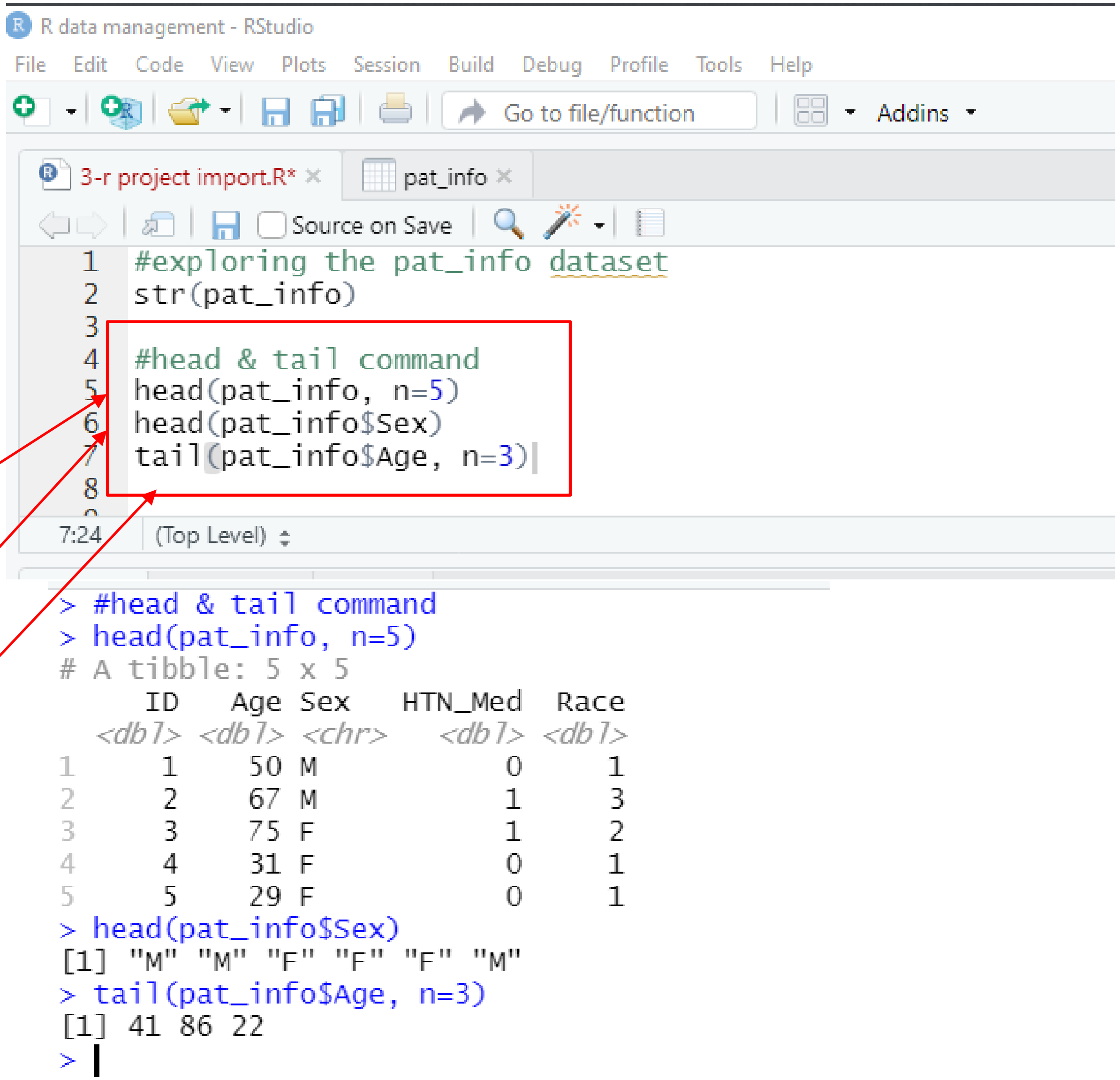
Note: If you don’t specify the n option, the first six observations will be displayed by default.

The tail function in R, provides the last obseverations. Similarly, the last 6 are the default unless specified by the “n”.

Here, I’m requesting the first five observations from the entire dataset pat_info

Next, I specify that I want to display the first default observations for the variable Sex in the pat_info dataset

Finally, I am requesting the last three observations of the variable Age.



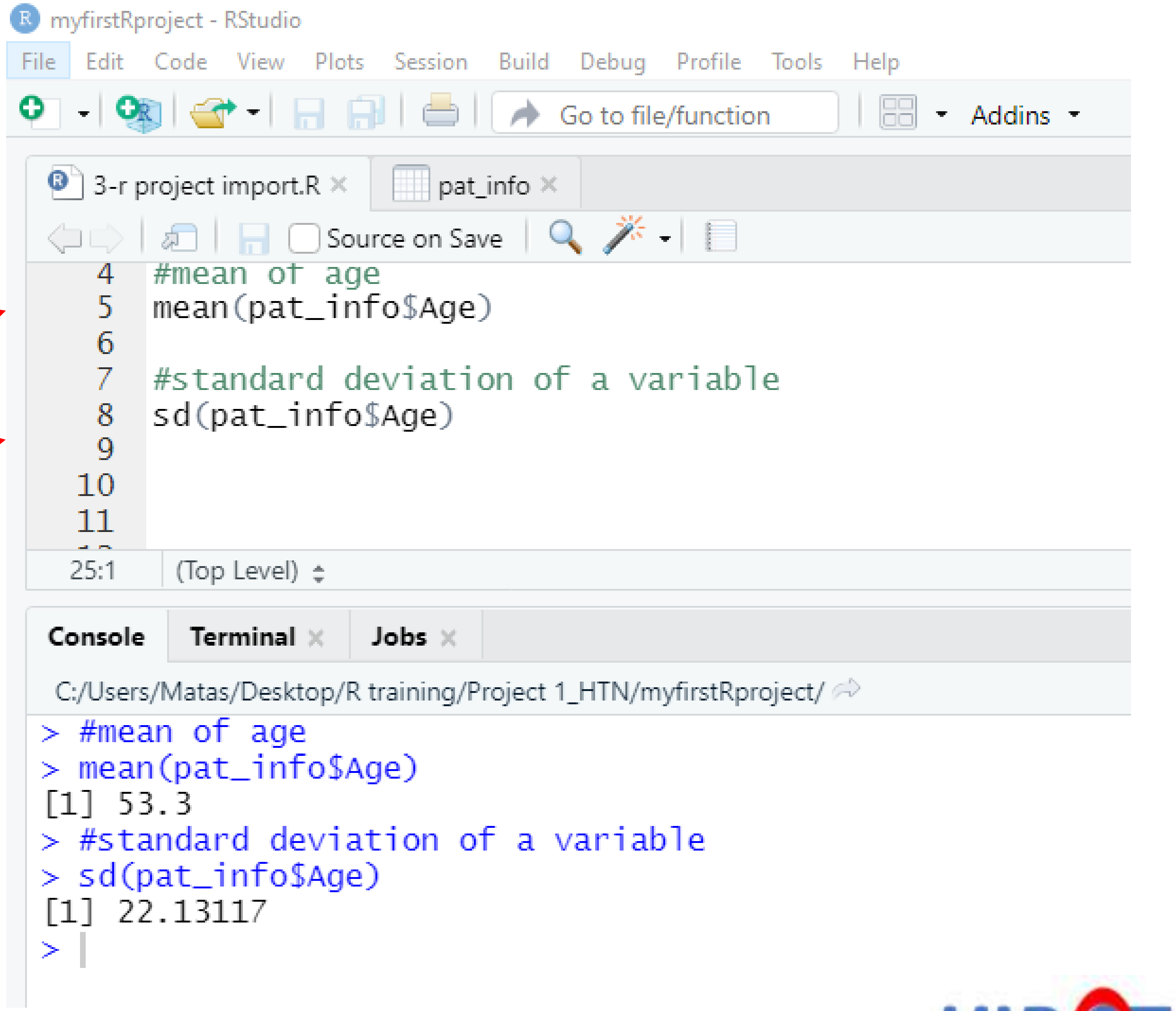
```
#exploring the pat_info dataset
str(pat_info)

#head & tail command
head(pat_info, n=5)
head(pat_info$Sex)
tail(pat_info$Age, n=3)
```

```
> #head & tail command
> head(pat_info, n=5)
# A tibble: 5 x 5
     ID   Age Sex   HTN_Med  Race
  <dbl> <dbl> <chr>   <dbl> <dbl>
1     1    50 M           0     1
2     2    67 M           1     3
3     3    75 F           1     2
4     4    31 F           0     1
5     5    29 F           0     1
> head(pat_info$Sex)
[1] "M" "M" "F" "F" "F" "M"
> tail(pat_info$Age, n=3)
[1] 41 86 22
> 
```

Identifying the mean & standard deviation of a variable

Type:
Mean(pat_info$Age)
sd(pat_info$Age)

The first part represents the name of the data frame and the second part after the dollar sign represents the specific variable



```
#mean of age
mean(pat_info$Age)

#standard deviation of a variable
sd(pat_info$Age)
```

C:/Users/Matas/Desktop/R training/Project 1_HTN/myfirstRproject/

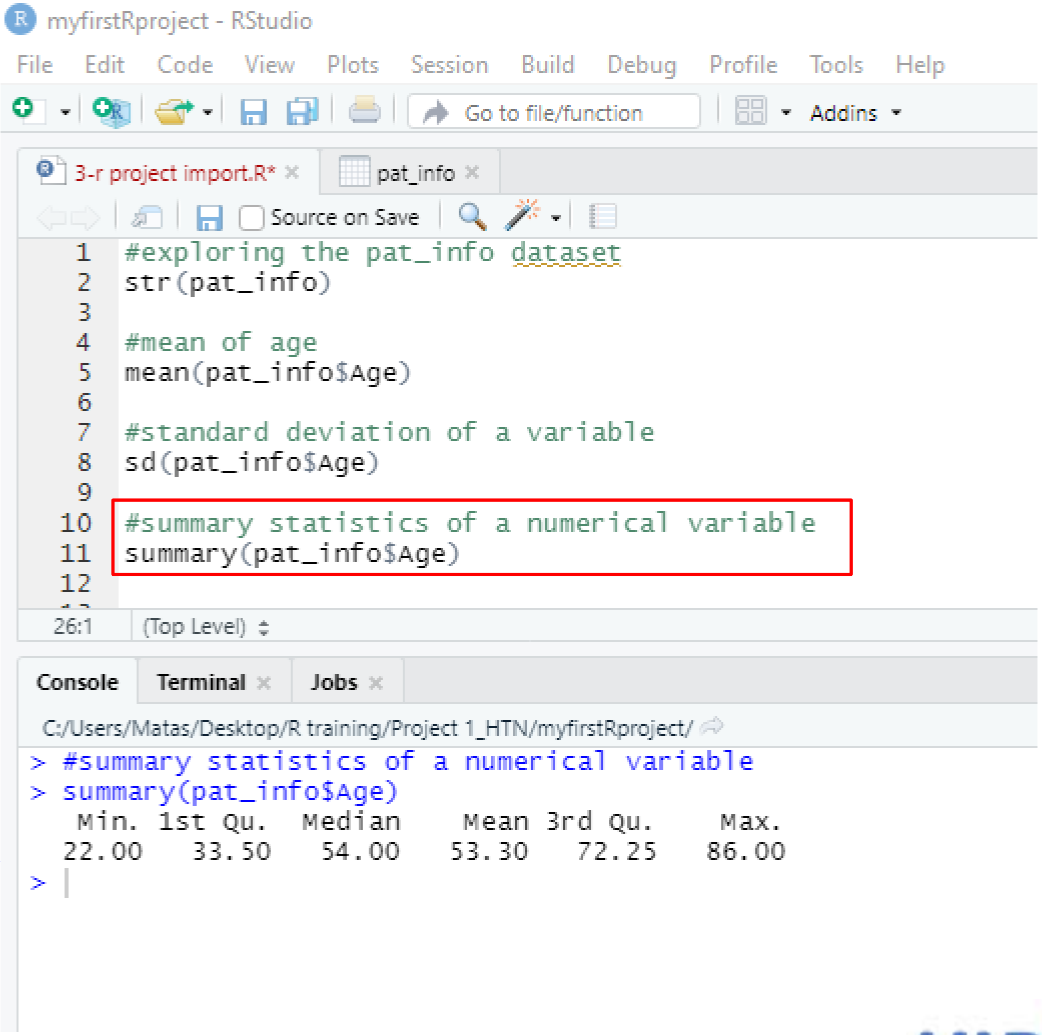
```
> #mean of age
> mean(pat_info$Age)
[1] 53.3
> #standard deviation of a variable
> sd(pat_info$Age)
[1] 22.13117
> 
```

summary of a variable

Type: summary(pat_info$Age)

The first part represents the name of the data frame and the second part after the dollar sign represents the specific variable

Summary statistics are here


myfirstRproject - RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
3-r project import.R*
pat_info
Source on Save
26:1
(Top Level)
Console
Terminal
Jobs
C:/Users/Matas/Desktop/R training/Project 1_HTN/myfirstRproject/

```
#exploring the pat_info dataset
str(pat_info)

#mean of age
mean(pat_info$Age)

#standard deviation of a variable
sd(pat_info$Age)

#summary statistics of a numerical variable
summary(pat_info$Age)
```

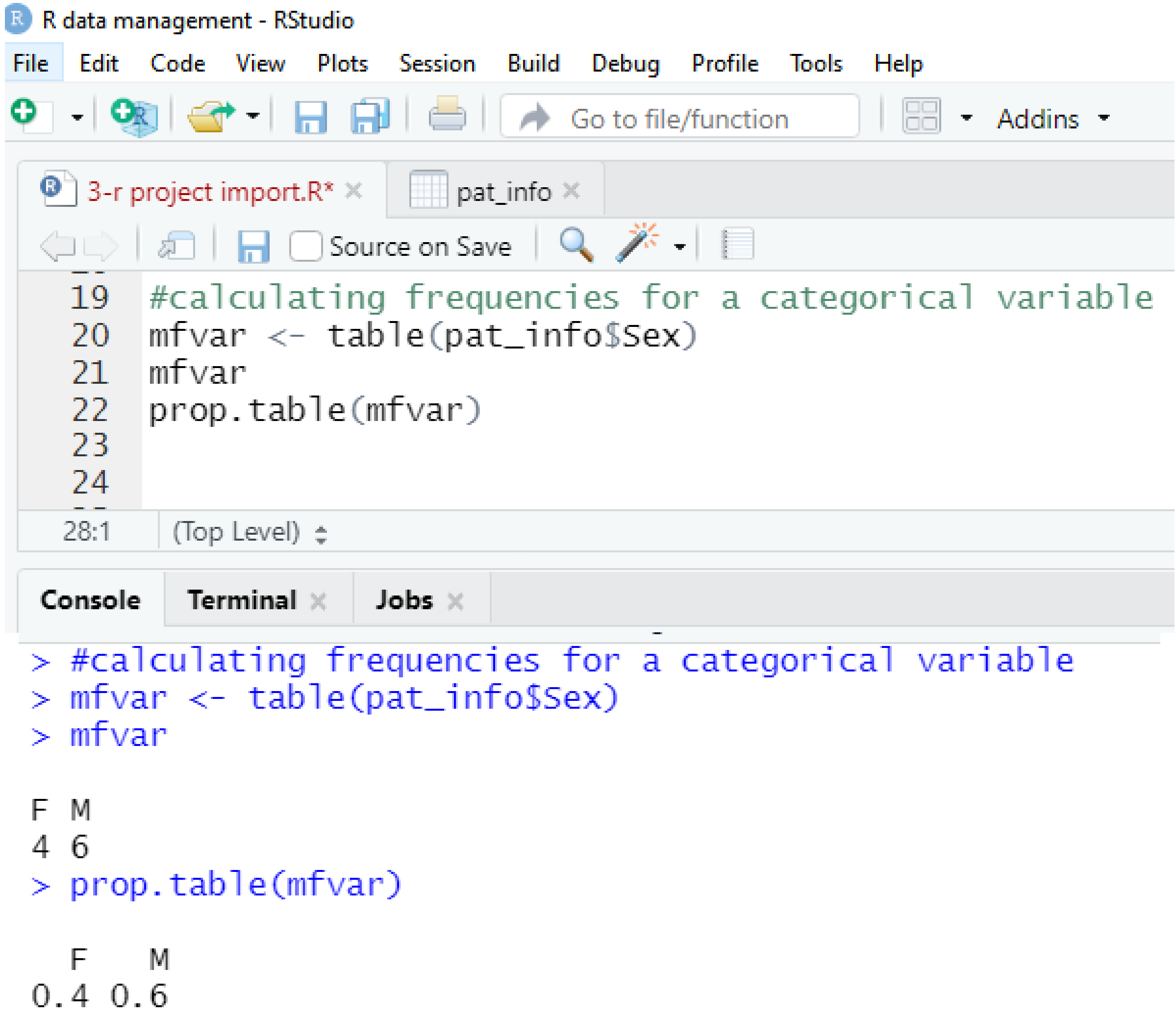
```
> #summary statistics of a numerical variable
> summary(pat_info$Age)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  22.00   33.50   54.00   53.30   72.25   86.00 
> 
```

© KIRCT

Management of categorical variables

Calculating a frequency of a categorical variable

We create a vector called "mfvar" to include the table counts of Sex.



```
#calculating frequencies for a categorical variable
mfvar <- table(pat_info$Sex)
mfvar
prop.table(mfvar)
```

```
> #calculating frequencies for a categorical variable
> mfvar <- table(pat_info$Sex)
> mfvar

F M 
4 6 
> prop.table(mfvar)

  F   M 
0.4 0.6 
```

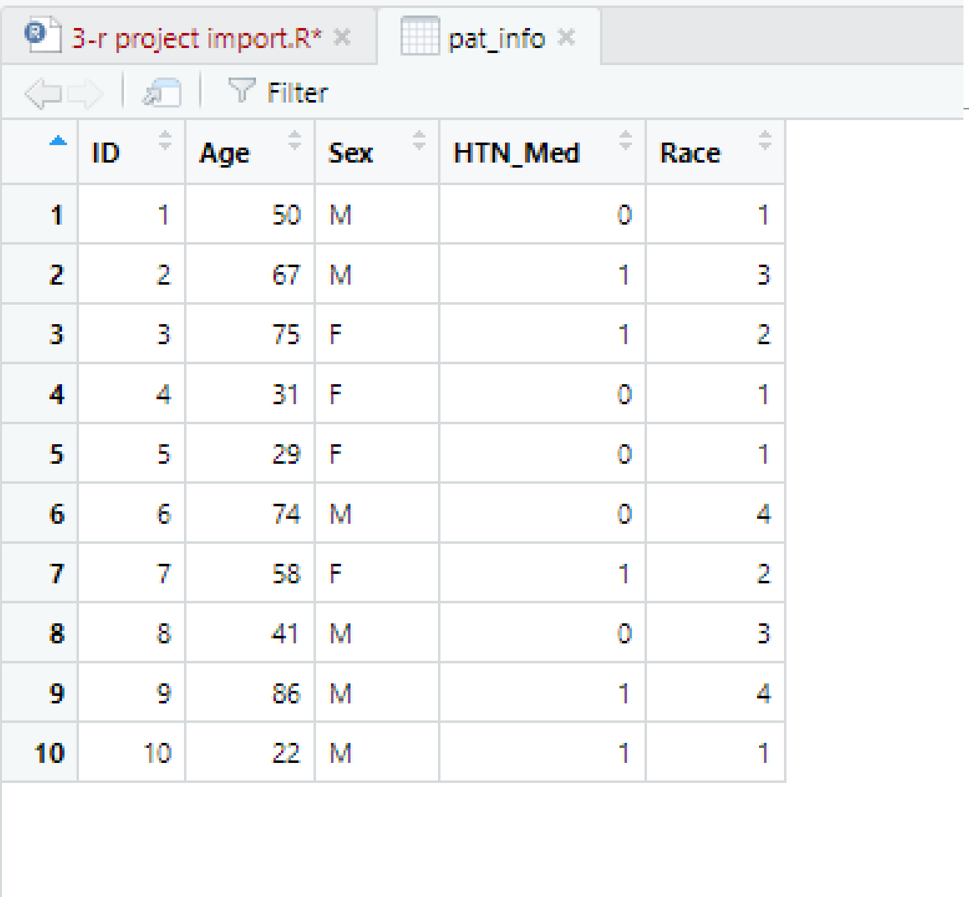
The table() command will provide you the count of the variable. In the dataset, we have 4 females and 6 males.

The prop.table(table) command will provide you with the proportion. 40% females and 60% males.

Adding labels

Notice we have a variable called race which identifies patients’ self-reported race as:
1= Non-Hispanic White (NHW)
2= Non-Hispanic Black (NHB)
3= Hispanic (HIS)
4= Other (OTH)

We will demonstrate how to add labels to the numerical values (1-4) of the qualitative variable “race”.

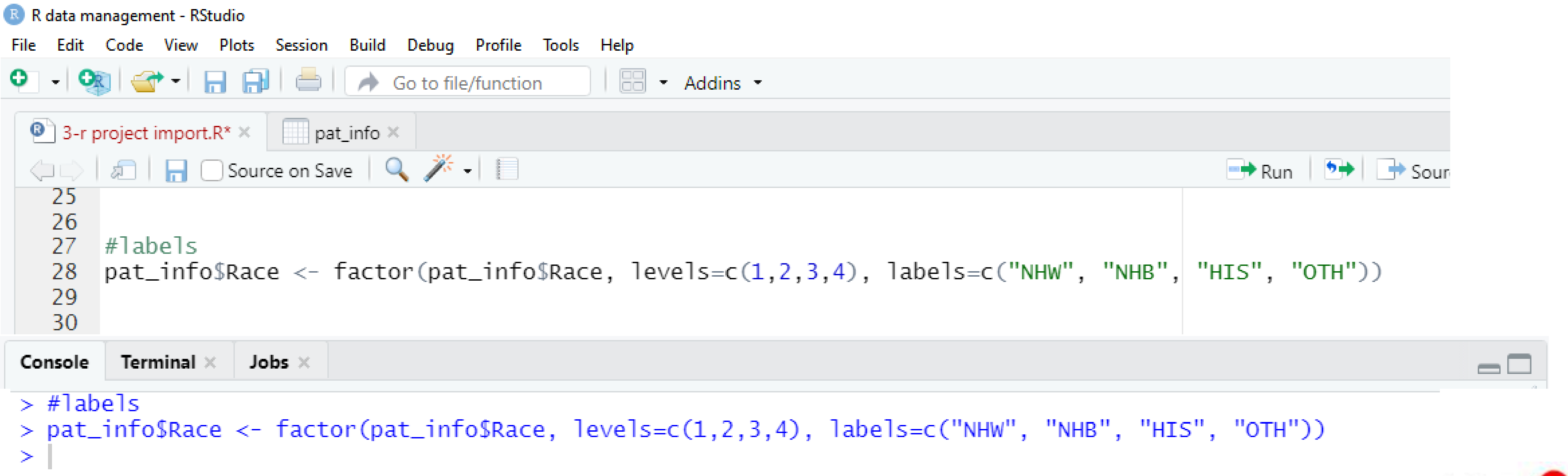


	ID	Age	Sex	HTN_Med	Race
1	1	50	M	0	1
2	2	67	M	1	3
3	3	75	F	1	2
4	4	31	F	0	1
5	5	29	F	0	1
6	6	74	M	0	4
7	7	58	F	1	2
8	8	41	M	0	3
9	9	86	M	1	4
10	10	22	M	1	1

Adding labels

We call out the variable Race by locating it in the pat_info dataset, convert it to a factor with 4 levels and label them in order as NHW, NHB, HIS, OTH.

Recall for categorical variables, we use factors. This is especially useful for statistical modeling.

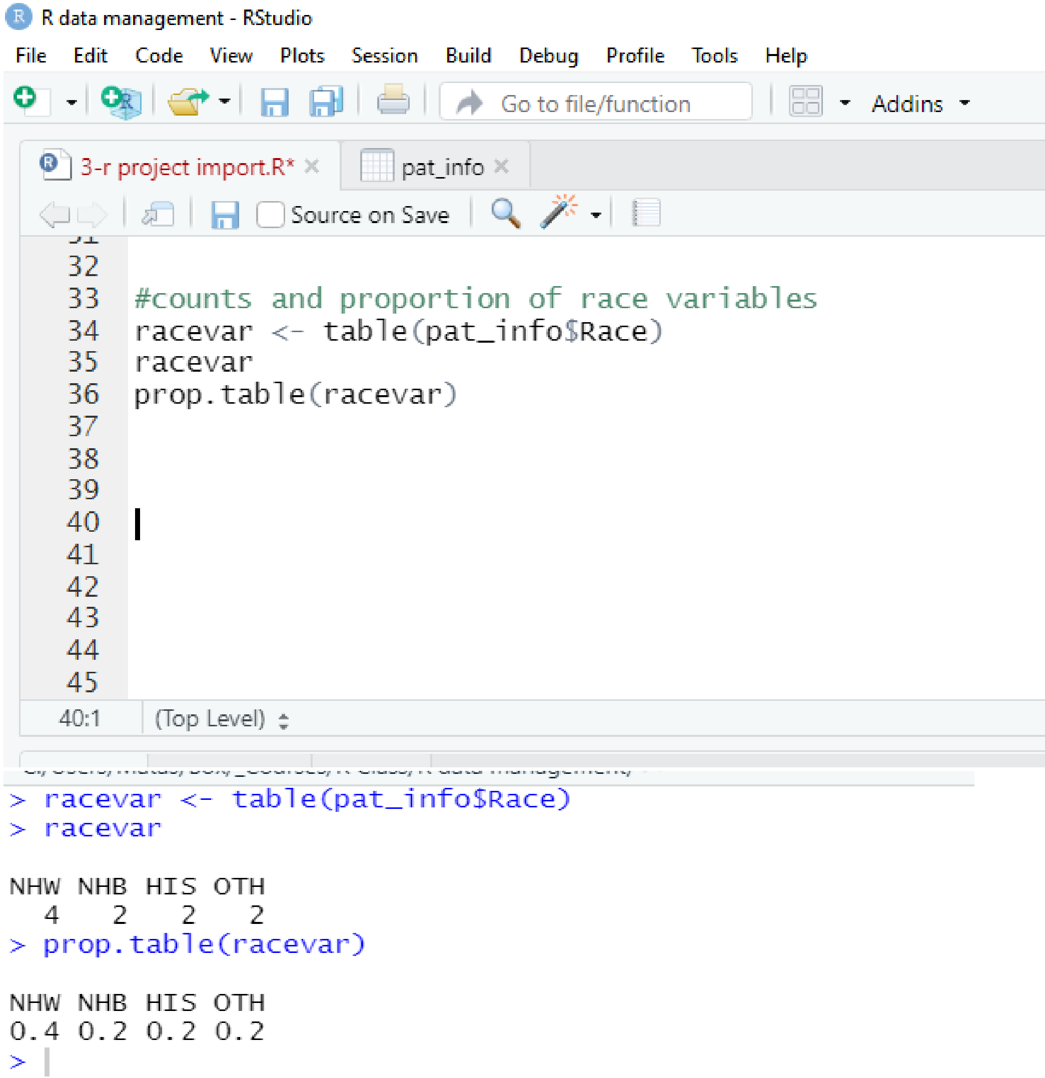


```
#labels
pat_info$Race <- factor(pat_info$Race, levels=c(1,2,3,4), labels=c("NHW", "NHB", "HIS", "OTH"))
```

```
> #labels
> pat_info$Race <- factor(pat_info$Race, levels=c(1,2,3,4), labels=c("NHW", "NHB", "HIS", "OTH"))
> 
```

Tabulate counts and frequencies by calling out the “table” function and “prop.table” function.

Results will show down here



```
#counts and proportion of race variables
racevar <- table(pat_info$Race)
racevar
prop.table(racevar)
```

```
> racevar <- table(pat_info$Race)
> racevar

NHW NHB HIS OTH
  4   2   2   2
> prop.table(racevar)

NHW NHB HIS OTH
0.4 0.2 0.2 0.2
> 
```

Management of both categorical and numerical variables

SUBGROUP ANALYSES

Subgroup analysis

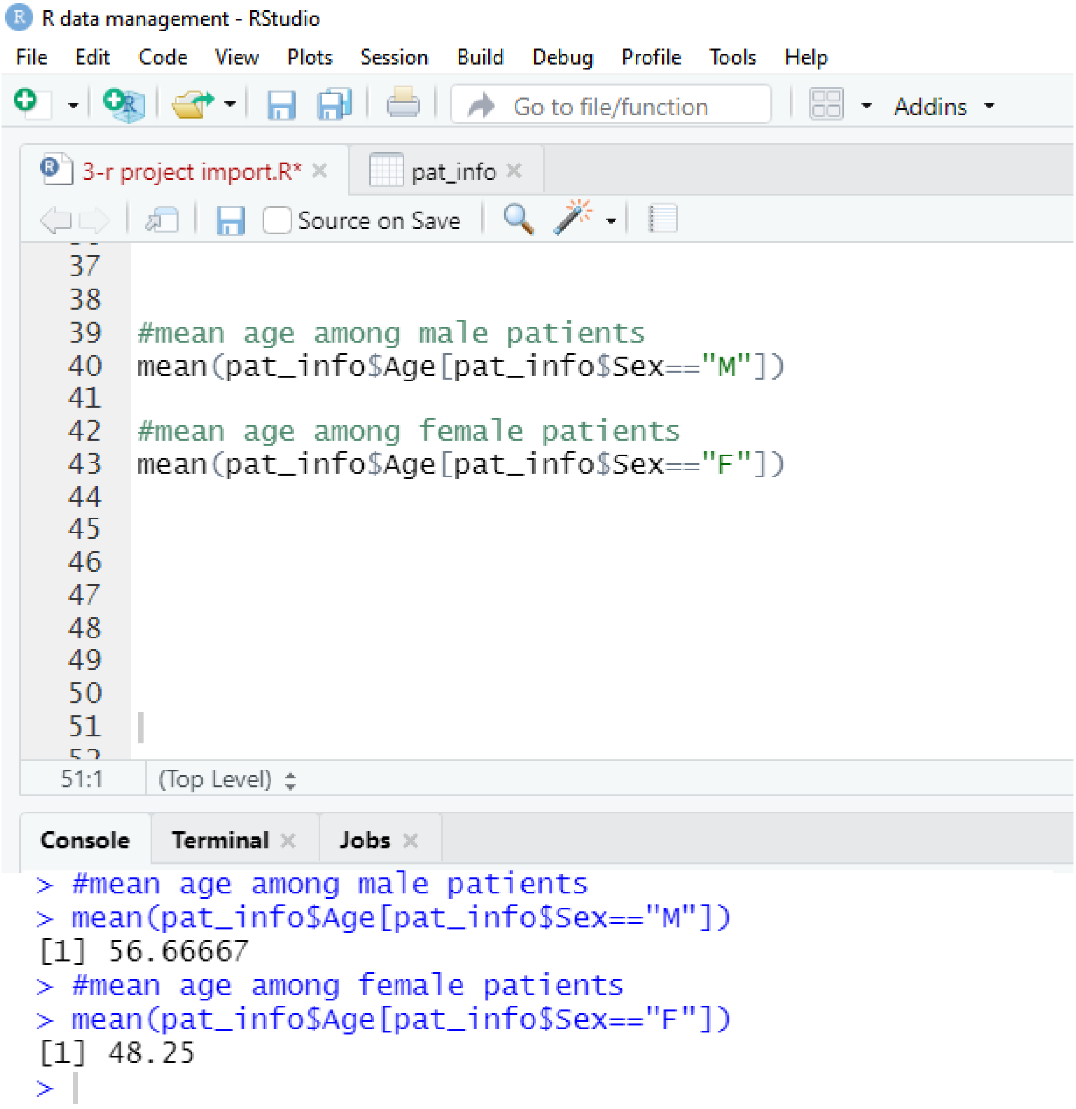
- Suppose you are interested in identifying the mean age of male and female patients.

- Code is provided on the next slide

This first part will give you the mean age of the patients. Notice how you have to include the dataframe name both times to identify the numerical variable, age and the categorical variable “M”

The same applies for “F”

The results show down here.



```
#mean age among male patients
mean(pat_info$Age[pat_info$Sex=="M"])

#mean age among female patients
mean(pat_info$Age[pat_info$Sex=="F"])
```

```
> #mean age among male patients
> mean(pat_info$Age[pat_info$Sex=="M"])
[1] 56.66667
> #mean age among female patients
> mean(pat_info$Age[pat_info$Sex=="F"])
[1] 48.25
> 
```

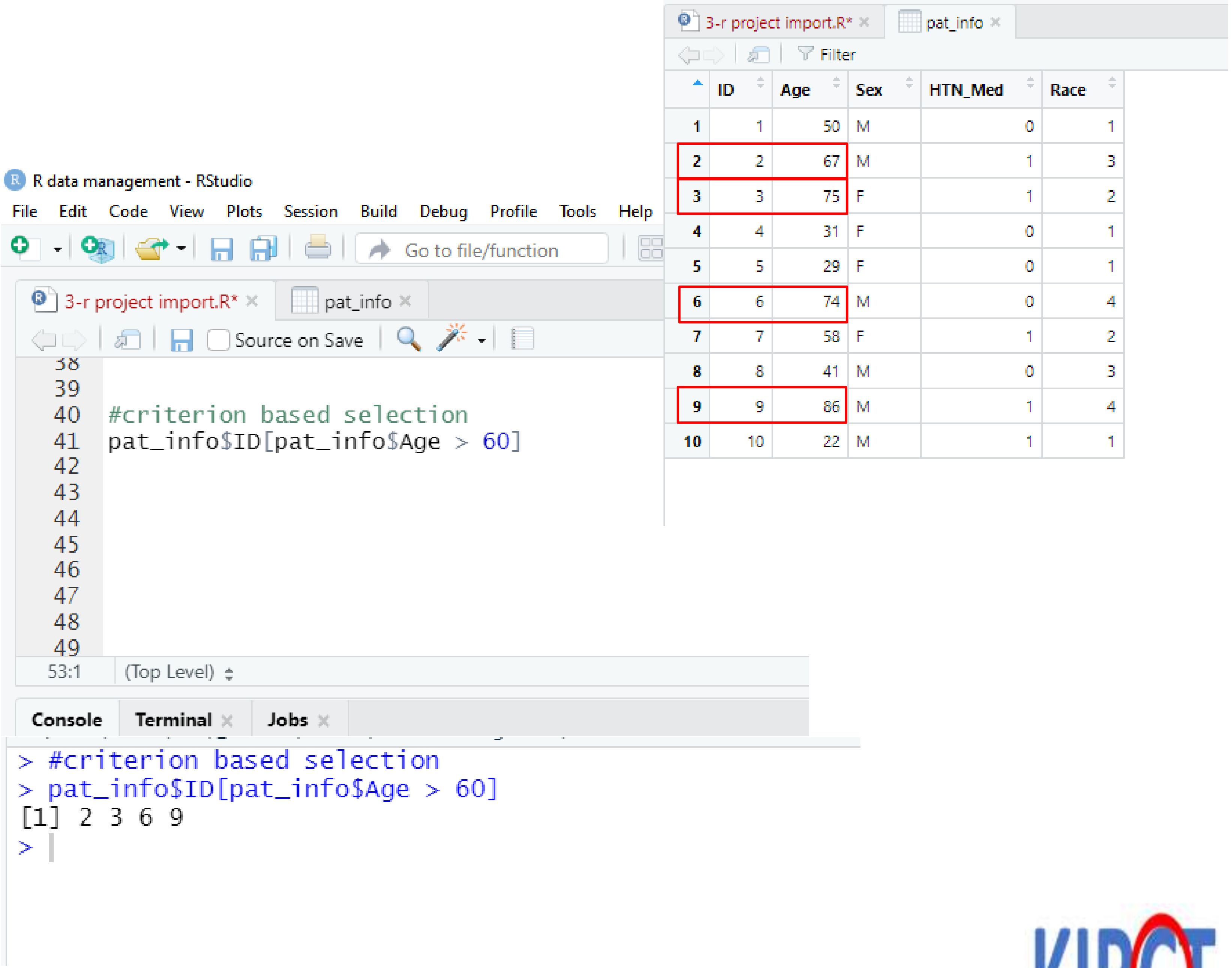
Criterion-based selection

Suppose you are interested in obtaining the ID numbers of patients above the age of 60.

Type pat_info$ID[pat_info$Age > 60]

The first part depicts the main variable of interest and the part inside the brackets depicts the specific criteria.

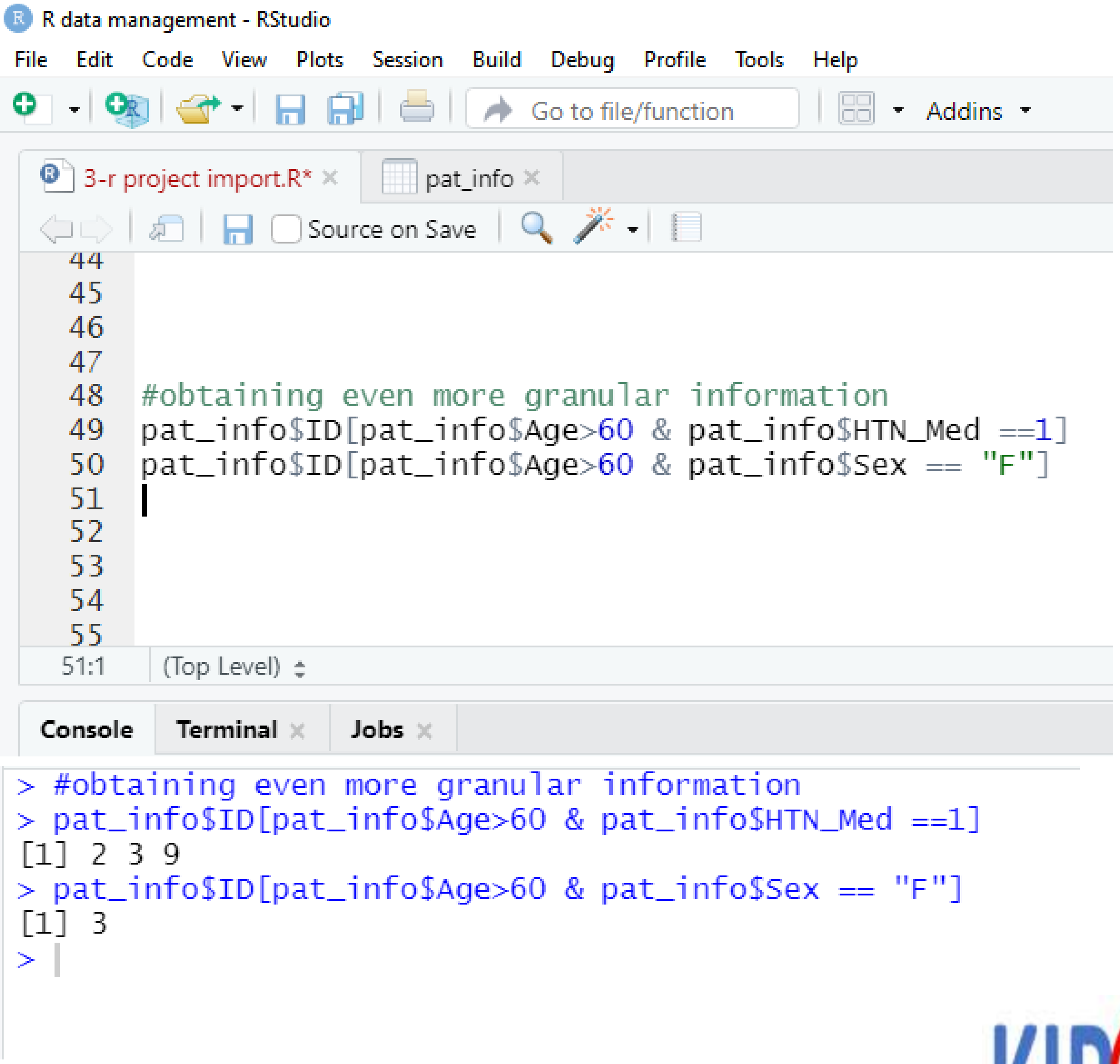
IDs are displayed down here:



```
#criterion based selection
pat_info$ID[pat_info$Age > 60]
```

```
> #criterion based selection
> pat_info$ID[pat_info$Age > 60]
[1] 2 3 6 9
>
```

	ID	Age	Sex	HTN_Med	Race
1	1	50	M	0	1
2	2	67	M	1	3
3	3	75	F	1	2
4	4	31	F	0	1
5	5	29	F	0	1
6	6	74	M	0	4
7	7	58	F	1	2
8	8	41	M	0	3
9	9	86	M	1	4
10	10	22	M	1	1



```
#obtaining even more granular information
pat_info$ID[pat_info$Age>60 & pat_info$HTN_Med ==1]
pat_info$ID[pat_info$Age>60 & pat_info$Sex == "F"]
```

```
> #obtaining even more granular information
> pat_info$ID[pat_info$Age>60 & pat_info$HTN_Med ==1]
[1] 2 3 9
> pat_info$ID[pat_info$Age>60 & pat_info$Sex == "F"]
[1] 3
> 
```

Obtaining IDs for patients above the age of 60 AND an indicator for being on hypertensive medication. Output: IDs 2,3,9

Obtaining IDs for patients about the age of 60 and female sex. Output: ID #3

Class Exercise #1: Module 03

The data in the Table below pertains to five patients with certain characteristics.

ID	Age	Sex	Diabetes	Race
1	55	M	1	2
2	70	F	0	1
3	40	M	0	1
4	20	M	1	3
5	63	F	1	2

Class Exercise #1: Module 03

Do the following:

a. Create the relevant variables in R

b. Frame the variables into a dataset

c. Calculate the frequency and proportion of individuals with diabetes by sex

Class Exercise #1: Module 03

d. Compare the mean age of individuals with versus those without diabetes

e. Label the race variable as follows: 1 = “White”, 2 = “Black”, and 3 = “Hispanic”.

Class Exercise #2: Module 03

The following Table contains information on age (in years), marital status (1= married, 2=divorced, 3=single), obesity status (1=obese,2=overweight, 3=normal weight) and race (1=White,2=Black, 3=Other) of ten patients as follows:

ID	Age	Marital status	Obesity status	Race
1	45	2	3	2
2	28	3	1	2
3	59	1	2	1
4	33	3	2	3
5	39	1	1	1
6	52	2	3	1
7	61	1	3	2
8	39	3	2	1
9	46	1	1	2
10	29	3	1	2

Class Exercise #2: Module 03

1. Create a vector for each of the variables in the Table

ii. Compile/frame the vectors into a dataset and give that dataset a name

iii. Change the numeric labels of variables marital status, obesity status and race to character labels

iv. Create a 3 by 3 table between race and obesity

v. Create a 3 by 3 table between race and marital status

Class Exercise #2: Module 03

vi. Create a 3 by 3 table between marital status and obesity

vii. Find the mean age for patients who are obese, for Blacks, Whites, and for those who are single

viii. How many patients are less than 40 years and obese?

ix. How many patients are black and less than 40?

x. How many patients are white and older than 40?

Summary

- In this lecture you learned:
  - How to import excel files into R environment
  - Numerical data manipulation
  - Categorical data manipulation
  - Basic sub-analyses incorporating both categorical and numerical values

- Next, we will discuss packages in R and how to use them.