Homework 03 - Nonstandard Evaluation and Git

Nonstandard Evaluation

Question 1

Imagine we have a data frame called `data`, with a `type` column. Which one works and why?
Function 1:

```
group_and_tally <- function(df, column){
    df %>% group_by({{ column }}) %>% tally();
}
group_and_tally(data, type);
```

Function 2:

```
group_and_tally <- function(df, column){
    df %>% group_by(column) %>% tally();
}
group_and_tally(data, type);
```

The {{}} brackets unquotes the column name, causing the function group_by to see whatever the column type is instead of the data frame and process it. the () brackets asks R to look for a column named column instead of whatever the value of "column" is as a string.

Git

For the questions below, please add the commands you used to complete these steps.

Question 2

Set up your git repo on your local computer. If you already make a git repo on GitHub, but it isn't on your local computer - clone it.

In [ ]:
```
I already had a git repo on GitHub but not on my local computer, so I cloned

git clone https://github.com/ssagar2930/BIOS512_assignments.git
Cloning into 'BIOS512_assignments'...
remote: Enumerating objects: 15, done.
remote: Counting objects: 100% (15/15), done.
```

```
remote: Compressing objects: 100% (11/11), done.
remote: Total 15 (delta 3), reused 3 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (15/15), 6.29 KiB | 2.10 MiB/s, done.
Resolving deltas: 100% (3/3), done.
```

Question 3

Set up your SSH key.

In [ ]:
```
I set up an SSH key the first week of class, so I did the following command

ls -al ~/.ssh

My output:
total 32
drwx------   6 ssagar  staff  192 Aug 23 16:14 .
drwxr-x---+ 31 ssagar  staff  992 Sep  9 08:29 ..
-rw-------   1 ssagar  staff  411 Aug 23 16:07 id_ed25519
-rw-r--r--   1 ssagar  staff  102 Aug 23 16:07 id_ed25519.pub
-rw-------   1 ssagar  staff  828 Aug 23 16:14 known_hosts
-rw-r--r--   1 ssagar  staff   92 Aug 23 16:14 known_hosts.old
```

Question 4

a) Add a HW2 directory to your git repo through the terminal with a HW.md file that says "This is for homework 2."

In [ ]:
```
mkdir HW2
cd /Users/ssagar/Desktop/UNC/BIOS512/BIOS512_assignments
mkdir HW2
cd HW2
echo "This is for homework 2." > HW2.md
git add HW2/HW2.md
git commit -m "Add HW2 directory with HW2.md for homework 2"
[main 5ea060e] Add HW2 directory with HW2.md for homework 2
```

b) *Add* HW2.md to the staging area. Then, use the command to see which files have been modified, staged for commit, or are untracked. What does it show? They should copy paste the terminal response after git status, and show that key used the commands below.

In [ ]:
```
git remote -v
origin  git@github.com:ssagar2930/BIOS512_assignments.git (fetch)
origin  git@github.com:ssagar2930/BIOS512_assignments.git (push)

git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (4/4), 370 bytes | 370.00 KiB/s, done.
```

```
Total 4 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:ssagar2930/BIOS512_assignments.git
   5613a95..5ea060e  main -> main
git status
On branch main
Your branch is up to date with 'origin/main'.
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        ../.DS_Store

nothing added to commit but untracked files present (use "git add" to track)
```

c) Save file changes to the main branch.

In [ ]:
```
ssagar@Shrutis-MBP HW2 % git branch
* main
ssagar@Shrutis-MBP HW2 % git commit -m "Add HW2.md for homework 2"
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        ../.DS_Store

nothing added to commit but untracked files present (use "git add" to track)
ssagar@Shrutis-MBP HW2 %
```

d) Now, edit the HW2.md file to give it a title.

In [ ]:
```
cd /Users/ssagar/Desktop/UNC/BIOS512/BIOS512_assignments
mv HW.md Homework2.md
```

e) Use the command that compares current, unsaved changes to the main branch. What does it say?

f) Use the command that checks the status of the working directory and the staging area *again*. What does it say?

g) Once again, add HW2.md to the staging area and save the file changes to the main branch. Then, get use the command that gives you project history and paste the output in your homework.

In [ ]:
```
diff --git a/HW2/HW.md b/HW2/HW2.md
deleted file mode 100644
index 1a010d3..0000000
--- a/HW2/HW2.md
+++ /dev/null
@@ -1 +0,0 @@
-This is for homework 2.

git status
```

```
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        deleted:    HW.md

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        ../.DS_Store
        Homework2.md

no changes added to commit (use "git add" and/or "git commit -a")
ssagar@shrutis-mbp HW2 %

git add Homework2.md
git commit -m "Add Homework2.md to staging and commit changes"
[main fa85ab9] Add Homework2.md to staging and commit changes
 Committer: Shruti Sagar <ssagar@shrutis-mbp.wireless-1x.unc.edu>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

 1 file changed, 1 insertion(+)
 create mode 100644 HW2/Homework2.md
```

h) Do some searching... What `git` command will provide you documentation on other commands? Use that command to find documentation on `git log` and `git show`. What does `--since` mean in regards to `git log`? Copy and paste what is written in the documentation.

In [ ]:

```
Commands:
git help log
git help show

Since:
--since=<date1> limits to commits newer than <date1>, and using it with
       --grep=<pattern> further limits to commits         whose log message
       that matches <pattern>), unless otherwise          noted.
--since=<date>, --after=<date>
           Show commits more recent than <date>.

--since-as-filter=<date>
           Show all commits more recent than                 <date>. This v
```

```
           in the range, rather than stopping at              the first comm
           older than <date>.
```

Tidyverse

Note: Please make sure Binder is set up correctly to run this section. You can follow the instructions here: https://github.com/rjenki/BIOS512.

Please show your code for this section! Before completing this section, please run the following.

In [2]:
```
library(tidyverse)
if (!dir.exists("intermediate")) dir.create("intermediate", recursive = TRUE
if (!exists("mdpre")) mdpre <- function(x) { print(x) }
if (!exists("ggmd"))  ggmd  <- function(p) { print(p) }
```

```
── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.
0 ──
✔ dplyr     1.1.4     ✔ readr     2.1.5
✔ forcats   1.0.0     ✔ stringr   1.5.1
✔ ggplot2   3.5.1     ✔ tibble    3.2.1
✔ lubridate 1.9.3     ✔ tidyr     1.3.1
✔ purrr     1.0.2
── Conflicts ────────────────────────────────────────── tidyverse_conflicts
() ──
✖ dplyr::filter() masks stats::filter()
✖ dplyr::lag()    masks stats::lag()
ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all c
onflicts to become errors
```

Question 5

Download the patient_names.csv and patient_properties.csv files from Canvas and read them into R. Manually set the date columns to be date variables. Print the first 10 observations of each.

In [16]:
```
library(dplyr)
library(lubridate)

patient_names <- read_csv("/Users/ssagar/Desktop/UNC/BIOS512/BIOS512_assignm
patient_properties <- read_csv("/Users/ssagar/Desktop/UNC/BIOS512/BIOS512_as

## checking structure of variables
str(patient_names)
str(patient_properties)

##BIRTHDATE and DEATHDATE in patient_names are dates
##patient_names$BIRTHDATE <- as.Date(patient_names$BIRTHDATE, format = "%m/%
##patient_names$DEATHDATE <- as.Date(patient_names$DEATHDATE, format = "%m/%

patient_names <- patient_names %>%
```

```
  mutate(
    BIRTHDATE = mdy(BIRTHDATE),
    DEATHDATE = mdy(DEATHDATE), ## reformatting characters to date variables
    BIRTHDATE = if_else(
      BIRTHDATE > as.Date("2020-01-01"),
      BIRTHDATE - years(100),
      BIRTHDATE
    )
  )

str(patient_names)

summary(patient_names$BIRTHDATE)

##printing first 10 observations
head(patient_names, 10)
head(patient_properties, 10)
```

```
Rows: 974 Columns: 7
── Column specification ────────────────────────────────────────────────────────
Delimiter: ","
chr (7): ID, BIRTHDATE, DEATHDATE, FIRST, LAST, CITY, STATE

ℹ Use `spec()` to retrieve the full column specification for this data.
ℹ Specify the column types or set `show_col_types = FALSE` to quiet this mes
sage.
Rows: 3896 Columns: 3
── Column specification ────────────────────────────────────────────────────────
Delimiter: ","
chr (3): ID, property, value

ℹ Use `spec()` to retrieve the full column specification for this data.
ℹ Specify the column types or set `show_col_types = FALSE` to quiet this mes
sage.
```

```
spc_tbl_ [974 × 7] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
 $ ID       : chr [1:974] "5605b66b-e92d-c16c-1b83-b8bf7040d51f" "6e5ae27c-8
038-7988-e2c0-25a103f01bfa" "8123d076-0886-9007-e956-d5864aa121a7" "770518e4
-6133-648e-60c9-071eb2f0e2ce" ...
 $ BIRTHDATE: chr [1:974] "3/19/77" "2/19/40" "6/4/58" "12/25/28" ...
 $ DEATHDATE: chr [1:974] NA NA NA "9/29/17" ...
 $ FIRST    : chr [1:974] "Nikita578" "Zane918" "Quinn173" "Abel832" ...
 $ LAST     : chr [1:974] "Erdman779" "Hodkiewicz467" "Marquardt819" "Smitha
m825" ...
 $ CITY     : chr [1:974] "Quincy" "Boston" "Quincy" "Boston" ...
 $ STATE    : chr [1:974] "Massachusetts" "Massachusetts" "Massachusetts" "M
assachusetts" ...
 - attr(*, "spec")=
  .. cols(
  ..   ID = col_character(),
  ..   BIRTHDATE = col_character(),
  ..   DEATHDATE = col_character(),
  ..   FIRST = col_character(),
  ..   LAST = col_character(),
  ..   CITY = col_character(),
  ..   STATE = col_character()
  .. )
 - attr(*, "problems")=<externalptr>
spc_tbl_ [3,896 × 3] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
 $ ID       : chr [1:3896] "5605b66b-e92d-c16c-1b83-b8bf7040d51f" "5605b66b-e
92d-c16c-1b83-b8bf7040d51f" "5605b66b-e92d-c16c-1b83-b8bf7040d51f" "5605b66b
-e92d-c16c-1b83-b8bf7040d51f" ...
 $ property: chr [1:3896] "MARITAL" "RACE" "ETHNICITY" "GENDER" ...
 $ value   : chr [1:3896] "M" "white" "nonhispanic" "F" ...
 - attr(*, "spec")=
  .. cols(
  ..   ID = col_character(),
  ..   property = col_character(),
  ..   value = col_character()
  .. )
 - attr(*, "problems")=<externalptr>
tibble [974 × 7] (S3: tbl_df/tbl/data.frame)
 $ ID       : chr [1:974] "5605b66b-e92d-c16c-1b83-b8bf7040d51f" "6e5ae27c-8
038-7988-e2c0-25a103f01bfa" "8123d076-0886-9007-e956-d5864aa121a7" "770518e4
-6133-648e-60c9-071eb2f0e2ce" ...
 $ BIRTHDATE: Date[1:974], format: "1977-03-19" "1940-02-19" ...
 $ DEATHDATE: Date[1:974], format: NA NA ...
 $ FIRST    : chr [1:974] "Nikita578" "Zane918" "Quinn173" "Abel832" ...
 $ LAST     : chr [1:974] "Erdman779" "Hodkiewicz467" "Marquardt819" "Smitha
m825" ...
 $ CITY     : chr [1:974] "Quincy" "Boston" "Quincy" "Boston" ...
 $ STATE    : chr [1:974] "Massachusetts" "Massachusetts" "Massachusetts" "M
assachusetts" ...
```

Min.: 1922-03-24 **1st Qu.:** 1933-05-23 **Median:** 1950-05-22 **Mean:** 1952-04-02

3rd Qu.: 1970-03-14 **Max.:** 1991-11-27

A tibble: 10 × 7

ID	BIRTHDATE	DEATHDATE	FIRST	LAST	CITY	
<chr>	<date>	<date>	<chr>	<chr>	<chr>	
5605b66b-e92d-c16c-1b83-b8bf7040d51f	1977-03-19	NA	Nikita578	Erdman779	Quincy	Massac
6e5ae27c-8038-7988-e2c0-25a103f01bfa	1940-02-19	NA	Zane918	Hodkiewicz467	Boston	Massac
8123d076-0886-9007-e956-d5864aa121a7	1958-06-04	NA	Quinn173	Marquardt819	Quincy	Massac
770518e4-6133-648e-60c9-071eb2f0e2ce	1928-12-25	2017-09-29	Abel832	Smitham825	Boston	Massac
f96addf5-81b9-0aab-7855-d208d3d352c5	1928-12-25	2014-02-23	Edwin773	Labadie908	Boston	Massac
8e9650d1-788a-78f9-4a28-d08f7f95354a	1928-12-25	NA	Frankie174	Oberbrunner298	Boston	Massac
183df435-4190-060e-8f8e-bf63c572b266	1957-11-08	NA	Eilene124	Walsh511	Cambridge	Massac
720560d4-51da-c38c-ee90-c15935278df1	1972-06-27	NA	Lowell343	Price929	Quincy	Massac
217851b0-5f47-d376-18b9-0fe4ba77207e	1954-03-06	NA	Adrian111	Gleason633	Boston	Massac
ff331e5c-ab16-e218-f39a-63e11de1ed75	1927-07-10	NA	Eugene421	Abernathy524	Boston	Massac

A tibble: 10 × 3

ID	property	value
<chr>	<chr>	<chr>
5605b66b-e92d-c16c-1b83-b8bf7040d51f	MARITAL	M
5605b66b-e92d-c16c-1b83-b8bf7040d51f	RACE	white
5605b66b-e92d-c16c-1b83-b8bf7040d51f	ETHNICITY	nonhispanic
5605b66b-e92d-c16c-1b83-b8bf7040d51f	GENDER	F
6e5ae27c-8038-7988-e2c0-25a103f01bfa	MARITAL	M
6e5ae27c-8038-7988-e2c0-25a103f01bfa	RACE	white
6e5ae27c-8038-7988-e2c0-25a103f01bfa	ETHNICITY	nonhispanic
6e5ae27c-8038-7988-e2c0-25a103f01bfa	GENDER	M
8123d076-0886-9007-e956-d5864aa121a7	MARITAL	M
8123d076-0886-9007-e956-d5864aa121a7	RACE	white

Question 6

In the data frame pulled from patient_properties, you'll notice that the data is long, not wide. Do a pivot to make the properties their own columns. Print the first 10 observations after you do so.

In [8]:
```
patient_properties_wide <- (patient_properties %>% pivot_wider(id_cols=ID, r
head(patient_properties_wide, 10)
```

A tibble: 10 × 5

ID	MARITAL	RACE	ETHNICITY	GENDER
<chr>	<chr>	<chr>	<chr>	<chr>
5605b66b-e92d-c16c-1b83-b8bf7040d51f	M	white	nonhispanic	F
6e5ae27c-8038-7988-e2c0-25a103f01bfa	M	white	nonhispanic	M
8123d076-0886-9007-e956-d5864aa121a7	M	white	nonhispanic	M
770518e4-6133-648e-60c9-071eb2f0e2ce	M	white	hispanic	M
f96addf5-81b9-0aab-7855-d208d3d352c5	M	white	hispanic	M
8e9650d1-788a-78f9-4a28-d08f7f95354a	M	white	hispanic	M
183df435-4190-060e-8f8e-bf63c572b266	M	asian	nonhispanic	F
720560d4-51da-c38c-ee90-c15935278df1	M	white	nonhispanic	M
217851b0-5f47-d376-18b9-0fe4ba77207e	S	black	hispanic	M
ff331e5c-ab16-e218-f39a-63e11de1ed75	M	native	hispanic	M

Question 7

Perform a left join of the names and properties_wide data frames by the ID column and print the first 10 rows.

In [9]:
```
patient_names_left <- patient_names %>% left_join(patient_properties_wide %>
head(patient_names_left, 10)
```

```
Joining with `by = join_by(ID)`
```

A tibble: 10 × 11

ID	BIRTHDATE	DEATHDATE	FIRST	LAST	CITY	
<chr>	<date>	<date>	<chr>	<chr>	<chr>	
5605b66b-e92d-c16c-1b83-b8bf7040d51f	1977-03-19	NA	Nikita578	Erdman779	Quincy	Massac
6e5ae27c-8038-7988-e2c0-25a103f01bfa	1940-02-19	NA	Zane918	Hodkiewicz467	Boston	Massac
8123d076-0886-9007-e956-d5864aa121a7	1958-06-04	NA	Quinn173	Marquardt819	Quincy	Massac
770518e4-6133-648e-60c9-071eb2f0e2ce	1928-12-25	2017-09-29	Abel832	Smitham825	Boston	Massac
f96addf5-81b9-0aab-7855-d208d3d352c5	1928-12-25	2014-02-23	Edwin773	Labadie908	Boston	Massac
8e9650d1-788a-78f9-4a28-d08f7f95354a	1928-12-25	NA	Frankie174	Oberbrunner298	Boston	Massac
183df435-4190-060e-8f8e-bf63c572b266	1957-11-08	NA	Eilene124	Walsh511	Cambridge	Massac
720560d4-51da-c38c-ee90-c15935278df1	1972-06-27	NA	Lowell343	Price929	Quincy	Massac
217851b0-5f47-d376-18b9-0fe4ba77207e	1954-03-06	NA	Adrian111	Gleason633	Boston	Massac
ff331e5c-ab16-e218-f39a-63e11de1ed75	1927-07-10	NA	Eugene421	Abernathy524	Boston	Massac

Question 8

Notice something interesting about the names in our data set. Fix the name formatting and print the first 10 observations.

In [10]:
```
library(stringr)
patient_names_left$FIRST <- str_replace_all(patient_names_left$FIRST, "[^A-Z
patient_names_left$LAST <- str_replace_all(patient_names_left$LAST, "[^A-Za-

head(patient_names_left, 10)
```

A tibble: 10 × 11

ID	BIRTHDATE	DEATHDATE	FIRST	LAST	CITY	STATE
<chr>	<date>	<date>	<chr>	<chr>	<chr>	<chr>
5605b66b-e92d-c16c-1b83-b8bf7040d51f	1977-03-19	NA	Nikita	Erdman	Quincy	Massachusetts
6e5ae27c-8038-7988-e2c0-25a103f01bfa	1940-02-19	NA	Zane	Hodkiewicz	Boston	Massachusetts
8123d076-0886-9007-e956-d5864aa121a7	1958-06-04	NA	Quinn	Marquardt	Quincy	Massachusetts
770518e4-6133-648e-60c9-071eb2f0e2ce	1928-12-25	2017-09-29	Abel	Smitham	Boston	Massachusetts
f96addf5-81b9-0aab-7855-d208d3d352c5	1928-12-25	2014-02-23	Edwin	Labadie	Boston	Massachusetts
8e9650d1-788a-78f9-4a28-d08f7f95354a	1928-12-25	NA	Frankie	Oberbrunner	Boston	Massachusetts
183df435-4190-060e-8f8e-bf63c572b266	1957-11-08	NA	Eilene	Walsh	Cambridge	Massachusetts
720560d4-51da-c38c-ee90-c15935278df1	1972-06-27	NA	Lowell	Price	Quincy	Massachusetts
217851b0-5f47-d376-18b9-0fe4ba77207e	1954-03-06	NA	Adrian	Gleason	Boston	Massachusetts
ff331e5c-ab16-e218-f39a-63e11de1ed75	1927-07-10	NA	Eugene	Abernathy	Boston	Massachusetts

Question 9

Using a for statement to loop through the categorical variables (excluding name and ID), print the counts of each unique value in descending order, using the mdpre() function for formatting.

In [11]:
```
mdpre <- function(x) {
  cat("```\n")
  print(x)
  cat("```\n\n")
}
# Columns to exclude
excluded_cols <- c("FIRST", "LAST", "ID")

# Loop through columns
for (colname in names(patient_names_left)) {
  if (!(colname %in% excluded_cols) && (is.character(patient_names_left[[col
  cat(paste0("### Counts for: ", colname, "\n"))

 # Count unique values in descending order
counts <- sort(table(patient_names_left[[colname]]), decreasing = TRUE)
mdpre(counts)
    }
}
```

Counts for: CITY
```
        Boston         Quincy      Cambridge         Revere        Chelsea
           541             80             45             42             39
      Weymouth     Somerville        Hingham       Winthrop      Brookline
            37             25             22             22             17
       Everett           Hull       Medford      Braintree       Cohasset
            16             15             13             10             10
        Malden       Scituate        Newton       Stoneham North Scituate
             8              8              6              5              3
       Reading        Belmont      Lynnfield        Melrose         Milton
             2              1              1              1              1
       Norwell        Waltham      Watertown     Winchester
             1              1              1              1
```

Counts for: STATE
```
Massachusetts
          974
```

Counts for: MARITAL
```

   M    S Fine male
 782  189    1    1
```

Counts for: RACE
```

   white    black    asian    other hawaiian   native   asiann
     680      163       90       16       13       11        1
```

Counts for: ETHNICITY
```

nonhispanic    hispanic  nonhispani     hispani
        781         190           2           1
```

Counts for: GENDER
```

     M      F Female   Male female
   493    478      1      1      1
```

Question 10

If you see any weird values, get rid of the ones that don't make sense, and combine the ones that are formatted wrong. Don't forget ot check the dates! Print the new tables for categorical values, and print the date ranges.

In [17]:
```
library(dplyr)

## cleaning marital values
patients_marital <- patient_names_left %>% mutate(MARITAL = case_when(
    MARITAL == "Fine" ~ NA_character_, ## replacing Fine with missing
    MARITAL == "male" ~ NA_character_, ## replacing male with missing
    TRUE ~ MARITAL ## everything else is the same
))

table(patients_marital$MARITAL) ## printing all unique values and counts to

## cleaning race values
patients_race <- patients_marital %>% mutate(RACE = case_when(
    RACE == "asiann" ~ "asian",
    TRUE ~ RACE ## everything else is the same
))

table(patients_race$RACE)

## cleaning ethnicity values
patients_ethnicity <- patients_race %>% mutate(ETHNICITY = case_when(
    ETHNICITY == "nonhispani" ~ "nonhispanic",
    ETHNICITY == "hispani" ~ "hispanic",
    TRUE ~ ETHNICITY ## everything else is the same
))

table(patients_ethnicity$ETHNICITY)

## cleaning gender values
patients_gender <- patients_race %>% mutate(GENDER = case_when(
    GENDER == "Female" ~ "F",
    GENDER == "female" ~ "F",
    GENDER == "Male" ~ "M",
    TRUE ~ GENDER ## everything else is the same
))

table(patients_gender$GENDER)

## print BIRTHDATE range
summary(patients_gender$BIRTHDATE, na.rm=TRUE)

## print DEATHDATE range
summary(patients_gender$DEATHDATE, na.rm=TRUE)
```

```
  M   S 
782 189 
```

```
   asian    black hawaiian   native    other    white 
      91      163       13       11       16      680 
```

```
  hispanic nonhispanic 
       191         783 
```

```
  F   M
480 494
```

Min.: 1922-03-24 **1st Qu.:** 1933-05-23 **Median:** 1950-05-22 **Mean:** 1952-04-02 **3rd Qu.:** 1970-03-14 **Max.:** 1991-11-27

Min.: 2011-02-03 **1st Qu.:** 2014-03-09 **Median:** 2017-07-20 **Mean:** 2017-01-20 **3rd Qu.:** 2019-07-31 **Max.:** 2022-01-27

Question 11

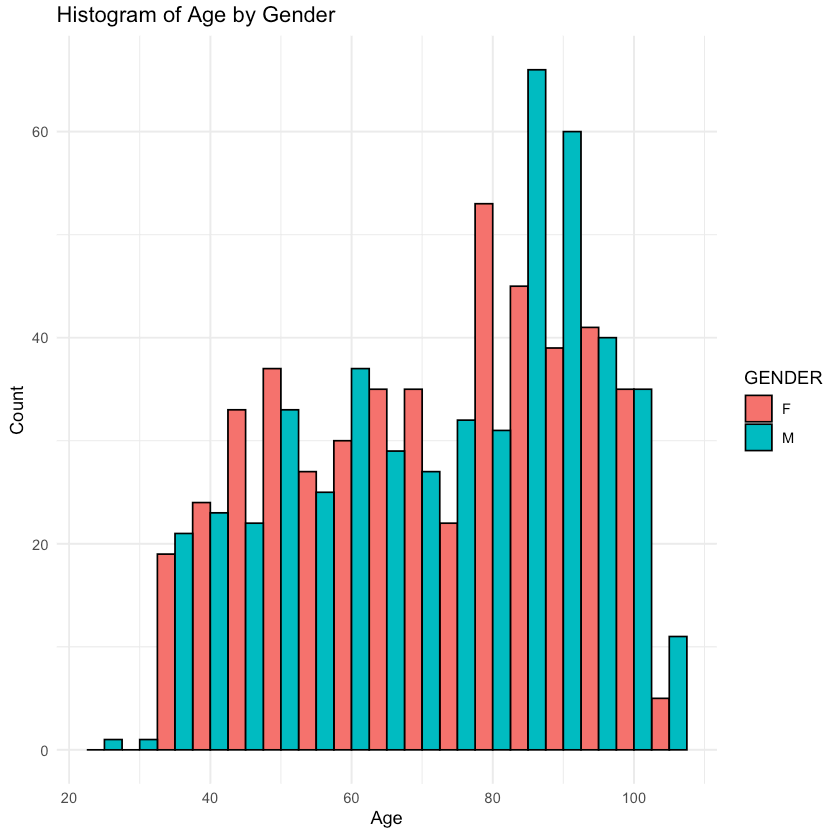
Make a histogram of the ages of patients by gender.

In [14]:
```
## creating an age variable

patients_clean <- patients_gender %>%
  mutate(AGE = if_else(
    !is.na(DEATHDATE) & DEATHDATE >= BIRTHDATE,
    as.numeric(DEATHDATE - BIRTHDATE) / 365.25,
    as.numeric(Sys.Date() - BIRTHDATE) / 365.25
  ))

library(ggplot2)

ggplot(patients_clean, aes(x=AGE, fill=GENDER)) +
     geom_histogram(position = "dodge", binwidth = 5, color = "black") +
  labs(title = "Histogram of Age by Gender",
       x = "Age",
       y = "Count") +
  theme_minimal()
```

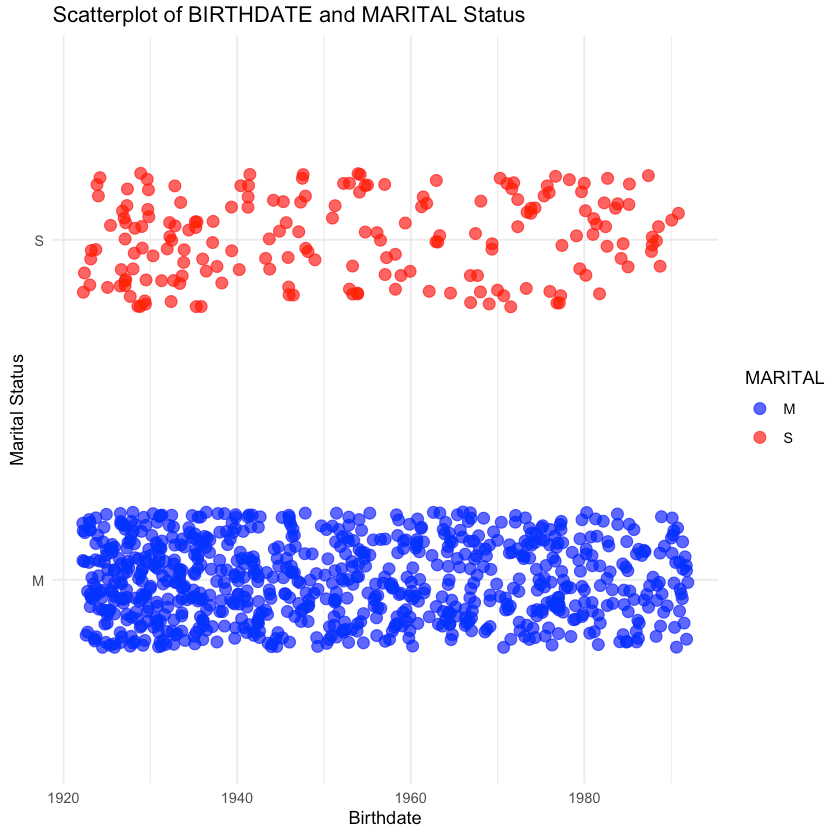


Question 12

Make a scatterplot of birthdate by martial status.

In [15]:
```
library(ggplot2)

patients_clean %>%
  filter(!is.na(MARITAL)) %>%
  ggplot(aes(x = BIRTHDATE, y = MARITAL, color = MARITAL)) +
  geom_jitter(width = 0, height = 0.2, alpha = 0.7, size = 3) +
  scale_color_manual(values = c("M" = "blue", "S" = "red")) +
  labs(title = "Scatterplot of BIRTHDATE and MARITAL Status",
       x = "Birthdate",
       y = "Marital Status") +
  theme_minimal()
```


Scatterplot of BIRTHDATE and MARITAL Status
Marital Status
S
M
1920
1940
1960
1980
Birthdate
MARITAL
M
S

In [ ]: