```
# Load Packages ------------------------------------------------------------
library(readxl)
library(dplyr)
library(tidyr)
library(tidyverse)
library(naniar)
library(lubridate)
library(ggthemes)
library(scales)

# Import Data --------------------------------------------------------------
Income_2019 <- read_excel("~/Data Analysis/Sailboat Charter Business/2019.xlsx", sheet = "Income", skip = 1)
Income_2019 <- Income_2019[-(228:253), ] # Remove bottom rows
Income_2020 <- read_excel("~/Data Analysis/Sailboat Charter Business/2020.xlsx", sheet = "Income", skip = 1)
Income_2020 <- Income_2020[-(206:290), ] # Remove bottom rows
Income_2021 <- read_excel("~/Data Analysis/Sailboat Charter Business/2021.xlsx", sheet = "Income", skip = 1)
Income_2021 <- Income_2021[-(320:343), ] # Remove bottom rows
Income_2022 <- read_excel("~/Data Analysis/Sailboat Charter Business/2022.xlsx", sheet = "Income", skip = 1)
Income_2022 <- Income_2022[-(368:392), ] # Remove bottom rows
Income_2022 <- Income_2022 %>% select(-Captain) # Remove extra column from 2022 data to match number of columns in other years

# Combine all four years into one table
Income <- rbind(Income_2019, Income_2020, Income_2021, Income_2022)

# Data Cleaning ------------------------------------------------------------
Income <- Income %>% select(-`Event #`) # Event is irrelevant for combined table
Income <- subset(Income, Date != "TBD") # Removes events that never occurred (e.g., cancellations)
Income <- subset(Income, Date != "TBD`") # One "TBD" entry was misspelled...
# More important question - does "misspelled" have one or two 's's?
Income <- subset(Income, Date != "8 Seo") #Removes erroneous entry

# Convert data types
Income$Date <- as.numeric(Income$Date) # Convert Date column from chr to num
Income$Date <- as.Date(Income$Date, origin = as.Date("1899-12-31")) # Convert Date column from num to Date
```

```
Income$Date <- as.POSIXct(Income$Date, format = "%y-%m-%d")
Income <- Income %>% mutate(Year = format(Income$Date, format = "%Y")) # Creates a column with just the year
Income$Discounts <- as.numeric(Income$Discounts) # Convert Discounts column from chr to num
Income$`Sailing Charter` <- as.numeric(Income$`Sailing Charter`) # Convert Sailing Charter column from chr to num

# Further cleaning of data
# Remove income from "Other" sources (e.g., boat delivery, private lessons, and purchases of material)
Income <- subset(Income, !is.na(`Intro to Sailing`) | !is.na(`ASA 101`) | !is.na(`ASA 103`) | !is.na(`ASA 104`)
                 | !is.na(`ASA 105`) | !is.na(`ASA 106`) | !is.na(`Sailing Charter`) | !is.na(`Dinner Charters`))
Income <- Income %>% select(-Other) # Removes the now irrelevant "Other" column
Income[is.na(Income)] <- 0 # Convert all NAs to 0
Income <- Income %>% select(-Discounts) # Removes the "Discounts" column since the lesson prices are already adjusted to include the discount
Income$Total <- rowSums(Income[ , c(4:16)]) # Creates a column with the total income earned from each event
Income <- filter(Income, Year < 2023) # Limit scope of date to 2019-2022

# "Description" column needs to be split to identify whether the event is a charter or lesson and the length of the charter or type of lesson, as
applicable
Income <- Income %>% separate(Description, c("Charter Length", "Lesson Type"), " |-") # "Charter Length" now lists the length of the charter in
hours
Income <- Income %>% replace_with_na(Income$`Charter Length`, replace = list(`Charter Length` = c("ASA", "ITS"))) # Changes non-charter
values to NA
#Some more data cleaning...
Income$`Lesson Type`[Income$`Lesson Type` == "Yvette"] <- "ITS" # One Introduction to Sailing lesson was listed by the student's name
Income$`Lesson Type`[Income$`Lesson Type` == ""] <- "104" # One ASA 104 lesson was listed as blank
Income$`Lesson Type`[Income$`Lesson Type` == "Susnet"] <- "Sunset" # Correct misspelling of "sunset"
Income$`Lesson Type`[Income$`Lesson Type` == "TBD"] <- "Day" # This entry wasn't categorized by time of day, but there is another event on the
same day that occured at sunset, so presumably this one was during the day
Income$`Lesson Type`[Income$`Lesson Type` == "Charter"] <- "Sunset" # This entry wasn't categorized, assuming "sunset"
Income$`Lesson Type`[Income$`Lesson Type` == "101/3/4"] <- "101/103/104" # Matches terminology for 101/103/104 course
# Continue split of "Description" column
Income <- transform(Income, `Event Type` = ifelse(is.na(`Charter Length`), "Lesson", "Charter")) # Creates column to identify each event as a
"lesson" or "charter"
Income <- Income %>% mutate(Charter.Type = "") # Create empty column for the type of charter
Income$Charter.Type <- ifelse(!is.na(Income$Charter.Length), Income$Lesson.Type, NA) # Populate "Charter Type" column
```

```
# Reformat Columns ----------------------------------------------------------
# Create a column listing how many hours each event required
Lesson.Type <- c("ITS", "101", "103", "104", "103/104", "101/103/104", "105", "106") # All course names
Hours <- c(8, 24, 24, 24, 40, 64, 4, 40) # How many hours each course lasts (assuming an 8-hour workday)
Lesson.Hours <- data.frame(Lesson.Type, Hours) # Create data frame
Income <- Income %>% left_join(Lesson.Hours, by = 'Lesson.Type') # Join hours per course to the Income sheet
Income$Hours <- ifelse(Income$Event.Type == "Charter", as.numeric(Income$Charter.Length), Income$Hours) # Populates the length of charters
Income$Lesson.Type <- ifelse(!is.na(Income$Charter.Length), NA, Income$Lesson.Type) # Changes charter type data in lesson type column to NA
Income <- Income %>% mutate(Hourly.Rate = Total / Hours) # Creates column to calculate hourly rate

# Change data types to factor
Income$Lesson.Type <- as.factor(Income$Lesson.Type)
Income$Charter.Type <- as.factor(Income$Charter.Type)
Income$Event.Type <- as.factor(Income$Event.Type)

# Calculations ----------------------------------------------------------

# Create table showing number of charters/lessons, total income, number of hours spent, and hourly income for
# lessons and charters each year from 2019-2022
Charters_Lessons_Years <- Income %>%
  select(Year, Event.Type, Total, Hours) %>%
  group_by(Year, Event.Type) %>%
  summarise(Number.of.Events = n(), Total.Income = sum(Total), Total.Hours = sum(Hours)) %>%
  mutate(Hourly.Income = round(Total.Income / Total.Hours, 2))

# Calculate total income earned from 2019-2022
Four_Year_Total_Income <- sum(Charters_Lessons_Years$Total.Income)

# Calculate total number of hours worked from 2019-2022
Four_Year_Total_Hours <- sum(Charters_Lessons_Years$Total.Hours)

# Calculate total average hourly income from 2019-2022
```

```
Four_Year_Total_Hourly_Income <- mean(Charters_Lessons_Years$Hourly.Income)

# Calculate totals from lessons in 2019-2022
Lessons_Only <- Charters_Lessons_Years %>%
  filter(as.character(Event.Type) == "Lesson")
Four_Year_Lesson_Income = sum(Lessons_Only$Total.Income) # Total income
Four_Year_Lesson_Hours = sum(Lessons_Only$Total.Hours) # Total hours worked
Four_Year_Lesson_Hourly_Income = mean(Lessons_Only$Hourly.Income) # Total average hourly income

# Calculate total from charters in 2019-2022
Charters_Only <- Charters_Lessons_Years %>%
  filter(as.character(Event.Type) == "Charter")
Four_Year_Charter_Income = sum(Charters_Only$Total.Income) # Total income
Four_Year_Charter_Hours = sum(Charters_Only$Total.Hours) # Total hours worked
Four_Year_Charter_Hourly_Income = mean(Charters_Only$Hourly.Income) # Total average hourly income

# Plots ----------------------------------------------------------------------
# Plot income from lessons and charters side-by-side each year
ggplot(Charters_Lessons_Years, aes(x = as.factor(Year), y = Total.Income, fill = Event.Type)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Year", y = "Income", title = "Figure 1", subtitle = "Income Earned Each Year") +
  scale_fill_manual(values = c("cyan4", "darkgoldenrod1"), labels = c("Charters", "Lessons") ,name = NULL) +
  scale_y_continuous(breaks = seq(0, 100000, 10000), labels = label_number(prefix = "$", suffix = "K", scale = 1e-3))

# Plot hours worked from lesson and charters side-by-side each year
ggplot(Charters_Lessons_Years, aes(x = as.factor(Year), y = Total.Hours, fill = Event.Type)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Year", y = "Hours Worked", title = "Figure 2", subtitle = "Hours Worked Each Year") +
  scale_fill_manual(values = c("cyan4", "darkgoldenrod1"), labels = c("Charters", "Lessons") ,name = NULL) +
  scale_y_continuous(breaks = seq(0, 3000, 500))

# Plot hourly income from charters and lessons side-by-side each year
ggplot(Charters_Lessons_Years, aes(x = as.factor(Year), y = Hourly.Income, fill = Event.Type)) +
  geom_bar(stat = "identity", position = "dodge") +
```

```
  labs(x = "Year", y = "Hourly Income", title = "Figure 3", subtitle = "Hourly Income Each Year") +
  scale_fill_manual(values = c("cyan4", "darkgoldenrod1"), labels = c("Charters", "Lessons") ,name = NULL) +
  scale_y_continuous(breaks = seq(0, 100, 10), labels = label_number(prefix = "$"))

# Future Projections ----------------------------------------------------
# Calculate potential income from each year if hours had been spent doing charters instead of lessons
Potential.Income <- data.frame(c(Charters_Lessons_Years[1,6] * Charters_Lessons_Years[2,5] + Charters_Lessons_Years[1,4],
                                 Charters_Lessons_Years[3,6] * Charters_Lessons_Years[4,5] + Charters_Lessons_Years[3,4],
                                 Charters_Lessons_Years[5,6] * Charters_Lessons_Years[6,5] + Charters_Lessons_Years[5,4],
                                 Charters_Lessons_Years[7,6] * Charters_Lessons_Years[8,5] + Charters_Lessons_Years[7,4]))
Potential.Income <- pivot_longer(Potential.Income, everything(), names_to = "Temp", values_to = "Potential.Income")
Potential.Income <- Potential.Income %>%
  select(-Temp)
Four_Year_Potential_Total_Income <- sum(Potential.Income) # Calculate total potential income from 2019-2022

# Calculate hour many hours would have had to been spent doing charters to earn the same income based on hourly charter income
Potential.Hours <- data.frame(c((Charters_Lessons_Years[1,4] + Charters_Lessons_Years[2,4]) / Charters_Lessons_Years[1,6],
                                (Charters_Lessons_Years[3,4] + Charters_Lessons_Years[4,4]) / Charters_Lessons_Years[3,6],
                                (Charters_Lessons_Years[5,4] + Charters_Lessons_Years[6,4]) / Charters_Lessons_Years[5,6],
                                (Charters_Lessons_Years[7,4] + Charters_Lessons_Years[8,4]) / Charters_Lessons_Years[7,6]))
Potential.Hours <- pivot_longer(Potential.Hours, everything(), names_to = "Temp", values_to = "Potential.Hours")
Potential.Hours <- Potential.Hours %>%
  select(-Temp)
Four_Year_Potential_Total_Hours <- sum(Potential.Hours) # Calculate total potential hours from 2019-2022

# Isolate year, income actually earned, and potential income
Income_Potential <- Charters_Lessons_Years %>%
  select(Year, Event.Type, Total.Income) %>%
  summarise(Actual.Income = sum(Total.Income))
Income_Potential <- cbind(Income_Potential, Potential.Income)
Income_Potential <- Income_Potential %>%
  pivot_longer(!Year, names_to = "Income.Type", values_to = "Income")

# Isolate year, hours actually worked, and reduced hours needed
```

```
Hours_Potential <- Charters_Lessons_Years %>%
  select(Year, Event.Type, Total.Hours) %>%
  summarise(Actual.Hours = sum(Total.Hours))
Hours_Potential <- cbind(Hours_Potential, Potential.Hours)
Hours_Potential <- Hours_Potential %>%
  pivot_longer(!Year, names_to = "Income.Type", values_to = "Hours")

# Plot actual income earned and potential income that could have been earned spending that time doing charters
ggplot(Income_Potential, aes(x = as.factor(Year), y = Income, fill = Income.Type)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Year", y = "Income", title = "Figure 4", subtitle = "Actual vs. Potential Income Each Year") +
  scale_fill_manual(values = c("cyan4", "darkgoldenrod1"), labels = c("Actual Income", "Potential Income") ,name = NULL) +
  scale_y_continuous(breaks = seq(0, 250000, 25000), labels = label_number(prefix = "$", suffix = "K", scale = 1e-3))

# Plot actual numbers of hours worked and how many hours would have had to have been spent doing just charters to earn the same amout
ggplot(Hours_Potential, aes(x = as.factor(Year), y = Hours, fill = Income.Type)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Year", y = "Hours Worked", title = "Figure 5", subtitle = "Actual vs. Potential Hours Worked Each Year") +
  scale_fill_manual(values = c("cyan4", "darkgoldenrod1"), labels = c("Actual Hours Worked", "Potential Hours Worked") ,name = NULL) +
  scale_y_continuous(breaks = seq(0, 4000, 500))

# Calculate income in 2024 by working the same number of hours as in 2022
Potential_Income_2024 = (Charters_Lessons_Years[7,5] + Charters_Lessons_Years[8,5]) * Charters_Lessons_Years[7,6]

# Calculate hours needed to work in order to make same income as in 2022
Potential_Hours_2024 = (Charters_Lessons_Years[7,4] + Charters_Lessons_Years[8,4]) / Charters_Lessons_Years[7,6]

# Percentage difference between average hourly income for lessons and charters in 2019-2022
Hourly_Income_Pct_Difference = round(100 * (Four_Year_Charter_Hourly_Income - Four_Year_Lesson_Hourly_Income) /
Four_Year_Lesson_Hourly_Income, 2)

# Projected percentage difference of income from 2022 to 2024 if hours are kept the same
Income_2022_To_2024_Pct_Difference = round(100 * (Potential_Income_2024 - Income_Potential[7,3]) / Income_Potential[7,3], 2)
```

```
# Projected percentage difference of work hours from 2022 to 2024 to maintain the same income
Hours_2022_To_2024_Pct_Difference = round(100 * (Potential_Hours_2024 - Hours_Potential[7,3]) / Hours_Potential[7,3], 2)

Four_Year_Total_Income
Four_Year_Lesson_Income
Four_Year_Charter_Income
Four_Year_Total_Hours
Four_Year_Lesson_Hours
Four_Year_Charter_Hours
Four_Year_Total_Hourly_Income
Four_Year_Lesson_Hourly_Income
Four_Year_Charter_Hourly_Income
Four_Year_Potential_Total_Income
Four_Year_Potential_Total_Hours
Potential_Income_2024
Potential_Hours_2024
Hourly_Income_Pct_Difference
Income_2022_To_2024_Pct_Difference
Hours_2022_To_2024_Pct_Difference

view(Income)
view(Charters_Lessons_Years)
```