

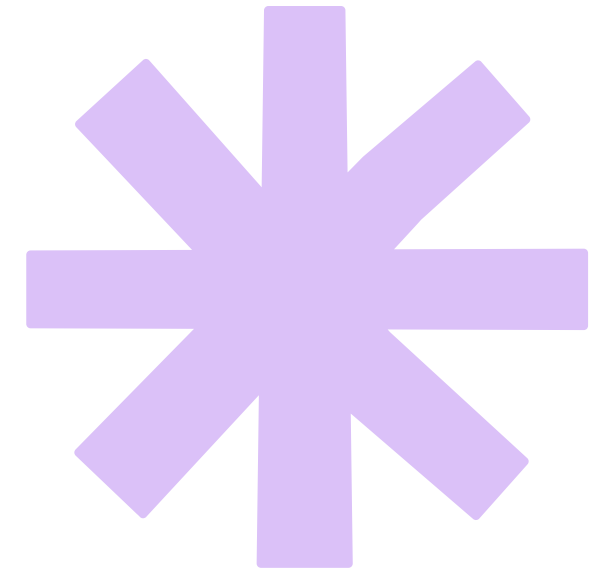
WIC WEBSITE MEETING



**Website meetings do not
count towards attendance pts

Today

- ✓ Announcements
- ✓ Semester Agenda
- ✓ Data Analysis in R



Future Meetings

- 2/5: Databases by Anna Ross
- 2/19: Power Automate by Sabeen Ahmad
- 3/5: Javascript by Christina Sorensen
- 3/26: Cloud Deployment by Samantha Adorno
- 4/9: Client vs Server Interaction by Tanu Sakaray
- 4/23: Open Q&A Session

Audrey Pan

Computer Science Junior

Incoming SWE Intern @ Cargill

WIC Vice President



Intro to R

What is R?

- A programming language and software environment for statistical computing and data visualization.

Why Learn R?

- Built for data analysis and statistics.
- Excels in data visualization with tools like ggplot2.
- Widely used in research, academia, and industry.

What Makes R Unique?

- Thousands of specialized packages for diverse fields (e.g., healthcare, finance, biology).
- Built-in datasets for quick exploration and practice.
- Easily generates reports and reproducible research with R Markdown.

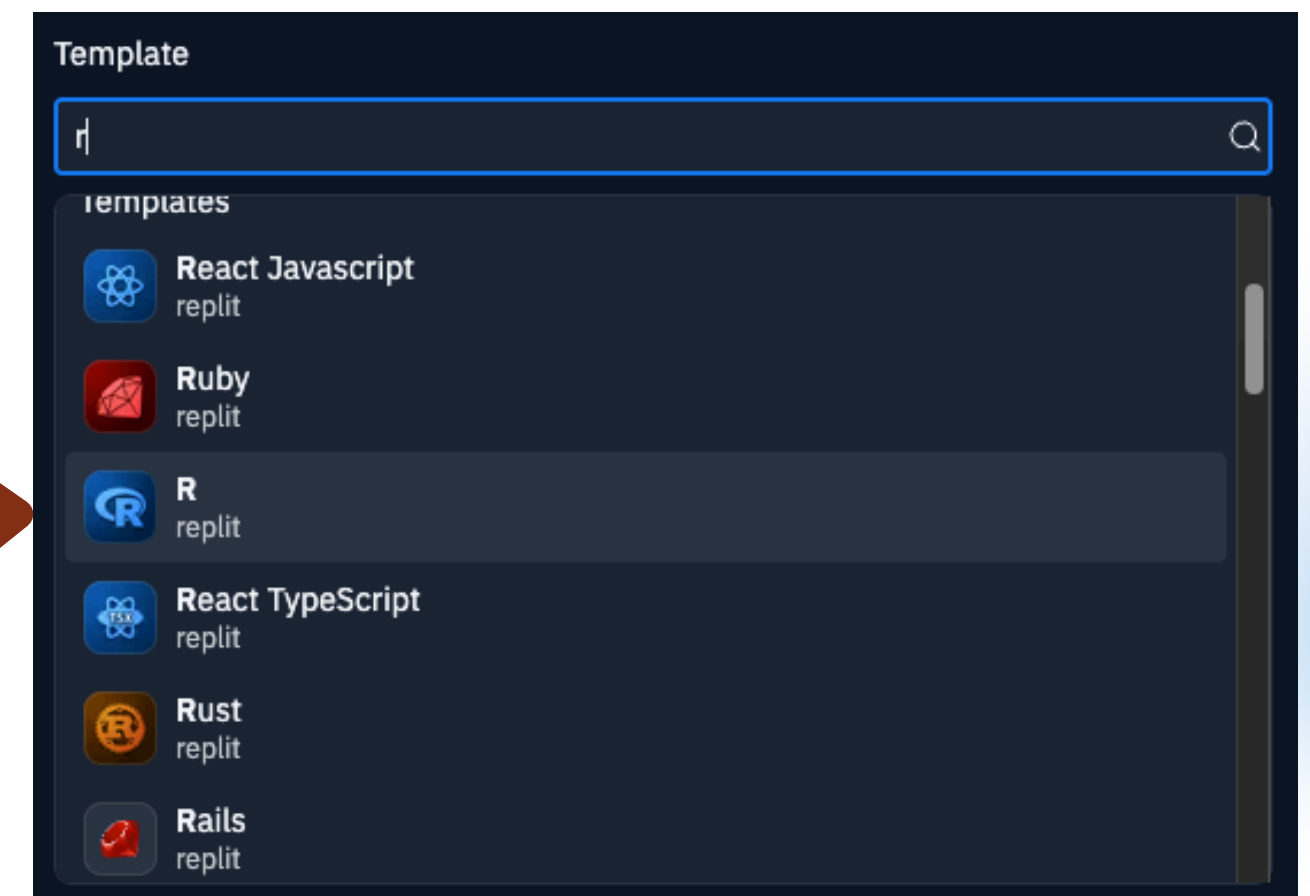
Intro to Mini Project

We will use the built-in dataset in R called “gapminder” that provides data on global development trends, focusing on health, economy, and population. It has data on country, continent, year, lifeExp, pop, gdpPercap.

1. Setup
2. Filtering Data
3. Visualization
4. Statistical Analysis (simple)

Initial Setup

1. Search up replit.com
2. Quickly create a free account
3. Click on the "Create Repl" button on the top left
4. Under Template, search up "R" and select it



5. Title the project as "Mini Data Analysis with R"

Part 1 - Setup

```
1 #Part 1 - Setup and Basic Syntax
2 # 1a. Run this code so we can install and load packages
3
4 # 1b. Comment out all the "install.packages()" after you finish running it for the first time
5 install.packages("tidyverse")
6 install.packages("datasets")
7 install.packages("gapminder") #gapminder is the dataset we will work with
8
9 library(dplyr)
10 library(gapminder)
11
12 # 1c. What do you think "<-" does?
13 data <- gapminder
14
15 # 1d. Run just data and observe the output. You will need this for 1e.
16 data
17
18 # 1e. Uncomment head(data) and run the code.
19 # After you're done, comment data, line16, and head(data), line20, out.
20 #head(data)
```

Discuss:

Compare the output of ``data`` and ``head(data)``.
What does the function `head()` do?

Part 2: Filter

Key Concepts & Functions:

- **Variables:** `nameOfVariable <- whatGoesIntoVar` (Ex. `total <- 32`)
- **print():** Same thing as in Python
- **Pipes (%>%):** simple way for us to pass the output of a function to the next one. This helps keeps our code readable and easy to interpret.
- **filter():** selects rows based on conditions we put

Part 2: Filter and Summarize Data

Task:

Filter the dataset where the year is 2007 and the continent is "Asia".

Here's an example of filter and pipes using the mtcars dataset. HINT: the dataset we are using is called gapminder

HINT: the dataset we are using is called gapminder

Variable Name



```
#Example for Part 2  
# Filtering rows in the mtcars dataset  
library(dplyr)  
  
# Filter cars with 6 cylinders and 4 gears  
filtered_cars <- mtcars %>%  
  filter(cyl == 6 & gear == 4)  
  
# View the first few rows of the filtered data  
filtered_cars
```


Part 3 - Visualize Data

- Replit does not support any of the packages for data visualization.

```
#Part 3 - This part does not work on Replit  
  
#let's create a scatter plot of GDP per capital (gdpPercap) vs life expectancy (lifeExp)  
ggplot(data, aes(x=gdpPercap,y=lifeExp, color=continent)) + geom_point()  
  
ggplot(gapminder, aes(x = gdpPercap, y = lifeExp, color = continent)) +  
  geom_point() +  
  scale_x_log10() +  
  labs(title = "Life Expectancy vs. GDP (2007)",  
        x = "GDP Per Capita (log scale)",  
        y = "Life Expectancy")
```

Part 4 - Statistical Analysis

Task:

Calculate the mean and median life expectancy (lifeExp) from the year 1957. Save the results as variables and print them.

Example code using
gdpPerCap instead

```
# Calculate average and median GDP per capita in 2007
library(dplyr)

# Filter data for the year 2007
gap_2007 <- gapminder %>%
  filter(year == 2007)

# Calculate average and median GDP per capita
avg_gdp <- mean(gap_2007$gdpPerCap)
median_gdp <- median(gap_2007$gdpPerCap)

# Print the results
print(avg_gdp)
print(median_gdp)
```

Wrap Up

What We Learned:

- **Introduction to R:** What R is and why it's useful.
- **Basic Syntax:** Variables, pipes (`%>%`), and functions.
- **Data Manipulation:** Filtering (`filter()`)
- **Data Visualization:** Creating plots with `ggplot2`, layering, and transforming scales.
- **Statistical Analysis:** Basic descriptive stats