```r
# Create a blank visualization

votes <- read.csv("votes.csv")

ggplot()
```

```r
# Supply data

votes <- read.csv("votes.csv")

ggplot(votes)
```

```r
# Add first geometry

votes <- read.csv("votes.csv")

ggplot(votes) +
  geom_col()
```

```r
# Add x and y aesthetics

votes <- read.csv("votes.csv")

ggplot(votes, aes(x = candidate, y = votes)) +
  geom_col()
```

```r
# Adjust y scale

votes <- read.csv("votes.csv")

ggplot(votes, aes(x = candidate, y = votes)) +
  geom_col() +
  scale_y_continuous(limits = c(0, 250))
```

```r
# Add labels

votes <- read.csv("votes.csv")

ggplot(votes, aes(x = candidate, y = votes)) +
  geom_col() +
  scale_y_continuous(limits = c(0, 250)) +
  labs(
    x = "Candidate",
    y = "Votes",
    title = "Election Results"
  )
```

```r
# Add fill aesthetic mapping for geom_col

votes <- read.csv("votes.csv")

ggplot(votes, aes(x = candidate, y = votes)) +
  geom_col(aes(fill = candidate)) +
  scale_y_continuous(limits = c(0, 250)) +
  labs(
    x = "Candidate",
    y = "Votes",
    title = "Election Results"
  )
```

```r
# Use viridis scale to design for color blindness

votes <- read.csv("votes.csv")

ggplot(votes, aes(x = candidate, y = votes)) +
  geom_col(aes(fill = candidate)) +
  scale_fill_viridis_d("Candidate") +
  scale_y_continuous(limits = c(0, 250)) +
  labs(
    x = "Candidate",
    y = "Votes",
    title = "Election Results"
  )
```

```r
# Adjust ggplot theme

votes <- read.csv("votes.csv")

ggplot(votes, aes(x = candidate, y = votes)) +
  geom_col(aes(fill = candidate)) +
  scale_fill_viridis_d("Candidate") +
  scale_y_continuous(limits = c(0, 250)) +
  labs(
    x = "Candidate",
    y = "Votes",
    title = "Election Results"
  ) +
  theme_classic()
```

```r
# Save file

votes <- read.csv("votes.csv")

p <- ggplot(votes, aes(x = candidate, y = votes)) +
  geom_col(aes(fill = candidate)) +
  scale_fill_viridis_d("Candidate") +
  scale_y_continuous(limits = c(0, 250)) +
  labs(
    x = "Candidate",
    y = "Votes",
    title = "Election Results"
  ) +
  theme_classic()

ggsave(
  "votes.png",
  plot = p,
  width = 1200,
  height = 900,
  units = "px"
)
```

```r
# Introduce geom_point

load("candy.RData")

ggplot(
  candy,
  aes(x = price_percentile, y = sugar_percentile)
) +
  geom_point()
```

```r
# Add labels and theme

load("candy.RData")

ggplot(
  candy,
  aes(x = price_percentile, y = sugar_percentile)
) +
  geom_point() +
  labs(
    x = "Price",
    y = "Sugar",
    title = "Price and Sugar"
  ) +
  theme_classic()
```

```r
# Introduce geom_jitter

ggplot(
  candy,
  aes(x = price_percentile, y = sugar_percentile)
) +
  geom_jitter() +
  labs(
    x = "Price",
    y = "Sugar",
    title = "Price and Sugar"
  ) +
  theme_classic()
```

```r
# Introduce size and color aesthetic

ggplot(
  candy,
  aes(x = price_percentile, y = sugar_percentile)
) +
  geom_jitter(
    color = "darkorchid",
    size = 2
  ) +
  labs(
    x = "Price",
    y = "Sugar",
    title = "Price and Sugar"
  ) +
  theme_classic()
```

```r
# Introduce point shape and fill color

ggplot(
  candy,
  aes(x = price_percentile, y = sugar_percentile)
) +
  geom_jitter(
    color = "darkorchid",
    fill = "orchid",
    shape = 21,
    size = 2
  ) +
  labs(
    x = "Price",
    y = "Sugar",
    title = "Price and Sugar"
  ) +
  theme_classic()
```

```r
# Visualize with geom_point

load("anita.RData")

ggplot(anita, aes(x = timestamp, y = wind)) +
  geom_point()
```

```r
# Introduce geom_line

load("anita.RData")

ggplot(anita, aes(x = timestamp, y = wind)) +
  geom_line()
```

```r
# Combine geom_line and geom_point

load("anita.RData")

ggplot(anita, aes(x = timestamp, y = wind)) +
  geom_line() +
  geom_point(color = "deepskyblue4")
```

```r
# Experiment with geom_line and geom_point aesthetics

load("anita.RData")

ggplot(anita, aes(x = timestamp, y = wind)) +
  geom_line(
    linetype = 1,
    linewidth = 0.5
  ) +
  geom_point(
    color = "deepskyblue4",
    size = 2
  )
```

```r
# Add labels and adjust theme

load("anita.RData")

ggplot(anita, aes(x = timestamp, y = wind)) +
  geom_line(
    linetype = 1,
    linewidth = 0.5
  ) +
  geom_point(
    color = "deepskyblue4",
    size = 2
  ) +
  labs(
    y = "Wind Speed (Knots)",
    x = "Date",
    title = "Hurricane Anita"
  ) +
  theme_classic()
```

```r
# Add horizontal line to demarcate hurricane status

load("anita.RData")

ggplot(anita, aes(x = timestamp, y = wind)) +
  geom_line(
    linetype = 1,
    linewidth = 0.5
  ) +
  geom_point(
    color = "deepskyblue4",
    size = 2
  ) +
  geom_hline(
    linetype = 3,
    yintercept = 64
  ) +
  labs(
    y = "Wind Speed (Knots)",
    x = "Date",
    title = "Hurricane Anita"
  ) +
  theme_classic()
```