Coding tasks: melt & facet_wrap for ggplot

Melina Klostermann

29 October, 2020

This is the coding task for 29.10. Please do not work on it more then half a day and leave all left over exercises. On the second page I wrote some tips, but you are also welcome to solve the tasks differently.

Example Data Set

```
library(datasets)
library(ggplot2)
library(tidyverse)

data("airquality")
head(airquality)
```
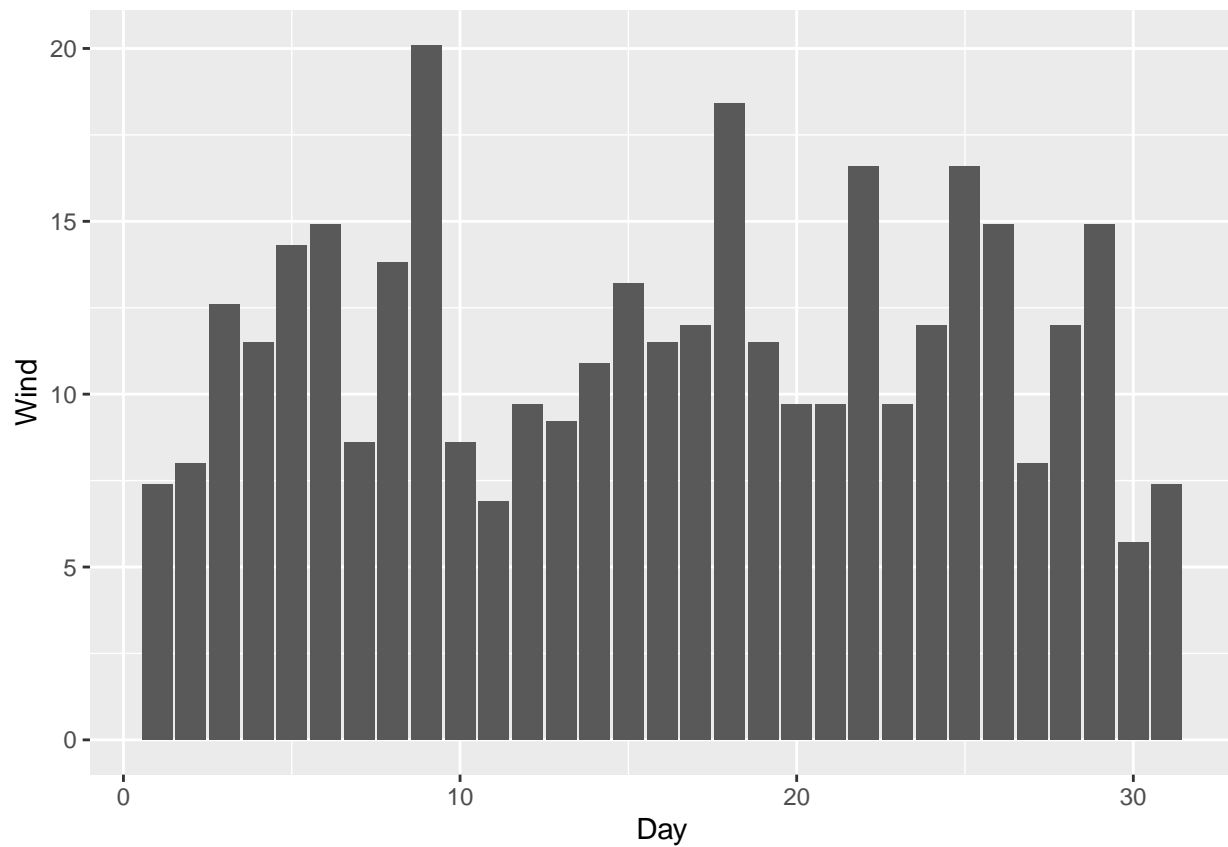
```
##   Ozone Solar.R Wind Temp Month Day
## 1    41     190  7.4   67     5   1
## 2    36     118  8.0   72     5   2
## 3    12     149 12.6   74     5   3
## 4    18     313 11.5   62     5   4
## 5    NA      NA 14.3   56     5   5
## 6    28      NA 14.9   66     5   6
```

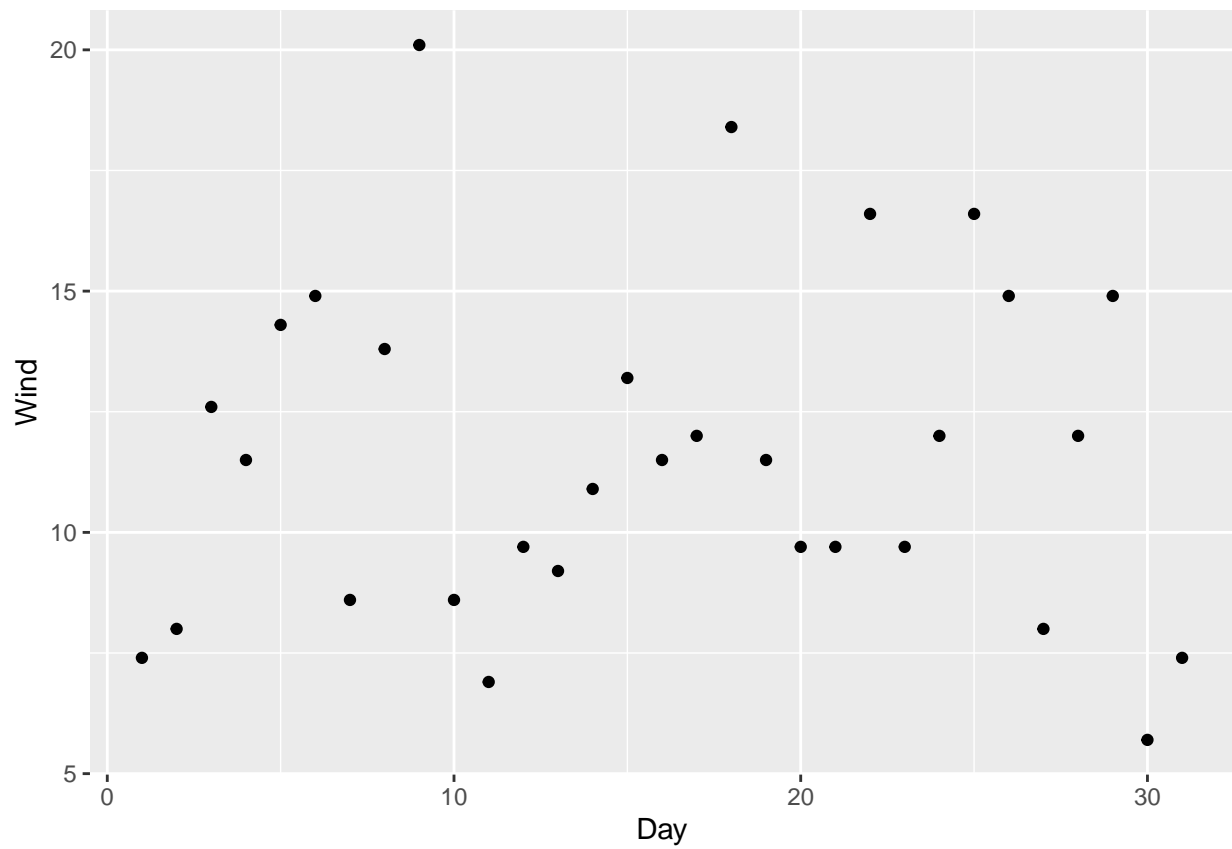## 1. Plot the wind on each day in may (month 5)

```
# histogram
p1 <- ggplot(airquality[airquality$Month==5,], aes(x=Day, y=Wind))+
  geom_col()

p1
```
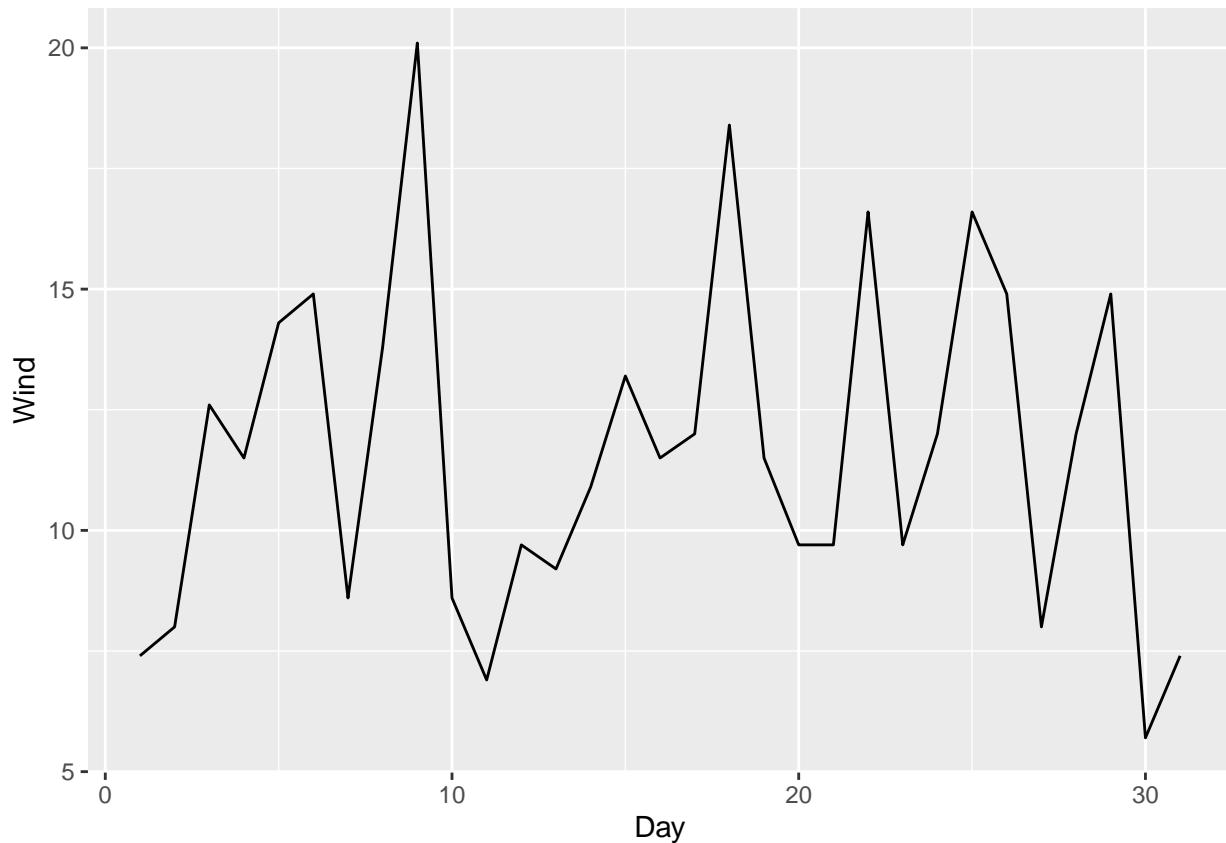
```
# dotplot
p2 <- ggplot(airquality[airquality$Month==5,], aes(x=Day, y=Wind))+
  geom_point()

p2
```

```
# line
p3 <- ggplot(airquality[airquality$Month==5,], aes(x=Day, y=Wind))+
  geom_line()

p3
```

2. Color wind intensities above 10 blue

```r
# mark strong wind with dplyr
airquality <- airquality %>% mutate(strong_wind = case_when(
  Wind > 10 ~ T,
  T~F))

# mark strong wind with base R
airquality$also_strong_wind <- F
airquality[airquality$Wind > 10,]$also_strong_wind <- T


# color in plot
# with color name
p1 <- ggplot(airquality[airquality$Month==5,], aes(x=Day, y=Wind, fill = strong_wind))+
  geom_col()+
  scale_fill_manual(values =c("grey", "blue"))
p1
```
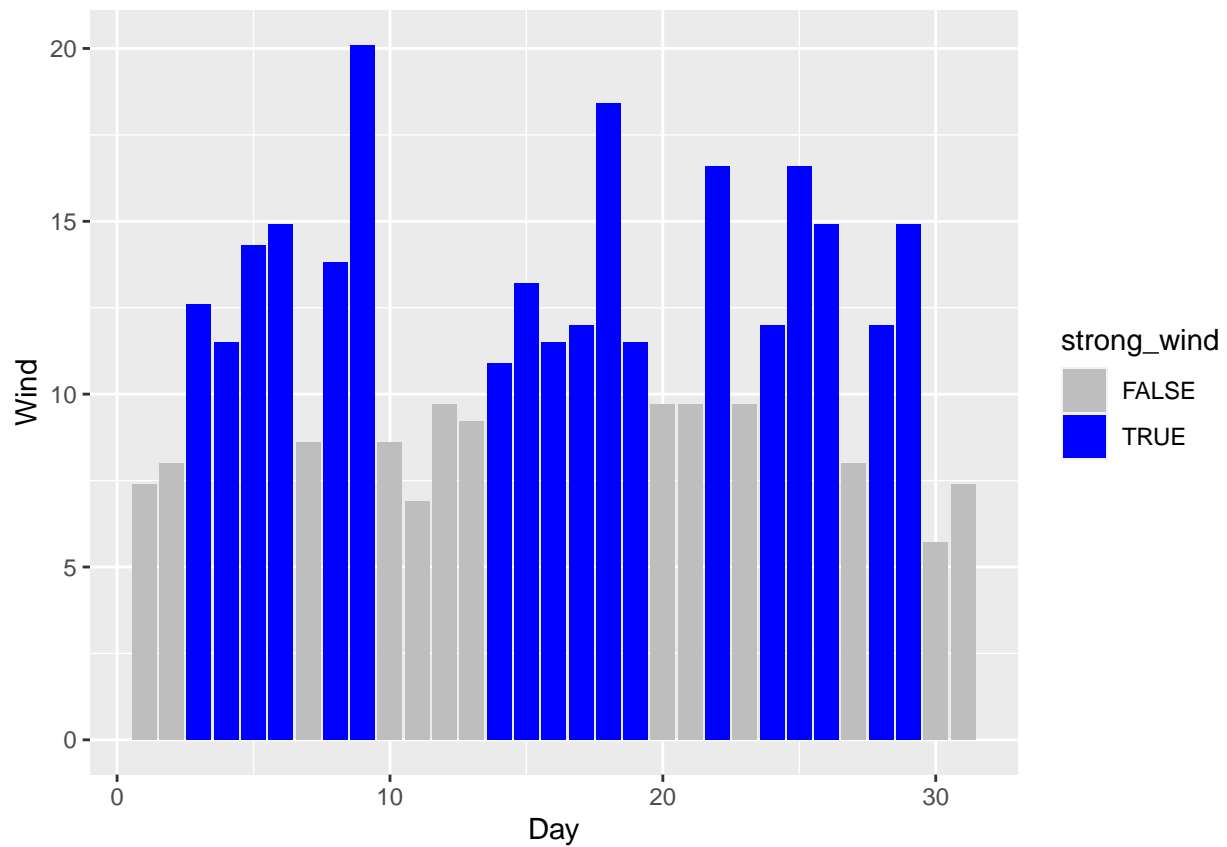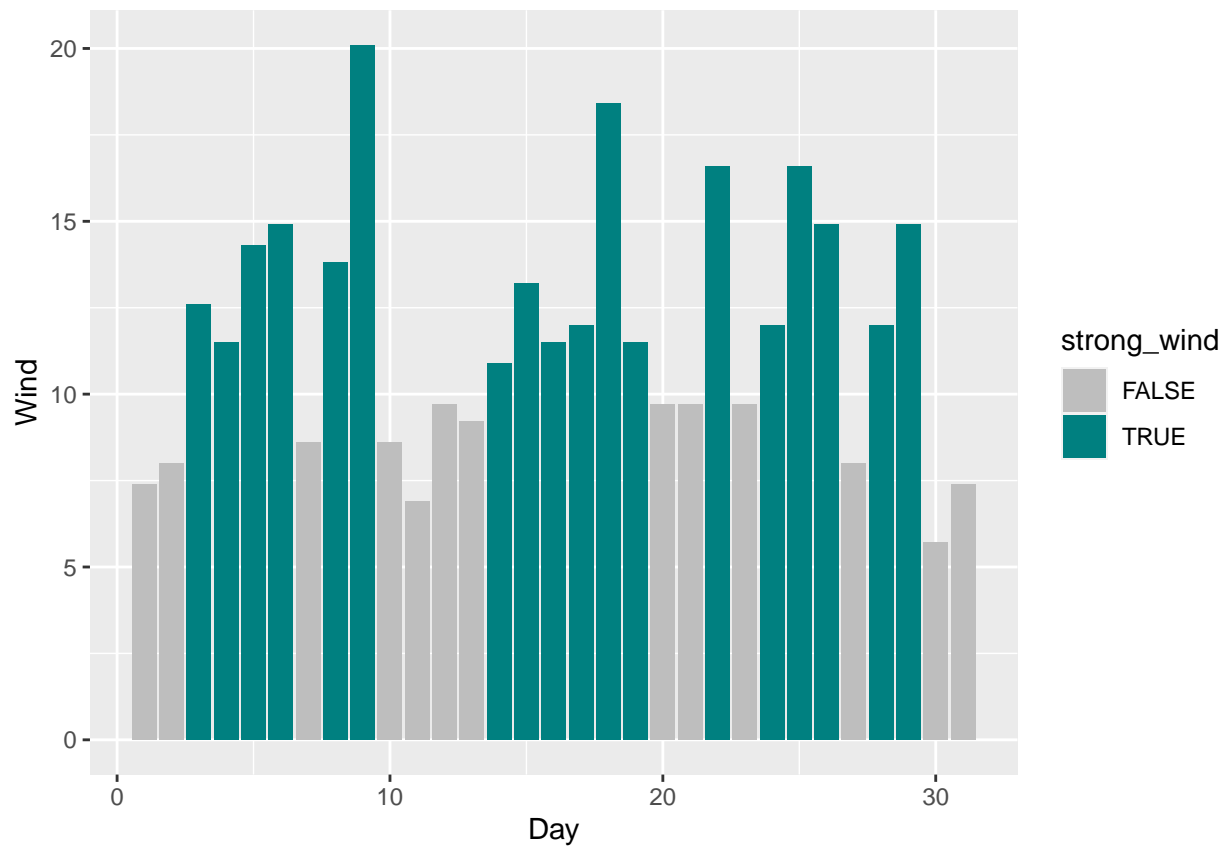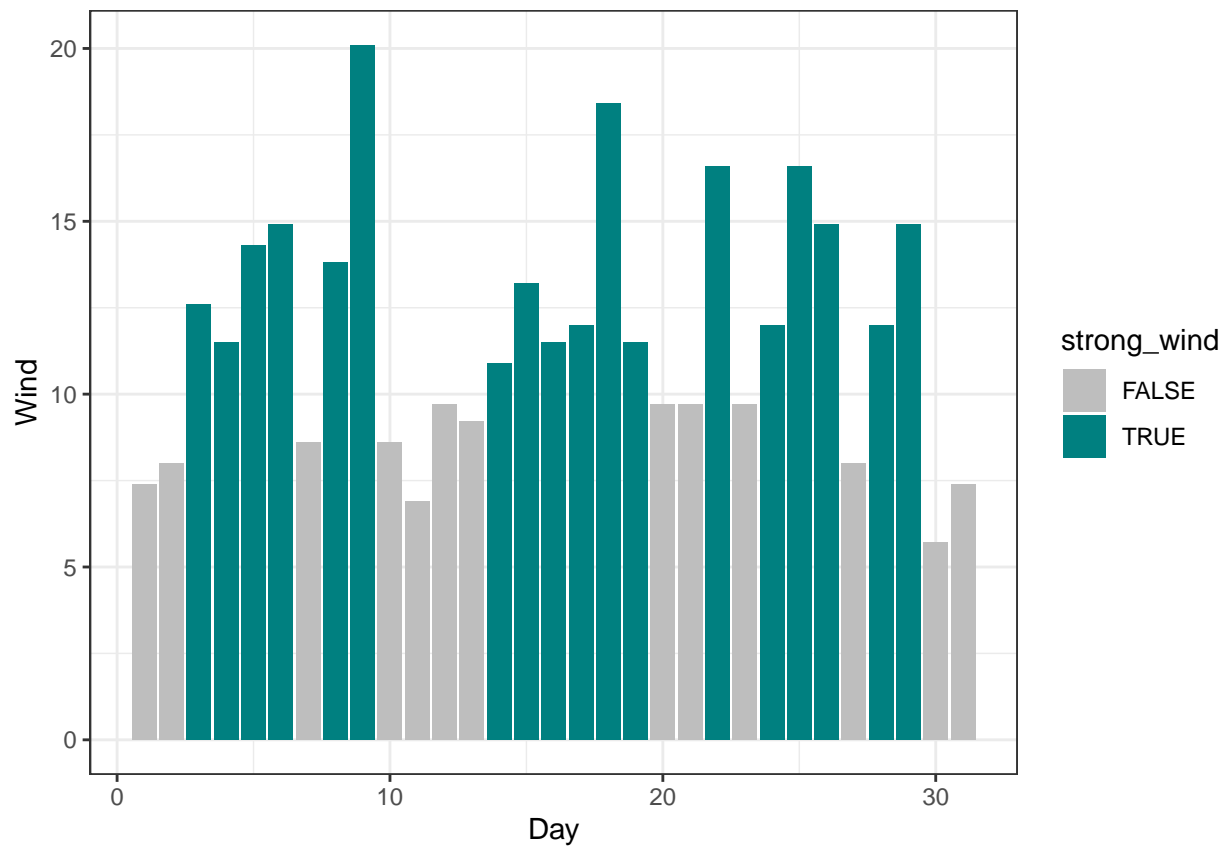
```
# with Hex code
p1 <- ggplot(airquality[airquality$Month==5,], aes(x=Day, y=Wind, fill = strong_wind))+
  geom_col()+
  scale_fill_manual(values =c("grey", "#008080"))
p1
```
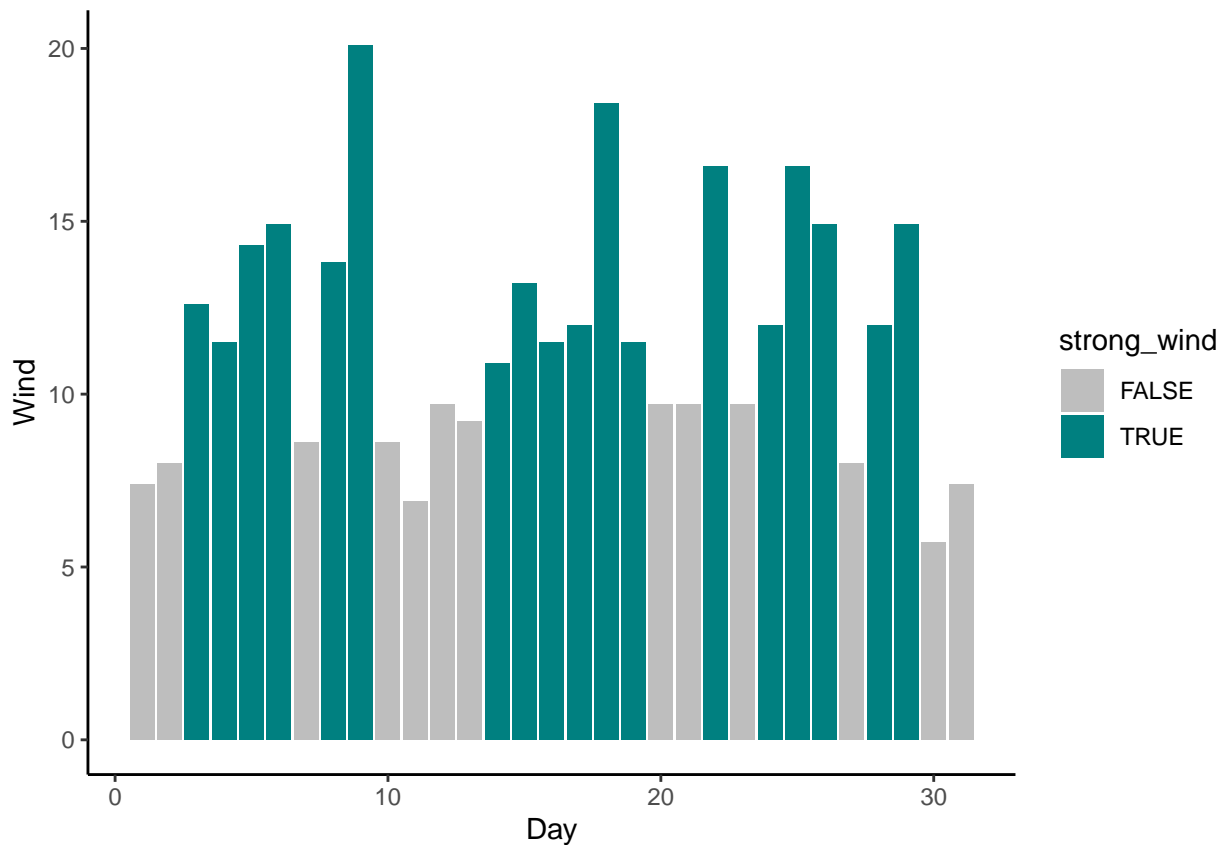
3. Change the theme of the plot

```
# there are many themes to choose from:
p1 + theme_bw()
```

```
p1 + theme_classic()
```

```
# more themes see: https://www.datanovia.com/en/blog/ggplot-themes-gallery/

# make your own theme:
p4 <- ggplot(airquality[airquality$Month==5,], aes(x=Day, y=Wind, fill = strong_wind))+
  geom_col()+
  scale_fill_manual(values =c("yellow", "blue"))+
  theme(
  # get rid of panel grids
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  # Change plot and panel background
  plot.background=element_rect(fill = "gray"),
  panel.background = element_rect(fill = 'black'),
  # Change legend
  legend.position = c(0.6, 0.07),
  legend.direction = "horizontal",
  legend.background = element_rect(fill = "black", color = NA),
  legend.key = element_rect(color = "gray", fill = "black"),
  legend.title = element_text(color = "white"),
  legend.text = element_text(color = "white")
  )

p4
```
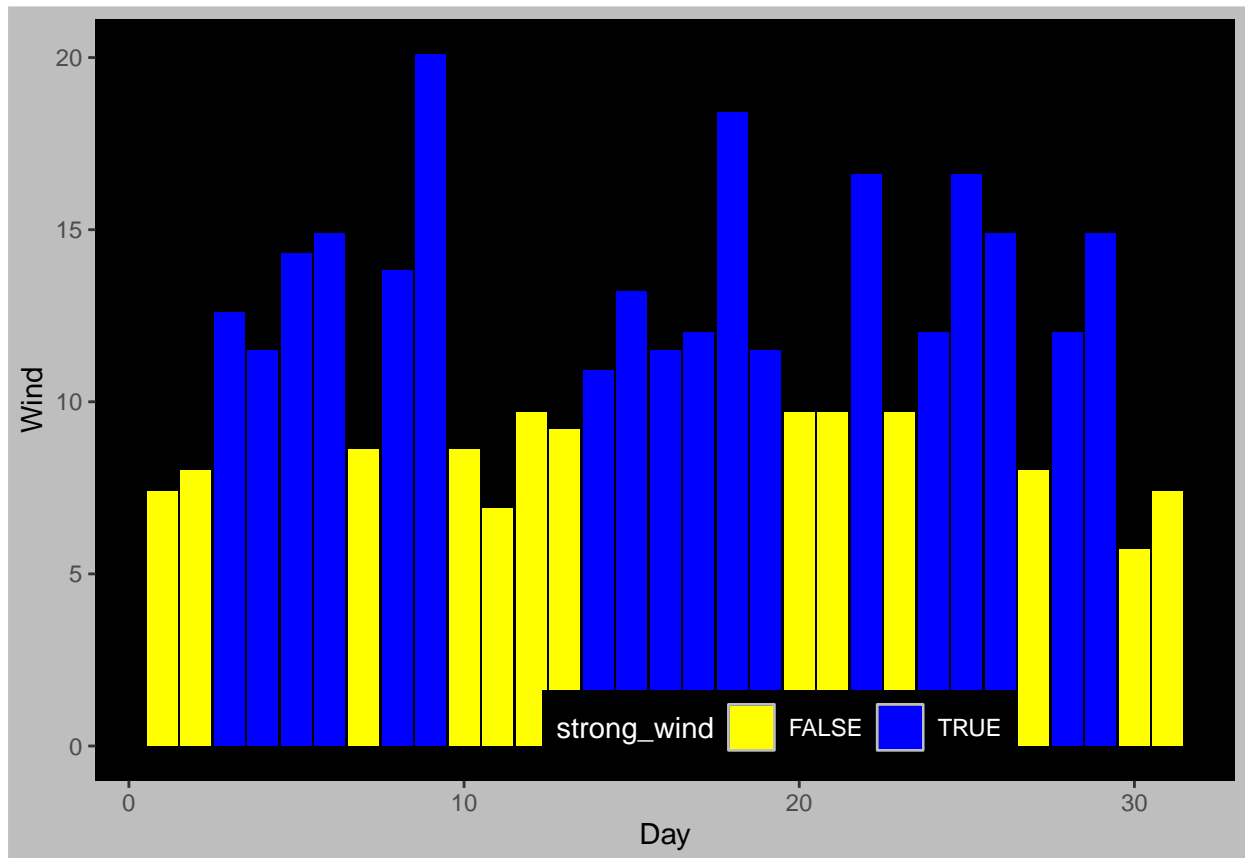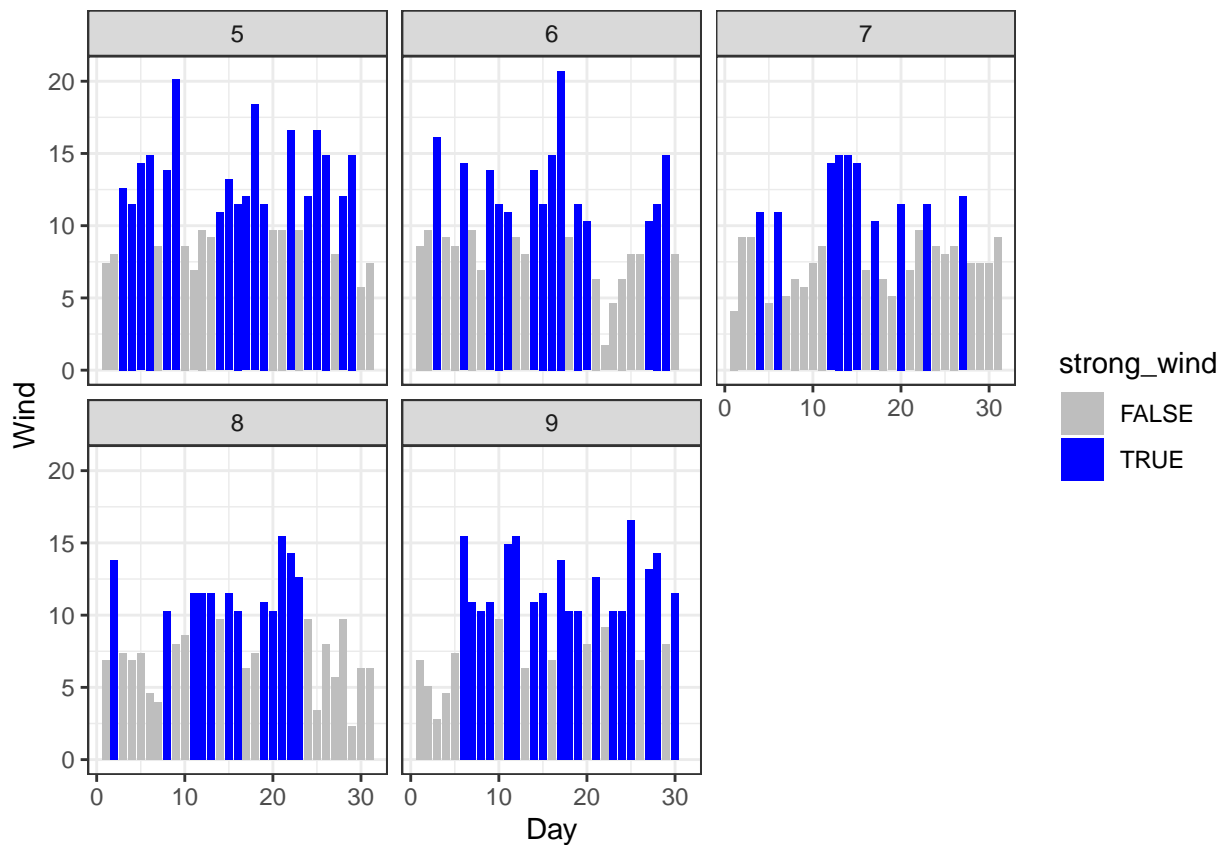
4. Use facet_wrap to plot the wind in each month

```
p1 <- ggplot(airquality, aes(x=Day, y=Wind, fill = strong_wind))+
  geom_col()+
  scale_fill_manual(values =c("grey", "blue"))+
  facet_wrap(~Month)+ # tell facet_wrap which column contains the facets
  theme_bw()

p1
```

5. Plot ozone, solar.r wind and temp for month 5 as facets of the same plot

```r
# you can use melt to get all values for ozone, solar.r wind and temp in one column

airquality_melt <- reshape2::melt(airquality, id.vars=c("Day", "Month", "strong_wind", "also_strong_wind

p5 <- ggplot(airquality_melt[airquality_melt$Month==5,], aes(x=Day, y=messured, fill = air_factor))+
  geom_col()+
  facet_wrap(~air_factor, scales = "free_y")+
  scale_fill_manual(values = c("darkblue", "yellow", "#0000FF", "red"))+
  theme_dark()

p5
```
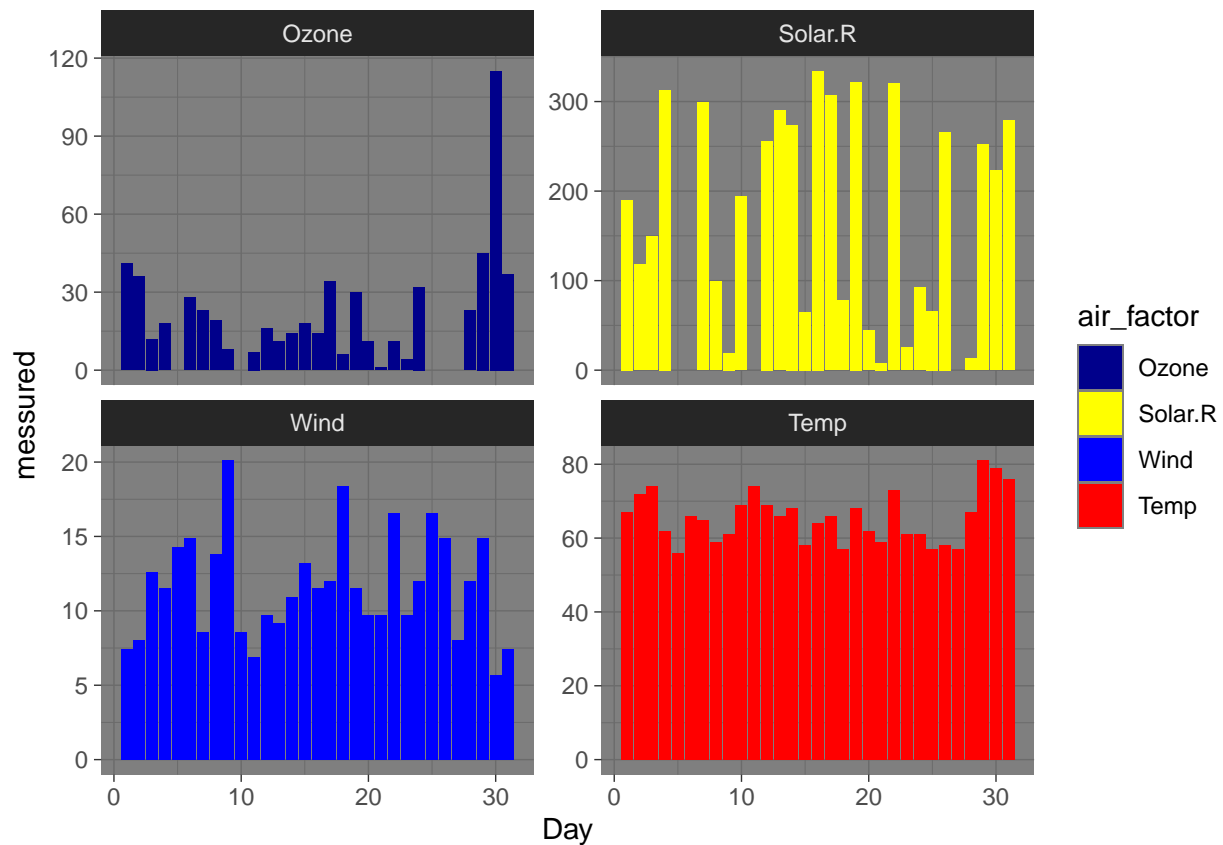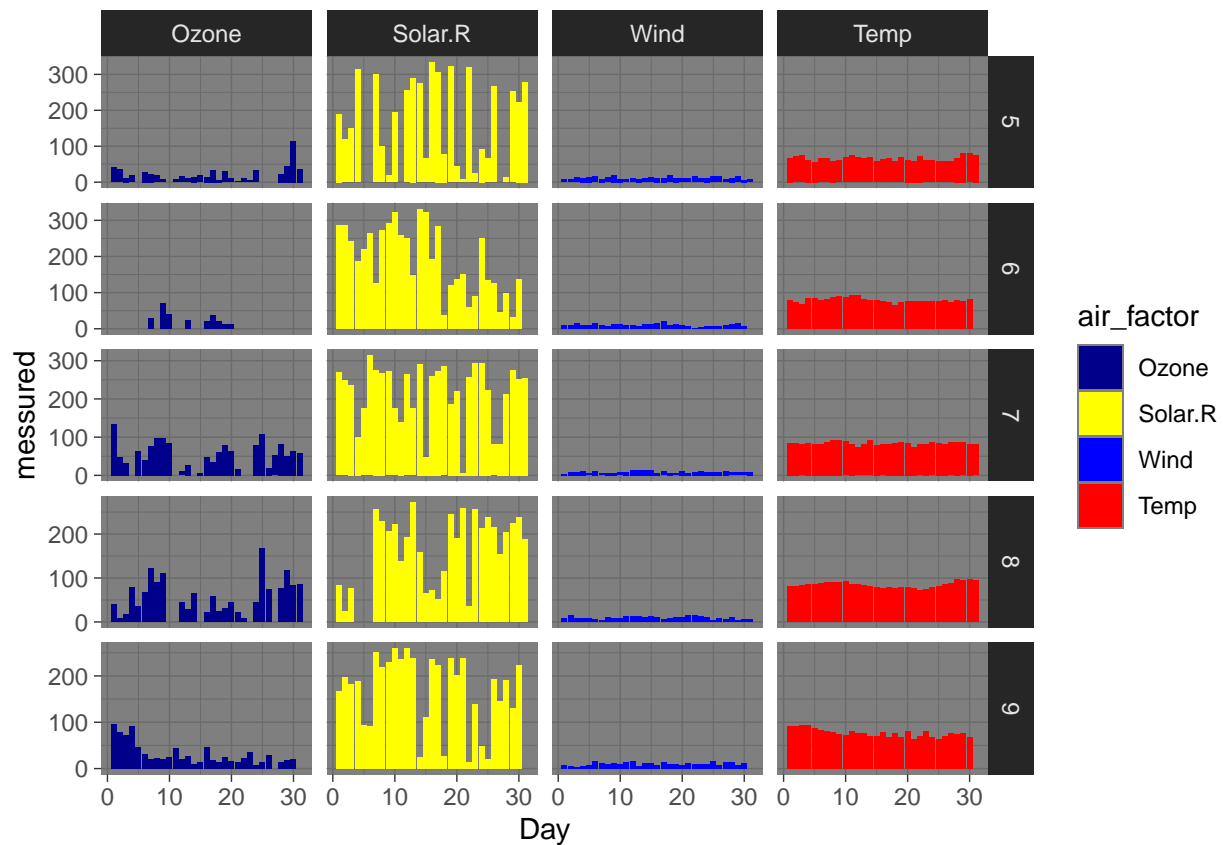
6. Make a plot with subplot for each condition and each month With facet_grid you can split up the plot by two columns.

```r
p5 <- ggplot(airquality_melt, aes(x=Day, y=messured, fill = air_factor))+
  geom_col()+
  facet_grid(Month~air_factor, scales = "free")+
  scale_fill_manual(values = c("darkblue", "yellow", "#0000FF", "red"))+
  theme_dark()

p5
```
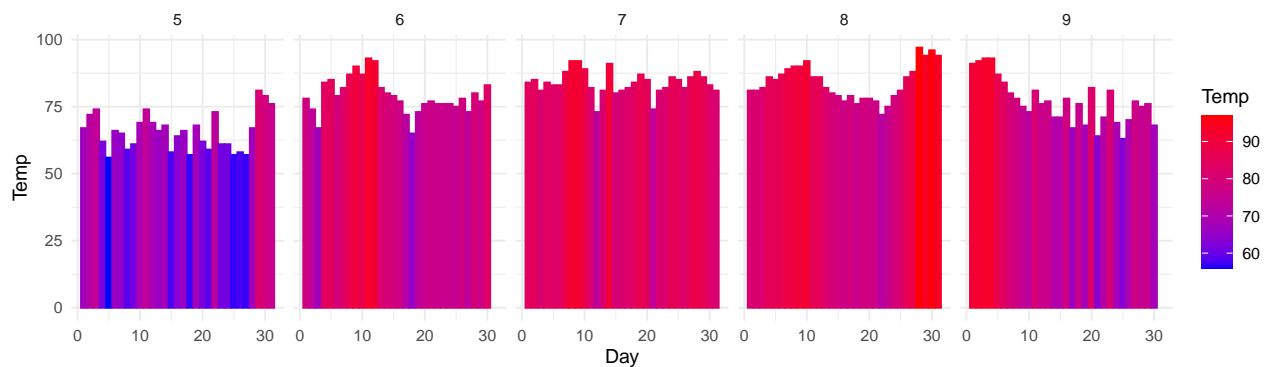
7. Plot the temperatures of each month and color by the temperature (warmest red, coldest blue)

```
p7 <- ggplot(airquality, aes(x=Day, y=Temp, fill = Temp, color = Temp))+
  geom_col()+
  facet_wrap(~Month, ncol = 5)+
  scale_fill_gradient(low="blue", high="red")+
  scale_color_gradient(low="blue", high="red")+
  theme_minimal()
p7
```

Tips

1.

You can use the filter function from dplyr

2.

Make a new column high_wind and use color = high_wind in the aes() of your ggplot

3.

See https://ggplot2.tidyverse.org/reference/ggtheme.html

4.

With facet wrap you can split up one plot into multiple by one variable. The syntax is

```
your_ggplot+
facet_wrap(~colname_to_split)
```

5.

You can use melt from the reshape2 package. It will melt all columns together. Try it out.

Now for the plot you want to keep month and day unmelted. You can specify id.vars which will not be melted. You can use value as y day as x and variable for facet_wrap.

6.

```
your_ggplot+
facet_grid(col1~col2)
```

7.

```
your_ggplot+
scale_colour_gradient()  or scale_fill_gradient()
```