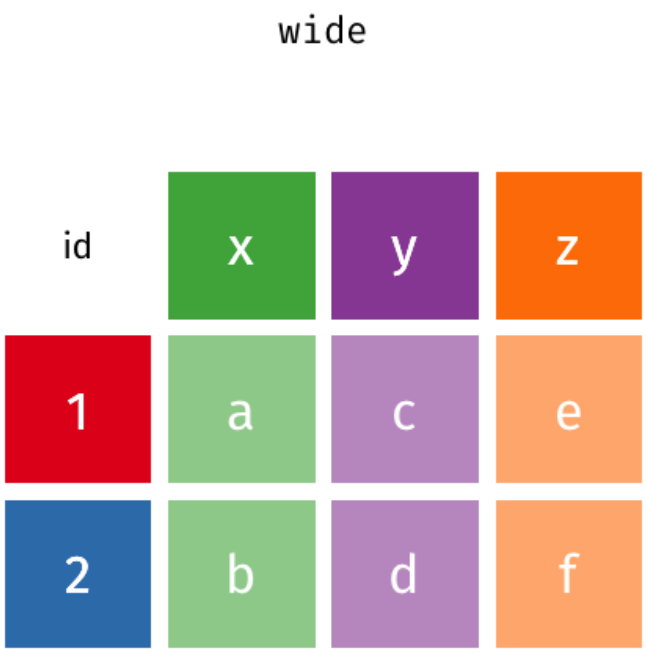
# Data Reshaping

Data Wrangling in R

# Reshaping: wide vs. long data

https://github.com/gadenbuie/tidyexplain/blob/main/images/tidyr-pivoting.gif



# What is wide/long data?

Data is stored *differently* in the tibble.

Wide: has many columns

```
# A tibble: 1 × 4
  State   June_vacc_rate May_vacc_rate April_vacc_rate
  <chr>            <dbl>         <dbl>           <dbl>
1 Alabama          0.516         0.514           0.511
```

Long: column names become data

```
# A tibble: 3 × 3
  State   name            value
  <chr>   <chr>           <dbl>
1 Alabama June_vacc_rate  0.516
2 Alabama May_vacc_rate   0.514
3 Alabama April_vacc_rate 0.511
```

# What is wide/long data?

Wide: multiple columns per individual, values spread across multiple columns

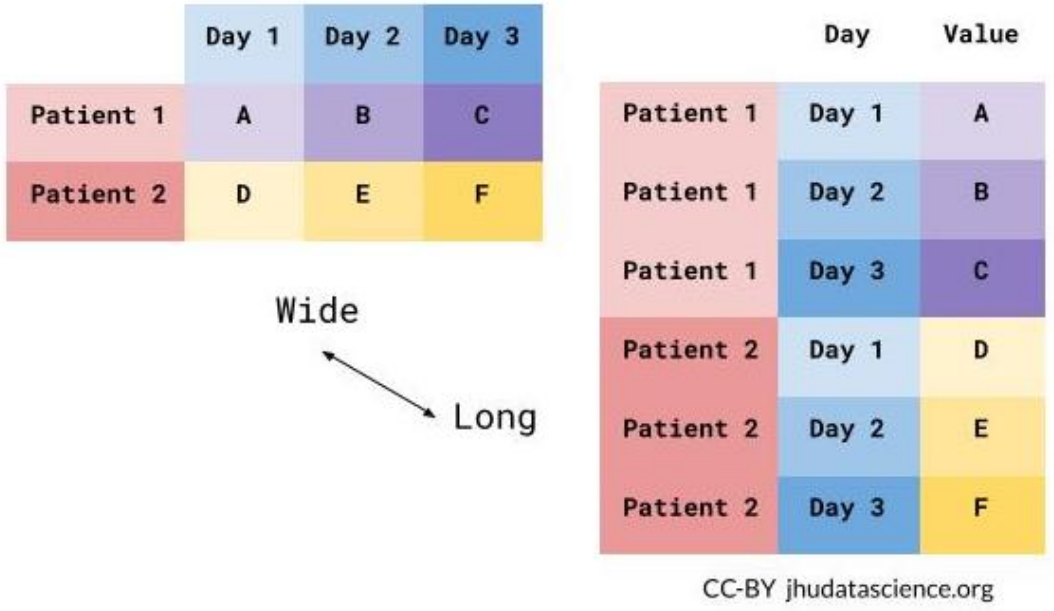
```
# A tibble: 2 × 4
  State   June_vacc_rate May_vacc_rate April_vacc_rate
  <chr>            <dbl>         <dbl>           <dbl>
1 Alabama          0.516         0.514           0.511
2 Alaska           0.627         0.626           0.623
```

Long: multiple rows per observation, a single column contains the values

```
# A tibble: 6 × 3
  State   name            value
  <chr>   <chr>           <dbl>
1 Alabama June_vacc_rate  0.516
2 Alabama May_vacc_rate   0.514
3 Alabama April_vacc_rate 0.511
4 Alaska  June_vacc_rate  0.627
5 Alaska  May_vacc_rate   0.626
6 Alaska  April_vacc_rate 0.623
```

## What is wide/long data?

# Data is wide or long **with respect** to certain variables.



# Why do we need to switch between wide/long data?

Wide: **Easier for humans to read**

```
# A tibble: 2 × 4
  State   June_vacc_rate May_vacc_rate April_vacc_rate
  <chr>            <dbl>         <dbl>           <dbl>
1 Alabama          0.516         0.514           0.511
2 Alaska           0.627         0.626           0.623
```

Long: **Easier for R to make plots & do analysis**

```
# A tibble: 6 × 3
  State   name            value
  <chr>   <chr>           <dbl>
1 Alabama June_vacc_rate  0.516
2 Alabama May_vacc_rate   0.514
3 Alabama April_vacc_rate 0.511
4 Alaska  June_vacc_rate  0.627
5 Alaska  May_vacc_rate   0.626
6 Alaska  April_vacc_rate 0.623
```

# Pivoting using `tidyr` package

`tidyr` allows you to "tidy" your data. We will be talking about:

- `pivot_longer` - make multiple columns into variables, (wide to long)
- `pivot_wider` - make a variable into multiple columns, (long to wide)
- `separate` - string into multiple columns

The `reshape` command exists. Its arguments are considered more confusing, so we don't recommend it.

You might see old functions `gather` and `spread` when googling. These are older iterations of `pivot_longer` and `pivot_wider`, respectively.

pivot_longer...

# Reshaping data from **wide to long**

`pivot_longer()` - puts column data into rows (`tidyr` package)

- First describe which columns we want to “pivot_longer”

```
{long_data} <- {wide_data} %>% pivot_longer(cols = {columns to pivot})
```

# Reshaping data from **wide to long**

```
wide_data
```

```
# A tibble: 1 × 3
  June_vacc_rate May_vacc_rate April_vacc_rate
           <dbl>         <dbl>           <dbl>
1          0.516         0.514           0.511
```

```
long_data <- wide_data %>% pivot_longer(cols = everything())
long_data
```

```
# A tibble: 3 × 2
  name            value
  <chr>           <dbl>
1 June_vacc_rate  0.516
2 May_vacc_rate   0.514
3 April_vacc_rate 0.511
```

# Reshaping data from **wide to long**

`pivot_longer()` - puts column data into rows (`tidyr` package)

- First describe which columns we want to “pivot_longer”
- `names_to` = gives a new name to the pivoted columns
- `values_to` = gives a new name to the values that used to be in those columns

```
{long_data} <- {wide_data} %>% pivot_longer(cols = {columns to pivot},
                                            names_to = {New column name: contains old column names},
                                            values_to = {New column name: contains cell values})
```

# Reshaping data from **wide to long**

```
wide_data
```

```
# A tibble: 1 × 3
  June_vacc_rate May_vacc_rate April_vacc_rate
           <dbl>         <dbl>           <dbl>
1          0.516         0.514           0.511
```

```
long_data <- wide_data %>% pivot_longer(cols = everything(),
                                        names_to = "Month",
                                        values_to = "Rate")
long_data
```

```
# A tibble: 3 × 2
  Month            Rate
  <chr>           <dbl>
1 June_vacc_rate  0.516
2 May_vacc_rate   0.514
3 April_vacc_rate 0.511
```

Newly created column names are enclosed in quotation marks.

# Data used: Charm City Circulator

https://jhudatascience.org/intro_to_r/data/Charm_City_Circulator_Ridership.csv

```
circ <-
  read_csv("https://sisbid.github.io/Data-Wrangling/data/Charm_City_Circulator_Ridership.csv")
head(circ, 5)
```

```
# A tibble: 5 × 15
  day       date      orangeBoardings orangeAlightings orangeAverage purpleBoardings purpleAlightings
  <chr>     <chr>               <dbl>            <dbl>         <dbl>            <dbl>             <dbl>
1 Monday    01/11/20…             877             1027           952               NA                NA
2 Tuesday   01/12/20…             777              815           796               NA                NA
3 Wednesday 01/13/20…            1203             1220          1212.              NA                NA
4 Thursday  01/14/20…            1194             1233          1214.              NA                NA
5 Friday    01/15/20…            1645             1643          1644               NA                NA
# ℹ 8 more variables: purpleAverage <dbl>, greenBoardings <dbl>, greenAlightings <dbl>,
#   greenAverage <dbl>, bannerBoardings <dbl>, bannerAlightings <dbl>, bannerAverage <dbl>,
#   daily <dbl>
```

# Reshaping data from **wide to long**

```
long <- circ %>%
  pivot_longer(starts_with(c("orange","purple","green","banner")))
long
```

```
# A tibble: 13,752 × 5
   day    date       daily name              value
   <chr>  <chr>      <dbl> <chr>             <dbl>
 1 Monday 01/11/2010   952 orangeBoardings     877
 2 Monday 01/11/2010   952 orangeAlightings   1027
 3 Monday 01/11/2010   952 orangeAverage       952
 4 Monday 01/11/2010   952 purpleBoardings      NA
 5 Monday 01/11/2010   952 purpleAlightings     NA
 6 Monday 01/11/2010   952 purpleAverage        NA
 7 Monday 01/11/2010   952 greenBoardings       NA
 8 Monday 01/11/2010   952 greenAlightings      NA
 9 Monday 01/11/2010   952 greenAverage         NA
10 Monday 01/11/2010   952 bannerBoardings      NA
# ℹ 13,742 more rows
```

# Reshaping data from **wide to long**

There are many ways to select the columns we want. Use `?tidyr_tidy_select` to look at more column selection options.

```
long <- circ %>%
  pivot_longer( !c(day, date, daily))
long
```

```
# A tibble: 13,752 × 5
   day    date       daily name              value
   <chr>  <chr>      <dbl> <chr>             <dbl>
 1 Monday 01/11/2010   952 orangeBoardings     877
 2 Monday 01/11/2010   952 orangeAlightings   1027
 3 Monday 01/11/2010   952 orangeAverage       952
 4 Monday 01/11/2010   952 purpleBoardings      NA
 5 Monday 01/11/2010   952 purpleAlightings     NA
 6 Monday 01/11/2010   952 purpleAverage        NA
 7 Monday 01/11/2010   952 greenBoardings       NA
 8 Monday 01/11/2010   952 greenAlightings      NA
 9 Monday 01/11/2010   952 greenAverage         NA
10 Monday 01/11/2010   952 bannerBoardings      NA
# ℹ 13,742 more rows
```

# Cleaning up long data

We will use `str_replace` from the `stringr` package to put _ in the names

```
long <- long %>% mutate(
  name = str_replace(name, "Board", "_Board"),
  name = str_replace(name, "Alight", "_Alight"),
  name = str_replace(name, "Average", "_Average")
)
long
```

```
# A tibble: 13,752 × 5
   day    date       daily name               value
   <chr>  <chr>      <dbl> <chr>              <dbl>
 1 Monday 01/11/2010   952 orange_Boardings     877
 2 Monday 01/11/2010   952 orange_Alightings   1027
 3 Monday 01/11/2010   952 orange_Average       952
 4 Monday 01/11/2010   952 purple_Boardings      NA
 5 Monday 01/11/2010   952 purple_Alightings     NA
 6 Monday 01/11/2010   952 purple_Average       NA
 7 Monday 01/11/2010   952 green_Boardings       NA
 8 Monday 01/11/2010   952 green_Alightings      NA
 9 Monday 01/11/2010   952 green_Average         NA
10 Monday 01/11/2010   952 banner_Boardings      NA
# ℹ 13,742 more rows
```

# Cleaning up long data

Now each `var` is Boardings, Averages, or Alightings. We use "`into =`" to name the new columns and "`sep =`" to show where the separation should happen.

```
long <- long %>%
  separate(name, into = c("line", "type"), sep = "_")
long
```

```
# A tibble: 13,752 × 6
   day    date       daily line   type       value
   <chr>  <chr>      <dbl> <chr>  <chr>      <dbl>
 1 Monday 01/11/2010   952 orange Boardings    877
 2 Monday 01/11/2010   952 orange Alightings  1027
 3 Monday 01/11/2010   952 orange Average      952
 4 Monday 01/11/2010   952 purple Boardings     NA
 5 Monday 01/11/2010   952 purple Alightings    NA
 6 Monday 01/11/2010   952 purple Average      NA
 7 Monday 01/11/2010   952 green  Boardings     NA
 8 Monday 01/11/2010   952 green  Alightings    NA
 9 Monday 01/11/2010   952 green  Average      NA
10 Monday 01/11/2010   952 banner Boardings     NA
# ℹ 13,742 more rows
```

`pivot_wider`...

# Reshaping data from **long to wide**

`pivot_wider()` - spreads row data into columns (`tidyr` package)

- `names_from` = the old column whose contents will be spread into multiple new column names.
- `values_from` = the old column whose contents will fill in the values of those new columns.

```
{wide_data} <- {long_data} %>%
  pivot_wider(names_from = {Old column name: contains new column names},
              values_from = {Old column name: contains new cell values})
```

# Reshaping data from long to wide

```
long_data
```

```
# A tibble: 3 × 2
  Month             Rate
  <chr>            <dbl>
1 June_vacc_rate  0.516
2 May_vacc_rate   0.514
3 April_vacc_rate 0.511
```

```
wide_data <- long_data %>% pivot_wider(names_from = "Month",
                                       values_from = "Rate")
wide_data
```

```
# A tibble: 1 × 3
  June_vacc_rate May_vacc_rate April_vacc_rate
           <dbl>         <dbl>           <dbl>
1          0.516         0.514           0.511
```

# Reshaping Charm City Circulator

```
long
```

```
# A tibble: 13,752 × 6
   day    date       daily line   type        value
   <chr>  <chr>      <dbl> <chr>  <chr>       <dbl>
 1 Monday 01/11/2010   952 orange Boardings     877
 2 Monday 01/11/2010   952 orange Alightings   1027
 3 Monday 01/11/2010   952 orange Average       952
 4 Monday 01/11/2010   952 purple Boardings      NA
 5 Monday 01/11/2010   952 purple Alightings     NA
 6 Monday 01/11/2010   952 purple Average       NA
 7 Monday 01/11/2010   952 green  Boardings      NA
 8 Monday 01/11/2010   952 green  Alightings     NA
 9 Monday 01/11/2010   952 green  Average       NA
10 Monday 01/11/2010   952 banner Boardings      NA
# ℹ 13,742 more rows
```

# Reshaping Charm City Circulator

```
wide <- long %>% pivot_wider(names_from = "type",
                             values_from = "value")
wide
```

```
# A tibble: 4,584 × 7
   day       date       daily line   Boardings Alightings Average
   <chr>     <chr>      <dbl> <chr>      <dbl>      <dbl>   <dbl>
 1 Monday    01/11/2010  952  orange       877       1027    952
 2 Monday    01/11/2010  952  purple        NA         NA     NA
 3 Monday    01/11/2010  952  green         NA         NA     NA
 4 Monday    01/11/2010  952  banner        NA         NA     NA
 5 Tuesday   01/12/2010  796  orange       777        815    796
 6 Tuesday   01/12/2010  796  purple        NA         NA     NA
 7 Tuesday   01/12/2010  796  green         NA         NA     NA
 8 Tuesday   01/12/2010  796  banner        NA         NA     NA
 9 Wednesday 01/13/2010 1212. orange      1203       1220   1212.
10 Wednesday 01/13/2010 1212. purple        NA         NA     NA
# ℹ 4,574 more rows
```

`datasets::airquality` the data shows various air quality metrics measured in New York in 1973.

```
air <- datasets::airquality %>% select(Temp, Month, Day)
air
```

```
    Temp Month Day
1     67     5   1
2     72     5   2
3     74     5   3
4     62     5   4
5     56     5   5
6     66     5   6
7     65     5   7
8     59     5   8
9     61     5   9
10    69     5  10
11    74     5  11
12    69     5  12
13    66     5  13
14    68     5  14
15    58     5  15
16    64     5  16
17    66     5  17
18    57     5  18
19    68     5  19
20    62     5  20
21    59     5  21
22    73     5  22
23    61     5  23
24    61     5  24
25    67     5  25
26    81     5  26
27    79     5  27
28    76     5  28
29    78     5  29
30    74     5  30
31    67     5  31
32    84     6   1
33    85     6   2
34    79     6   3
35    82     6   4
36    87     6   5
37    90     6   6
38    87     6   7
39    93     6   8
40    92     6   9
41    82     6  10
42    80     6  11
43    79     6  12
44    77     6  13
45    72     6  14
46    65     6  15
47    73     6  16
48    76     6  17
49    77     6  18
50    76     6  19
51    76     6  20
52    76     6  21
53    75     6  22
54    78     6  23
55    73     6  24
56    80     6  25
57    77     6  26
58    83     6  27
59    84     6  28
60    85     6  29
61    81     6  30
62    84     7   1
63    83     7   2
64    83     7   3
65    88     7   4
66    92     7   5
67    92     7   6
68    89     7   7
69    82     7   8
70    73     7   9
71    81     7  10
72    91     7  11
73    80     7  12
74    81     7  13
75    82     7  14
76    84     7  15
77    87     7  16
78    85     7  17
79    74     7  18
80    81     7  19
81    82     7  20
82    86     7  21
83    85     7  22
84    82     7  23
85    86     7  24
86    88     7  25
87    86     7  26
88    83     7  27
89    81     7  28
90    81     7  29
91    81     7  30
92    82     7  31
93    86     8   1
94    85     8   2
95    87     8   3
96    89     8   4
97    90     8   5
98    90     8   6
99    92     8   7
100   86     8   8
101   86     8   9
102   82     8  10
103   80     8  11
104   77     8  12
105   79     8  13
106   76     8  14
107   71     8  15
108   75     8  16
109   73     8  17
110   71     8  18
111   72     8  19
112   77     8  20
113   73     8  21
114   80     8  22
115   76     8  23
116   77     8  24
117   77     8  25
118   76     8  26
119   76     8  27
120   79     8  28
121   77     8  29
122   72     8  30
123   77     8  31
124   73     9   1
125   67     9   2
126   62     9   3
127   64     9   4
128   67     9   5
129   68     9   6
130   78     9   7
131   79     9   8
132   77     9   9
133   72     9  10
134   75     9  11
135   79     9  12
136   81     9  13
137   86     9  14
138   97     9  15
139   94     9  16
140   96     9  17
141   94     9  18
142   91     9  19
143   92     9  20
144   85     9  21
145   89     9  22
146   90     9  23
147   78     9  24
148   74     9  25
149   71     9  26
150   76     9  27
151   77     9  28
152   71     9  29
153   71     9  30
```

# Adding prefixes

Month

Let's pivot wider, but it might be helpful to add "Month" to the new column name so it isn't just numbers.

```
air %>% pivot_wider(names_from = "Month",
                    values_from = "Temp",
                    names_prefix = "Month_")
```

# Adding prefixes

# Summary

- `tidyr` package helps us convert between wide and long data
- `pivot_longer()` goes from wide -> long
  - Specify columns you want to pivot
  - Specify `names_to =` and `values_to =` for custom naming
- `pivot_wider()` goes from long -> wide
  - Specify `names_from =` and `values_from =`