INFO 601: Wrangling 2
Multiple-Table Data Wrangling

Keith VanderLinden
Calvin University

Multiple-Table Data Wrangling with dplyr

`dplyr`, Tidyverse's data manipulation package, provides a *grammar* of *multiple-table* data manipulation "verbs" implemented as functions:

- `inner_join()`
- `left_join()`
- `right_join()`
- `full_join()`

These functions integrate with the tidyverse as the single-table functions did.

Dataset: Women in Science

We'd like to work with this dataset of female scientists (taken from Meet 10 Women in Science Who Changed the World) with the goal of merging related information from multiple tables.

```
notability <-
  read_csv("data/notability.csv")
glimpse(notability)


professions <-
  read_csv("data/professions.csv")
glimpse(professions)


dates <- read_csv("data/dates.csv",
    col_types = cols(
      birth_year = col_integer(),
      death_year = col_integer())
    )
glimpse(dates)
```

```
Rows: 9
Columns: 2
$ name      <chr> "Ada Lovelace", "Marie Curie", "Jan
$ known_for <chr> "first computer algorithm", "theory

Rows: 10
Columns: 2
$ name       <chr> "Ada Lovelace", "Marie Curie", "Ja
$ profession <chr> "Mathematician", "Physicist and Ch

Rows: 8
Columns: 3
$ name       <chr> "Janaki Ammal", "Chien-Shiung Wu",
$ birth_year <int> 1897, 1912, 1918, 1920, 1928, 1930
$ death_year <int> 1984, 1997, 2020, 1958, 2016, NA,
```

Dataset: Multiple Tables

professions | dates | notability | Desired Output

```
# A tibble: 10 x 2
   name               profession
   <chr>              <chr>
 1 Ada Lovelace       Mathematician
 2 Marie Curie        Physicist and Chemist
 3 Janaki Ammal       Botanist
 4 Chien-Shiung Wu    Physicist
 5 Katherine Johnson  Mathematician
 6 Rosalind Franklin  Chemist
 7 Vera Rubin         Astronomer
 8 Gladys West        Mathematician
 9 Flossie Wong-Staal Virologist and Molecular Biologist
10 Jennifer Doudna    Biochemist
```

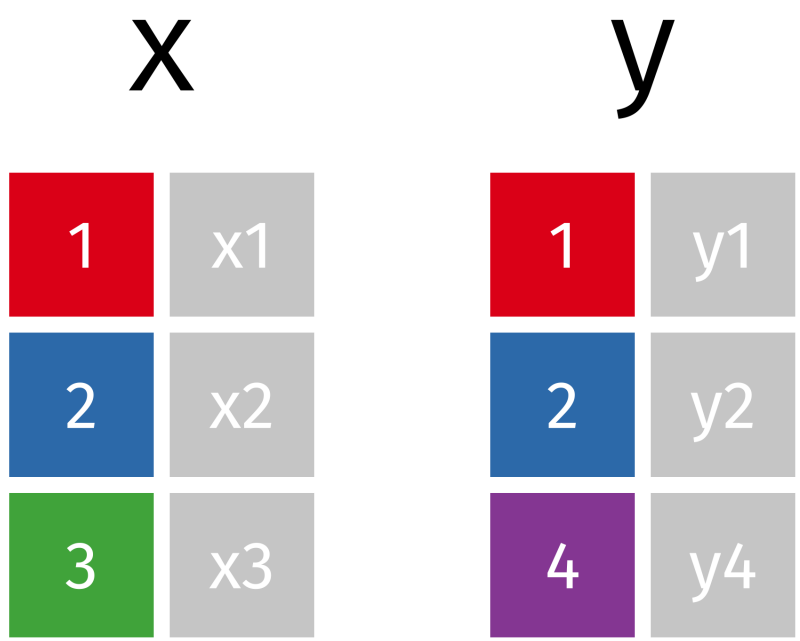
The `name` column is the *primary key* for all of these tables.

The Join Operation

Joining two tables "matches up" records from the two tables (e.g., `X` & `Y`) based on matching *key* values.

```
# A tibble: 3 x 2
    key xdata
  <dbl> <chr>
1     1 x1
2     2 x2
3     3 x3

# A tibble: 3 x 2
    key ydata
  <dbl> <chr>
1     1 y1
2     2 y2
3     4 y4
```

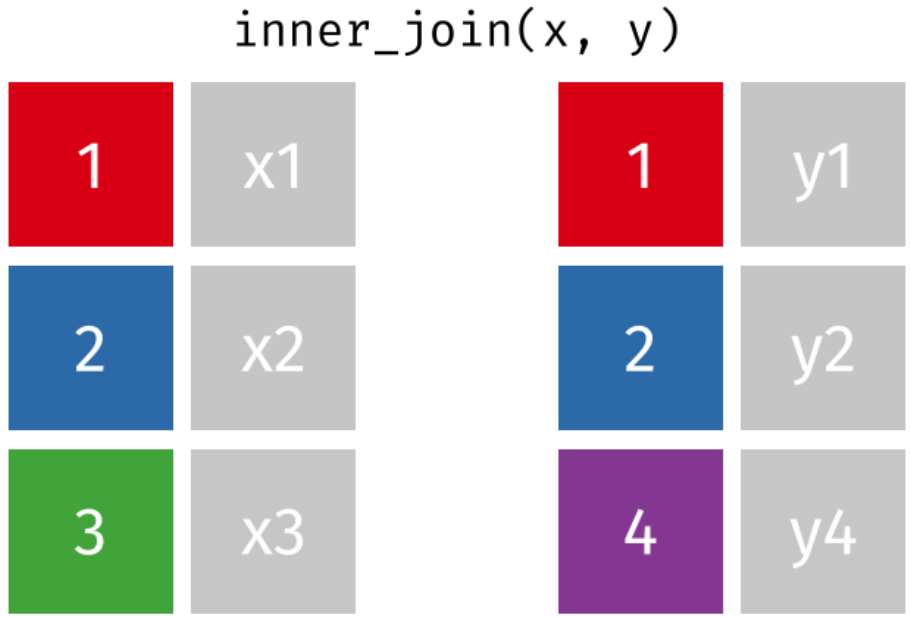


Images from: TidyExplain

Inner Join

```
inner_join(x, y, by = "key")
```

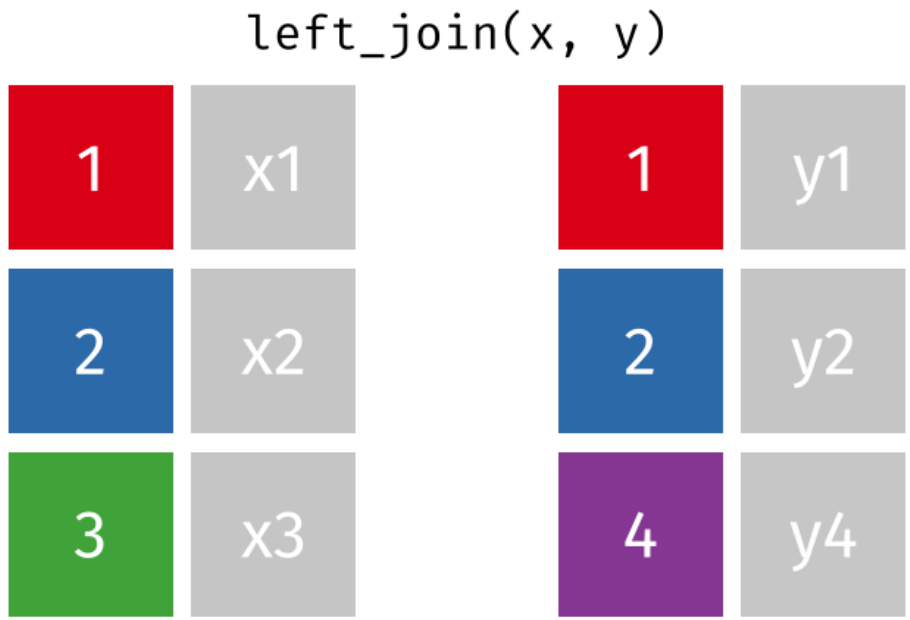
```
# A tibble: 2 x 3
    key xdata ydata
  <dbl> <chr> <chr>
1     1 x1    y1
2     2 x2    y2
```



Left Join

```
left_join(x, y, by = "key")
```

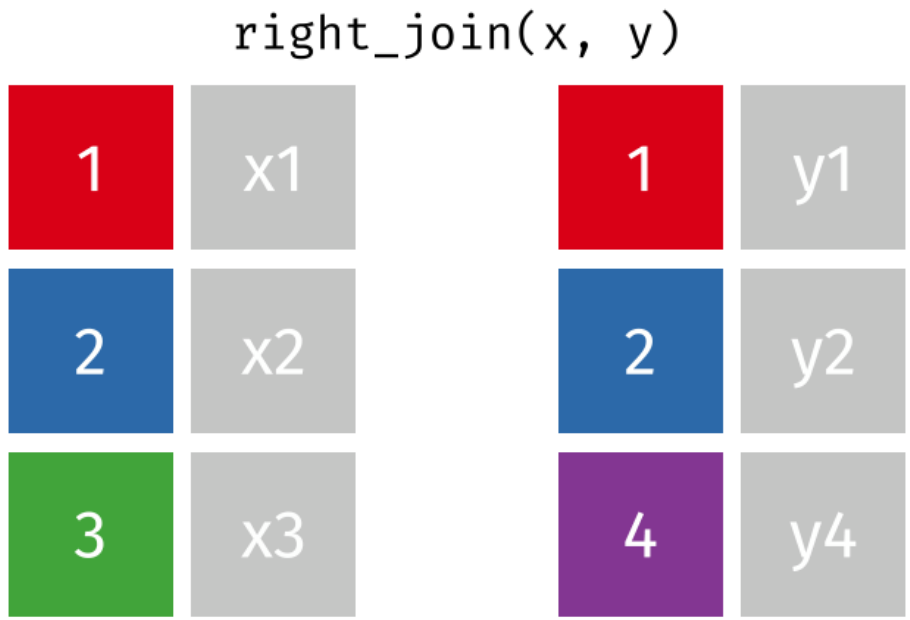
```
# A tibble: 3 x 3
    key xdata ydata
  <dbl> <chr> <chr>
1     1 x1    y1
2     2 x2    y2
3     3 x3    <NA>
```



Right Join

```
right_join(x, y, by = "key")
```

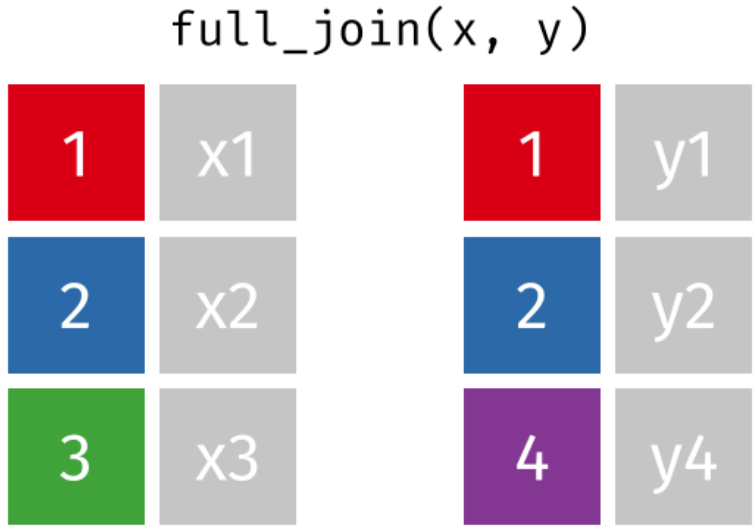
```
# A tibble: 3 x 3
    key xdata ydata
  <dbl> <chr> <chr>
1     1 x1    y1
2     2 x2    y2
3     4 <NA>  y4
```



Full (aka. Outer) Join

```
full_join(x, y, by = "key")
```

```
# A tibble: 4 x 3
    key xdata ydata
  <dbl> <chr> <chr>
1     1 x1    y1
2     2 x2    y2
3     3 x3    <NA>
4     4 <NA>  y4
```



Example: Women in Science

```
professions %>%
  left_join(dates, by = "name") %>%
  left_join(notability, by = "name")
```

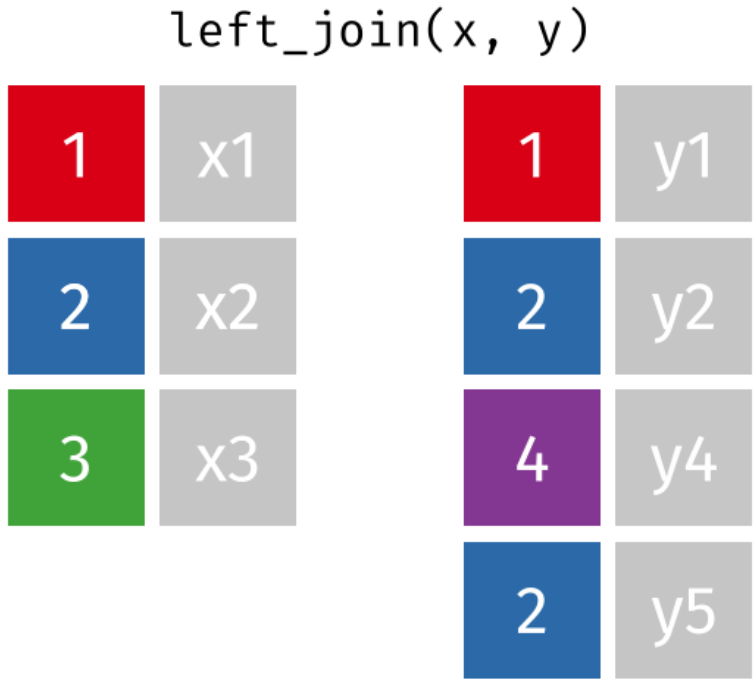
```
# A tibble: 10 x 5
   name               profession  birth_year death_year known_for
   <chr>              <chr>            <int>      <int> <chr>
 1 Ada Lovelace       Mathematic~         NA         NA first co~
 2 Marie Curie        Physicist ~         NA         NA theory o~
 3 Janaki Ammal       Botanist          1897       1984 hybrid s~
 4 Chien-Shiung Wu    Physicist         1912       1997 confim a~
 5 Katherine Johnson  Mathematic~       1918       2020 calculat~
 6 Rosalind Franklin  Chemist           1920       1958 <NA>
 7 Vera Rubin         Astronomer        1928       2016 existenc~
 8 Gladys West        Mathematic~       1930         NA mathemat~
 9 Flossie Wong-Staal Virologist~       1947         NA first sc~
10 Jennifer Doudna    Biochemist        1964         NA one of t~
```

The `name` field is used to join records. In cases where the field names don't match, we use `key = c("key_name_table1", "key_name_table2")` to specify the key names in the first and second tables respectively.

Left Join - Multiple Matches

```
left_join(x, y_extra, by = "key")
```

```
# A tibble: 4 x 3
    key xdata ydata
  <dbl> <chr> <chr>
1     1 x1    y1
2     2 x2    y2
3     2 x2    y5
4     3 x3    <NA>
```



Example: Women in Science with Multiple Matches

```
notability_multi <-
  read_csv("data/notability-multi.csv")
notability_multi
```

```
# A tibble: 13 x 2
  name          known_for
  <chr>         <chr>
1 Ada Lovelace first computer algorithm,
2 Marie Curie  theory of radioactivity,
3 Marie Curie  discovery of elements polonium and rad
4 Marie Curie  first woman to win a Nobel Prize,
5 Janaki Ammal hybrid species,
...
```

```
professions_multi <-
  read_csv("data/professions-multi.csv")
professions_multi
```

```
# A tibble: 12 x 2
   name               profession
   <chr>              <chr>
 1 Ada Lovelace       Mathematician
 2 Marie Curie        Physicist
 3 Marie Curie        Chemist
 4 Janaki Ammal       Botanist
 5 Chien-Shiung Wu    Physicist
...
```

```
# The dates dataset doesn't change.
```

Example: Women in Science with Multiple Matches

```
professions_multi %>%
  left_join(dates, by = "name") %>%
  left_join(notability_multi, by = "name")
```

```
# A tibble: 18 x 5
  name          profession    birth_year death_year known_for
  <chr>         <chr>              <int>      <int> <chr>
1 Ada Lovelace Mathematician         NA         NA first compute~
2 Marie Curie  Physicist             NA         NA theory of rad~
3 Marie Curie  Physicist             NA         NA discovery of ~
4 Marie Curie  Physicist             NA         NA first woman t~
5 Marie Curie  Chemist               NA         NA theory of rad~
6 Marie Curie  Chemist               NA         NA discovery of ~
# ... with 12 more rows
```