Joining Review

PUBLISHED

March 9, 2023

Load data

```
superheroes <- readRDS(here("data","superheroes.rds"))
publishers <- readRDS(here("data","publishers.rds"))
```

```
superheroes
```

```
# A tibble: 7 × 4
  name     alignment gender publisher
  <chr>    <chr>     <chr>  <chr>
1 Magneto  bad       male   Marvel
2 Storm    good      female Marvel
3 Mystique bad       female Marvel
4 Batman   good      male   DC
5 Joker    bad       male   DC
6 Catwoman bad       female DC
7 Hellboy  good      male   Dark Horse Comics
```

```
publishers
```

```
# A tibble: 3 × 2
  publisher yr_founded
  <chr>          <int>
1 DC              1934
2 Marvel          1939
3 Image           1992
```

“Inner” Join

The most common type we’ll use early on. Looks for only when things match up in both tables.

Let’s do it… we’ll use dplyr’s `inner_join()` function.

```
inner_join(superheroes, publishers)
```

```
Joining with `by = join_by(publisher)`

# A tibble: 6 × 5
  name     alignment gender publisher yr_founded
  <chr>    <chr>     <chr>  <chr>          <int>
1 Magneto  bad       male   Marvel          1939
2 Storm    good      female Marvel          1939
3 Mystique bad       female Marvel          1939
4 Batman   good      male   DC              1934
5 Joker    bad       male   DC              1934
6 Catwoman bad       female DC              1934
```

Wait, how did it even know what to join?

By default it looks for column names in columns. This can be good, but also can give you problems if you’re not careful.

How do we tell R to specifically match on a column?

```
inner_join(superheroes, publishers, by = "publisher")
```

```
# A tibble: 6 × 5
  name     alignment gender publisher yr_founded
  <chr>    <chr>     <chr>  <chr>          <int>
1 Magneto  bad       male   Marvel          1939
2 Storm    good      female Marvel          1939
3 Mystique bad       female Marvel          1939
4 Batman   good      male   DC              1934
5 Joker    bad       male   DC              1934
6 Catwoman bad       female DC              1934
```

Left Join

This is for when you want *everything* from the first table no matter what, but joining up those that match from the second.

Let’s give it a try…

```
left_join(superheroes, publishers, by = "publisher")
```

```
# A tibble: 7 × 5
  name     alignment gender publisher         yr_founded
  <chr>    <chr>     <chr>  <chr>                  <int>
1 Magneto  bad       male   Marvel                  1939
2 Storm    good      female Marvel                  1939
3 Mystique bad       female Marvel                  1939
4 Batman   good      male   DC                      1934
5 Joker    bad       male   DC                      1934
6 Catwoman bad       female DC                      1934
7 Hellboy  good      male   Dark Horse Comics         NA
```

Full Join

This is for when you want *everything* from *both* tables no matter what….matching where they match, and leaving blank where they don’t.

Let’s give it a go…

```
full_join(superheroes, publishers, by = "publisher")
```

```
# A tibble: 8 × 5
  name     alignment gender publisher         yr_founded
  <chr>    <chr>     <chr>  <chr>                  <int>
1 Magneto  bad       male   Marvel                  1939
2 Storm    good      female Marvel                  1939
3 Mystique bad       female Marvel                  1939
4 Batman   good      male   DC                      1934
5 Joker    bad       male   DC                      1934
6 Catwoman bad       female DC                      1934
7 Hellboy  good      male   Dark Horse Comics         NA
8 <NA>     <NA>      <NA>   Image                   1992
```

Anti-Join

Finally, there’s the “anti” join, which sounds like what it is – looking for records in one table that are *not* in the other based on the matching variable.

Let’s try it…

```
anti_join(superheroes, publishers, by = "publisher")
```

```
# A tibble: 1 × 4
  name    alignment gender publisher
  <chr>   <chr>     <chr>  <chr>
1 Hellboy good      male   Dark Horse Comics
```