JOINING DATA IN R WITH DPLYR

Advanced joining

What can go wrong?

	Missing	Duplicate
key values	**Missing key values**	**Duplicate key columns**
key columns	**Missing key columns**	Duplicate key columns

Missing key values

```
namesNA
  name    band
1 Mick  Stones
2   NA Beatles
3 Paul Beatles
```

```
> plays
   name  plays
1  John Guitar
2  Paul   Bass
3 Keith Guitar
```

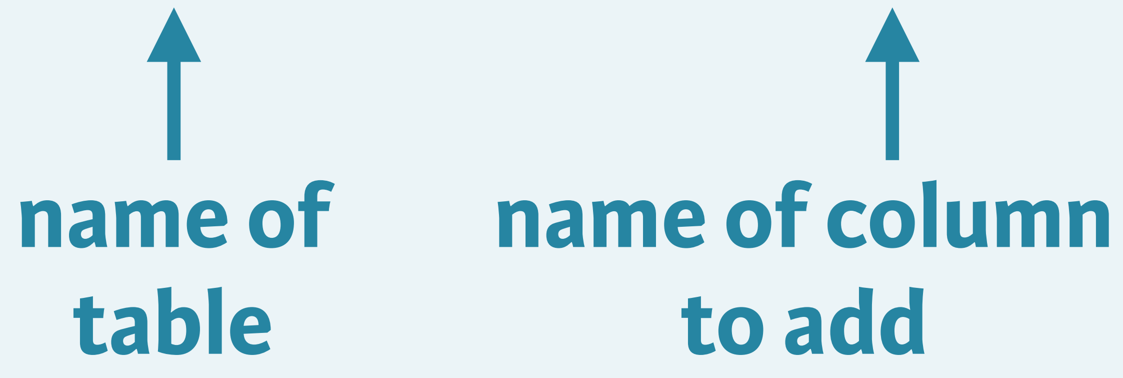
```
> namesNA %>%
+   filter(!is.na(name)) %>%
+   left_join(plays, by = "name")
```

Missing key columns

```
> noNames
        surname    band
Mick     Jagger    <NA>
John     Lennon Beatles
Paul McCartney Beatles
```

```
> plays
   name  plays
1  John Guitar
2  Paul   Bass
3 Keith Guitar
```

```
> library(tibble)
> rownames_to_column(noNames, var = "name")
```



name of table

name of column to add

Duplicate key values

```
> showNames
                name             composer
1 The Sound of Music       Richard Rogers
2 The Sound of Music Oscar Hammerstein II
3     The King and I       Richard Rogers
4     The King and I Oscar Hammerstein II
```

Let’s practice!

Defining the keys

```
> names
  name    band
1 Mick  Stones
2 John Beatles
3 Paul Beatles
```

```
> plays
   name  plays
1  John Guitar
2  Paul   Bass
3 Keith Guitar
```

```
> left_join(names, plays, by = "name")
  name    band  plays
1 Mick  Stones   <NA>
2 John Beatles Guitar
3 Paul Beatles   Bass
```

```
> left_join(names, plays)
Joining, by = "name"
  name    band  plays
1 Mick  Stones   <NA>
2 John Beatles Guitar
3 Paul Beatles   Bass
```

Mismatched key names

```
> members
  member    band
1   Mick  Stones
2   John Beatles
3   Paul Beatles
```

```
> plays
   name  plays
1  John Guitar
2  Paul   Bass
3 Keith Guitar
```

```
> left_join(members, plays, by = c("member" = "name"))
```

column in 1st table

column in 2nd table

Conflicting names

```
> playsWith
   name   plays
1  Mick  Stones
2  John Beatles
3  Paul Beatles
```

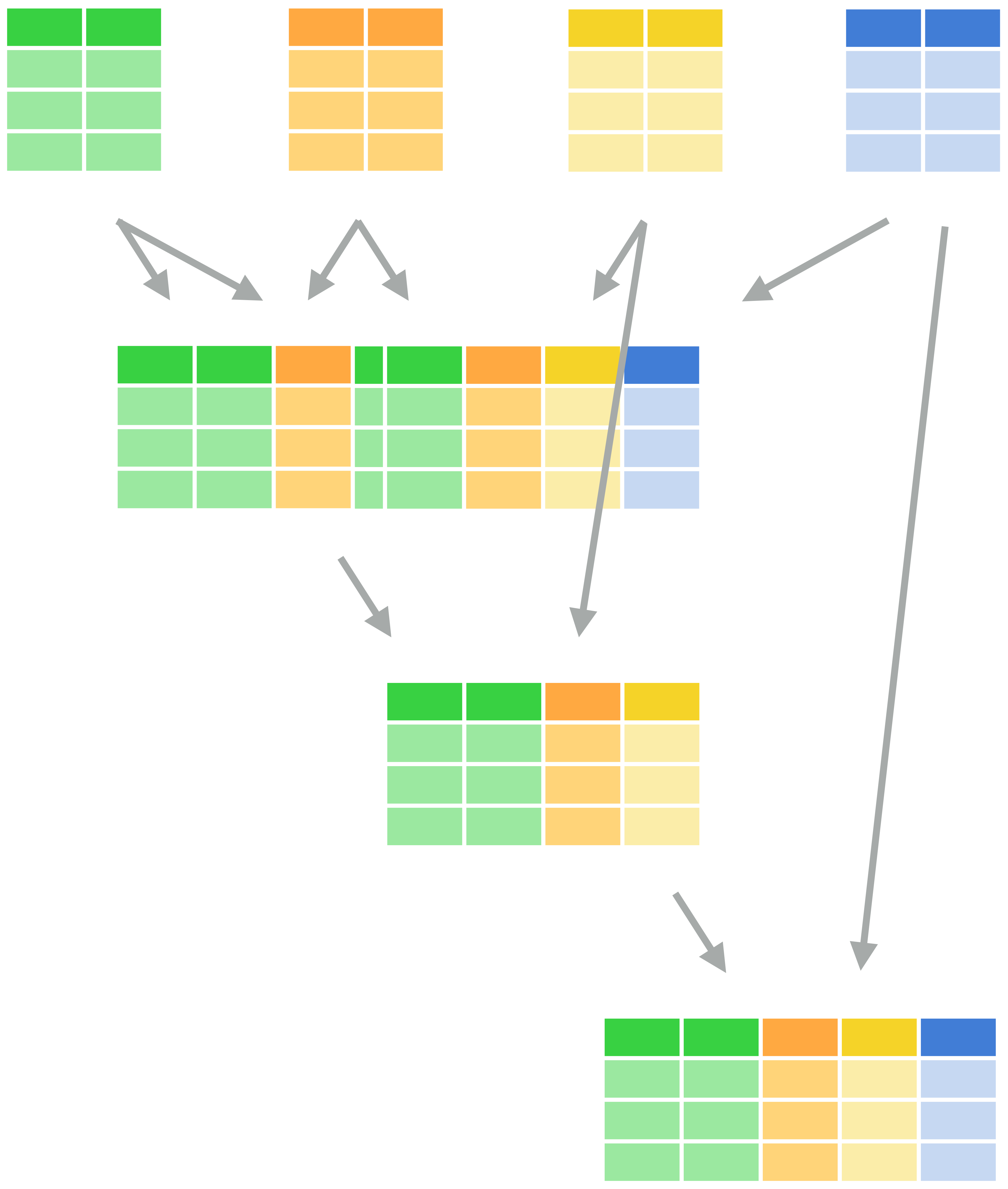
```
> plays
   name  plays
1  John Guitar
2  Paul   Bass
3 Keith Guitar
```

```
> left_join(playsWith, plays, by = "name", suffix = c("1", "2"))
  name  plays1 plays2
1 Mick  Stones   <NA>
2 John Beatles Guitar
3 Paul Beatles   Bass
```

JOINING DATA IN R WITH DPLYR

Let’s practice!

Joining multiple tables



```
> df1 %>%
>   left_join(df2) %>%
>   left_join(df3) %>%
>   left_join(df4)
```

purrr R package

- Applies functions in efficient ways
- `reduce()`
- Works well with `dplyr`

Installing purrr

```
> install.packages("purrr")
> library(purrr)
```

three r's

reduce()

```
> surnames
   name surname
1  Mick  Jagger
2  John  Lennon
3 Ringo   Starr
```

```
> names
  name    band
1 Mick  Stones
2 John Beatles
3 Paul Beatles
```

```
> plays
   name  plays
1  John Guitar
2  Paul   Bass
3 Keith Guitar
```

```
> tables <- list(surnames, names, plays)
> reduce(tables, left_join, by = "name")
```

```
    surnames %>%
      left_join(names, by = "name") %>%
      left_join(plays, by = "name")
```

JOINING DATA IN R WITH DPLYR

Let’s practice!

Other implementations

merge()

```
> merge(names, plays, by = "name", ...)
```

- `left_join(names, plays, by = "name")`

```
> merge(names, plays, by = "name", all.x = TRUE, all.y = FALSE)
```

- `right_join(names, plays, by = "name")`

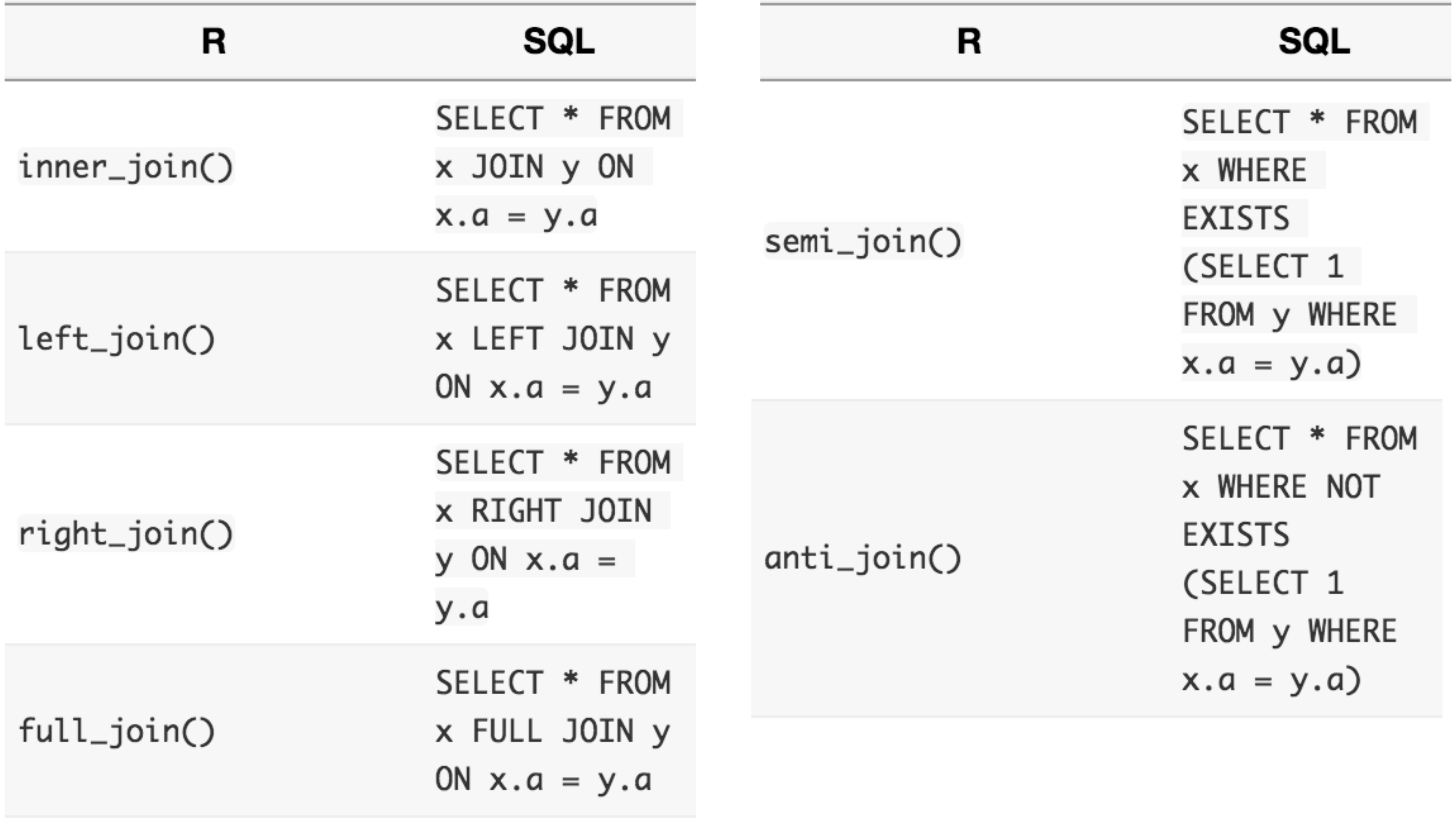
```
> merge(names, plays, by = "name", all.x = FALSE, all.y = TRUE)
```

- `inner_join(names, plays, by = "name")`

```
> merge(names, plays, by = "name", all = FALSE)
```

- `full_join(names, plays, by = "name")`

```
> merge(names, plays, by = "name", all = TRUE)
```



R	SQL
inner_join()	SELECT * FROM x JOIN y ON x.a = y.a
left_join()	SELECT * FROM x LEFT JOIN y ON x.a = y.a
right_join()	SELECT * FROM x RIGHT JOIN y ON x.a = y.a
full_join()	SELECT * FROM x FULL JOIN y ON x.a = y.a

R	SQL
semi_join()	SELECT * FROM x WHERE EXISTS (SELECT 1 FROM y WHERE x.a = y.a)
anti_join()	SELECT * FROM x WHERE NOT EXISTS (SELECT 1 FROM y WHERE x.a = y.a)

dplyr SQL connections

Function	DBMS
`src_sqlite()`	SQLite
`src_mysql()`	MySQL, MariaDB
`src_postgres()`	PostgreSQL

```
install.packages("DBI")
```

```
# Connect to a database
> air <- src_postgres(dbname = "airontime", host =
   "sol-eng-sparklyr.cyii7eabibhu.us-east-1.redshift.amazonaws.com",
   port = "5439", user = "redshift_user", password = "ABCd4321")

# View tables in database
> src_tbls(air)
"flights"  "planes"

# Create table references
> flights <- tbl(air, "flights")
> planes <- tbl(air, "planes")

# Manipulate tables
> flights <- left_join(flights, planes, by = "tailnum")

# Collect results
> flights <- collect(flights)
```

```r
vignette("databases", package = "dplyr")
```

Let’s practice!