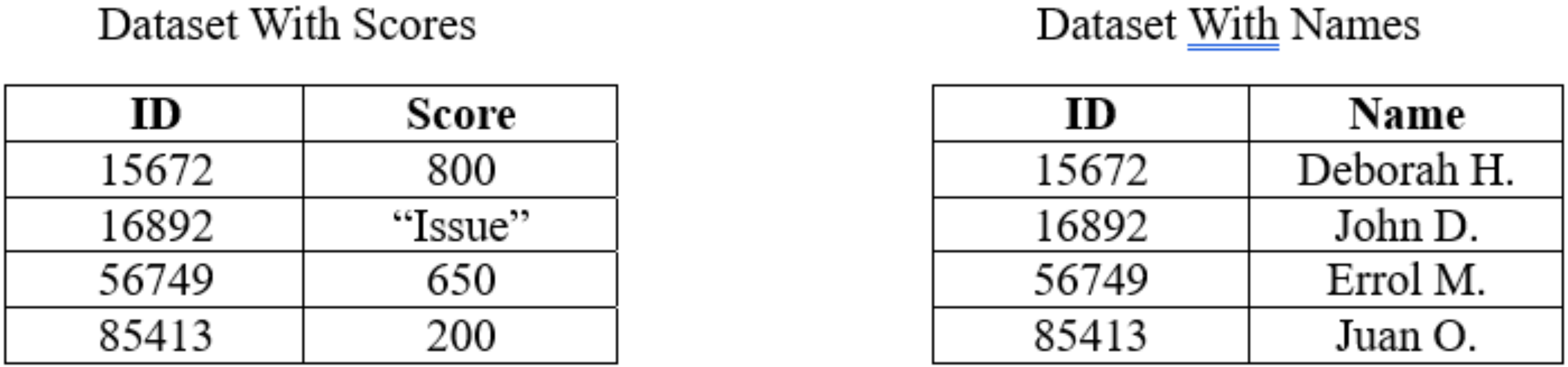
Joining Datasets
Part 1

Example:

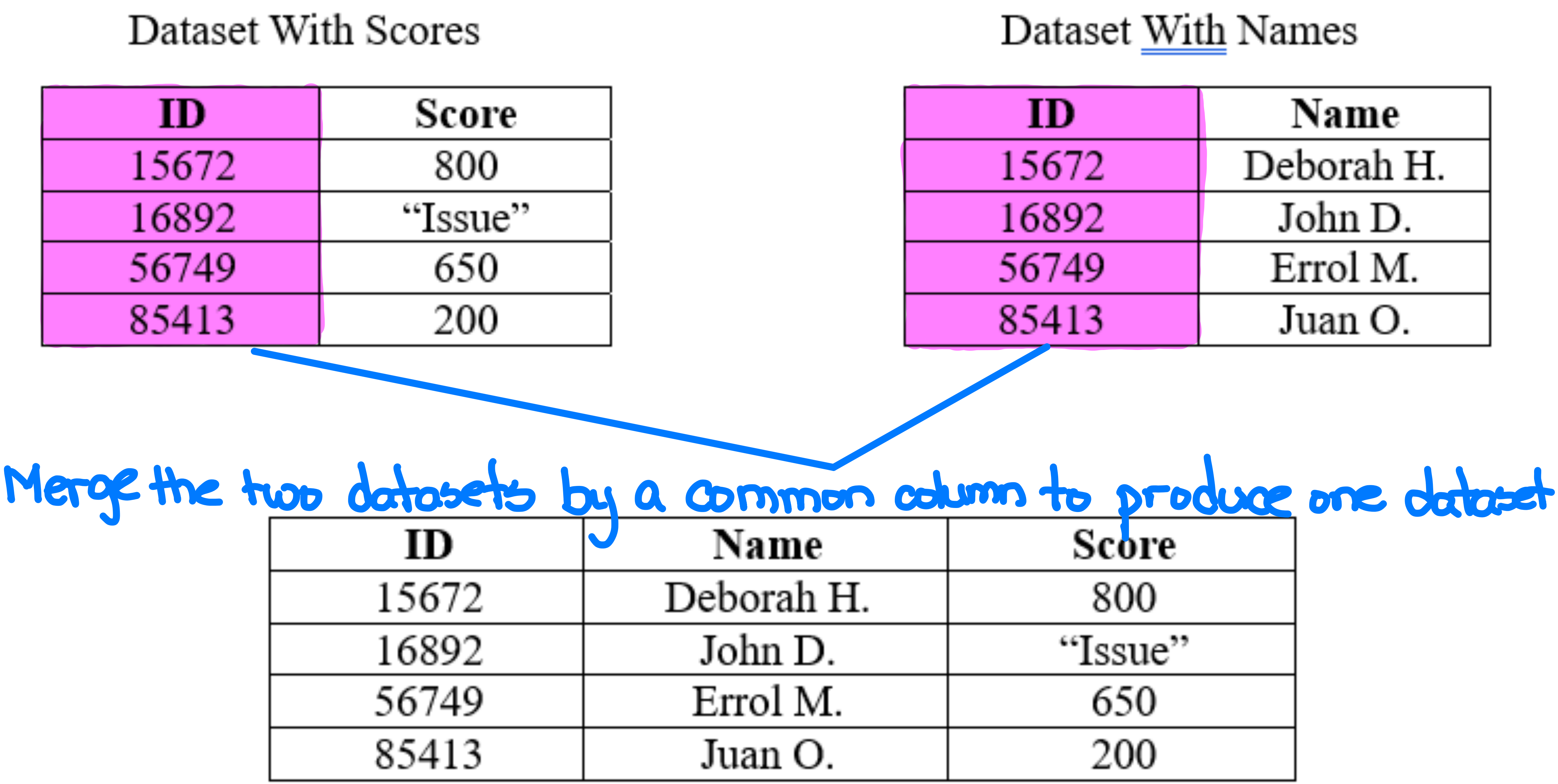


Dataset With Scores

ID	Score
15672	800
16892	“Issue”
56749	650
85413	200

Dataset With Names

ID	Name
15672	Deborah H.
16892	John D.
56749	Errol M.
85413	Juan O.



Dataset With Scores

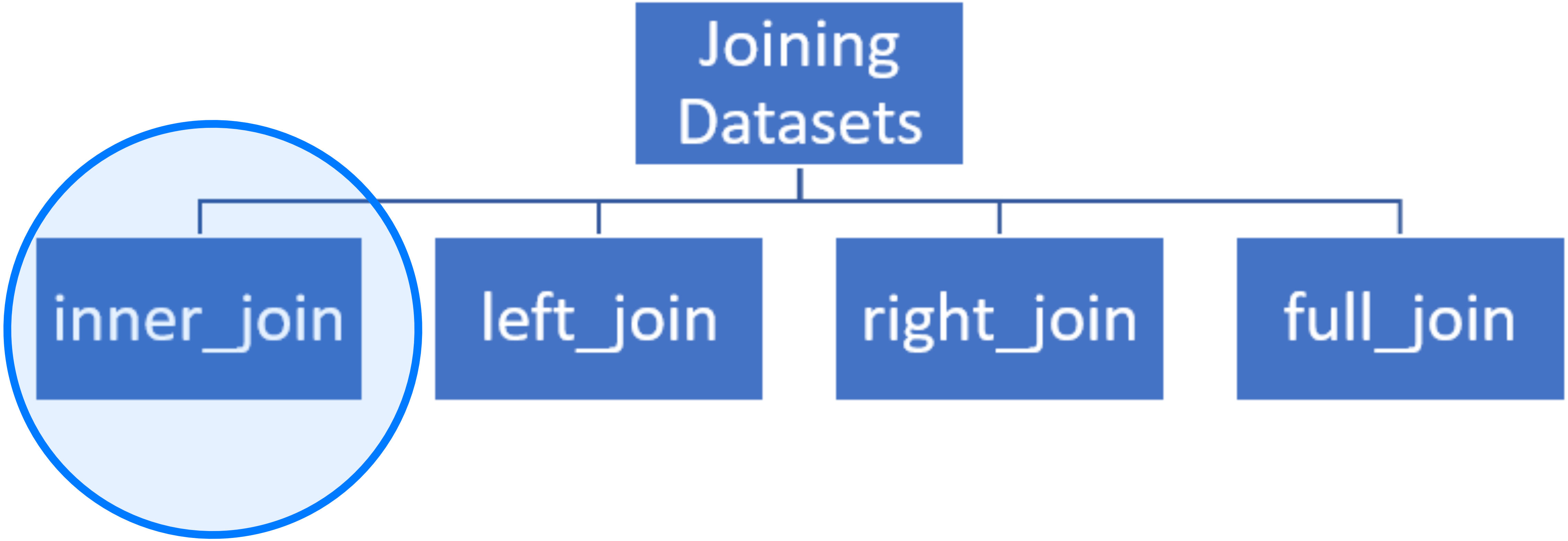
ID	Score
15672	800
16892	“Issue”
56749	650
85413	200

Dataset With Names

ID	Name
15672	Deborah H.
16892	John D.
56749	Errol M.
85413	Juan O.

Merge the two datasets by a common column to produce one dataset

ID	Name	Score
15672	Deborah H.	800
16892	John D.	“Issue”
56749	Errol M.	650
85413	Juan O.	200



Joining Datasets

inner_join

left_join

right_join

full_join

today's class

inner_join() Function

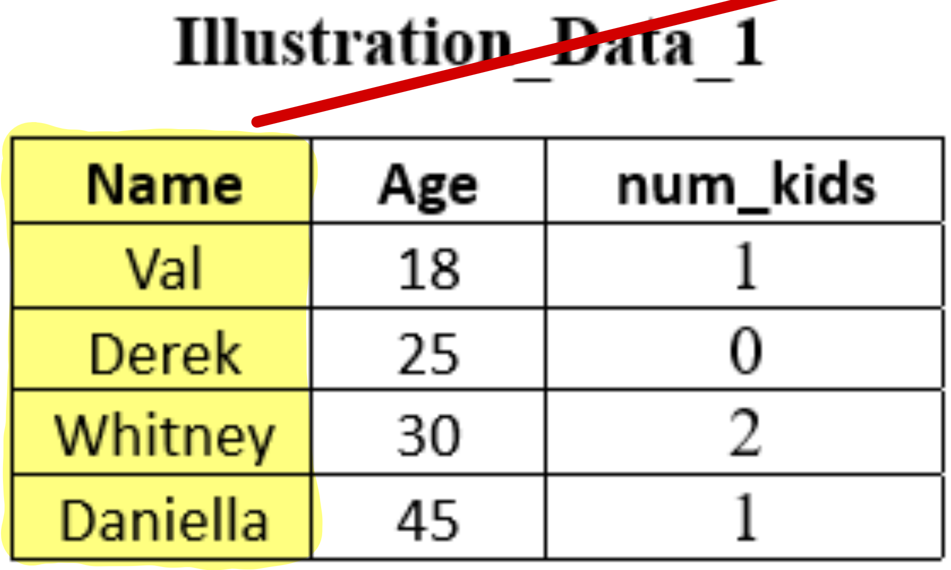
Command Illustration

column name
from this
dataframe

```
new_dataframe_name <- dataframe_name_1 %>%
     inner_join(dataframe_name_2, c("colname_1" = "colname_2"))
```

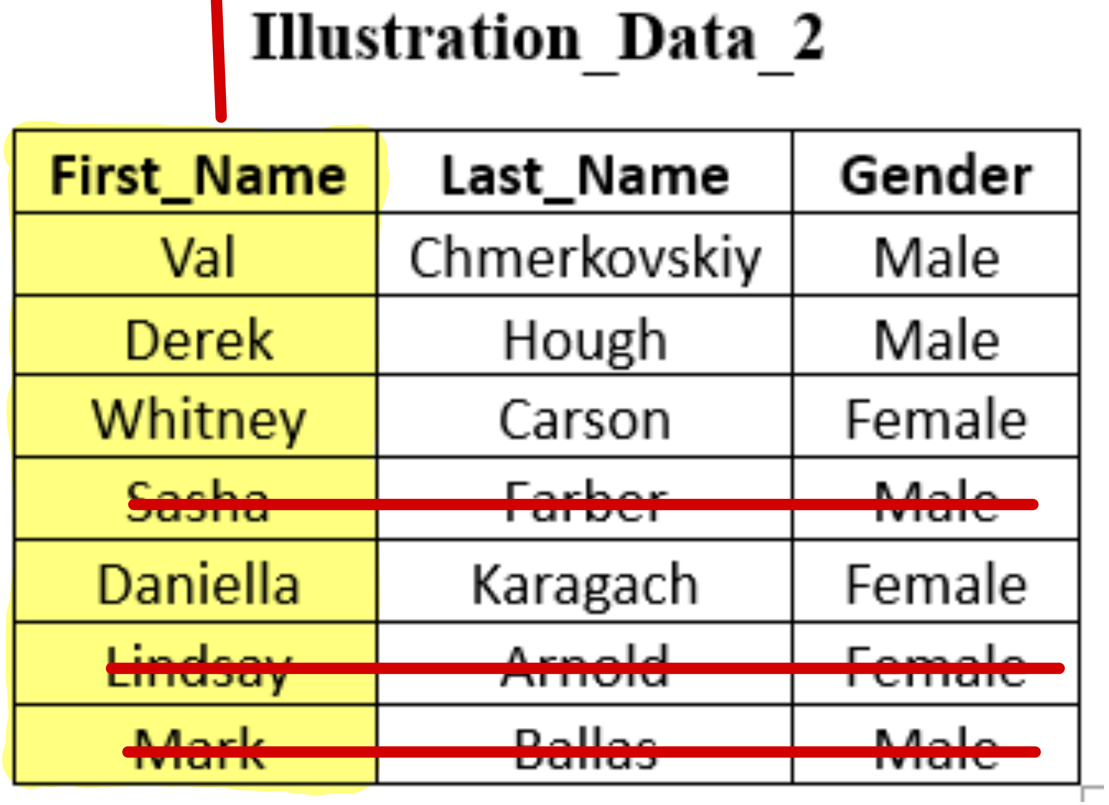
column name
from this
dataframe

Common column



Illustration_Data_1

Name	Age	num_kids
Val	18	1
Derek	25	0
Whitney	30	2
Daniella	45	1



Illustration_Data_2

First_Name	Last_Name	Gender
Val	Chmerkovskiy	Male
Derek	Hough	Male
Whitney	Carson	Female
~~Sasha~~	~~Farber~~	~~Male~~
Daniella	Karagach	Female
~~Lindsay~~	~~Arnold~~	~~Female~~
~~Mark~~	~~Ballas~~	~~Male~~

Inner join, drops rows that are not in both datasets.

Example 1: Perform an inner join between Illustration_Data_1 and Illustration_Data_2

```
example_1 <- Illustration_Data_1 %>%
  inner_join(Illustration_Data_2, c("Name" = "First_Name"))
```

Illustration_Data_1

Name	Age	num_kids
Val	18	1
Derek	25	0
Whitney	30	2
Daniella	45	1

column in common

Illustration_Data_2

First_Name	Last_Name	Gender
Val	Chmerkovskiy	Male
Derek	Hough	Male
Whitney	Carson	Female
~~Sasha~~	~~Farber~~	~~Male~~
Daniella	Karagach	Female
~~Lindsay~~	~~Arnold~~	~~Female~~
~~Mark~~	~~Ballas~~	~~Male~~

will take on the name of the column from the dataframe that came first.

Output will be a dataframe that looks like:

Name	Age	num_Kids	Last_Name	Gender
Val	18	1	Chmerkovsky	Male
Derek	25	0	Hough	Male
Whitney	30	2	Carson	Female
Daniella	45	1	Karagach	Female

Illustration_Data_1

Name	Age	num_kids
Val	18	1
Derek	25	0
Whitney	30	2
Daniella	45	1

Illustration_Data_3

Name	Last_Name	Car
Val	Chmerkovskiy	Mercedes
Val	Chmerkovskiy	Tesla
Val	Chmerkovskiy	Audi
Derek	Hough	Ferrari
Lindsay	Arnold	Tesla
Mark	Ballas	BMW

Example 2: Perform an inner join between Illustration_Data_1 and Illustration_Data_3

```r
example_2 <- Illustration_Data_1 %>%
  inner_join(Illustration_Data_3, c("Name" = "Name"))
```

Illustration_Data_1

Name	Age	num_kids
Val	18	1
Derek	25	0
~~Whitney~~	~~30~~	~~2~~
~~Daniella~~	~~45~~	~~1~~

Illustration_Data_3

Name	Last_Name	Car
Val	Chmerkovskiy	Mercedes
Val	Chmerkovskiy	Tesla
Val	Chmerkovskiy	Audi
Derek	Hough	Ferrari
~~Lindsay~~	~~Arnold~~	~~Tesla~~
~~Mark~~	~~Ballas~~	~~BMW~~

Output will be a dataframe that looks like:

Name	Age	num_kids	Last_Name	Car
Val	18	1	Chmerkovsky	Mercedes
Val	18	1	Chmerkovsky	Tesla
Val	18	1	Chmerkosky	Audi
Derek	25	0	Hough	Ferrari

It is always a good idea to carefully check that the number of rows returned by a join operation is what you expected. In particular, you should carefully check for rows in one table that matched to more than one row in the other table.

- Inspect the column by which you are joining.

```
nrow(Illustration_Data_1)
```

Output: 4

```
n_distinct(Illustration_Data_1$Name)
```

Output: 4

```
nrow(Illustration_Data_2)
```

Output: 7

```
n_distinct(Illustration_Data_2$First_Name)
```

Output: 7

- Check how many data values from one dataset are in the other dataset.

```
table(Illustration_Data_1$Name %in% Illustration_Data_2$First_Name)
```

How many from Data 1 are in Data 2?

Output:

TRUE
4

```
table(Illustration_Data_2$First_Name %in% Illustration_Data_1$Name)
```

How many from Data 2 are in Data 1?

Output:

TRUE
4

FALSE
3

Because every element in each dataframe is different, The merged dataframe will have 4 rows.