

Manually, in Excel we could open the two data frames as spreadsheets and cut and paste cases where the id number matches...

Probably ok for 48 participants, but what if you had 200 or 2,000?

In R, we can use the `inner_join` function from the `dplyr` package where we join the two data frames matched by ID.

```
> dataRT_all
  ID WM  IQ Comp Simple_Sentence Complex_Sentence
1  95 47  94   19           2154           2441
2 400 45 118   18           1824           2456
3 457 42 100   22           1857           2324
4 1138 41  77   18           1902           2341
5 1587 54  67   21           1844           2320
6 1805 52 109   19           2224           2256
7 1864 57 111   19           1880           2391
8 2006 44 110   19           2091           2456
9 2183 55 125   23           1926           2218
10 2318 51  91   21           1960           2440
```

We can use the assignment symbol `<-` to assign the output of this `inner_join` function to a new variable I'm calling `dataRT_all`. We can ask for the structure of this new data frame using the `str()` function:

```
> dataRT_all <- inner_join(data, dataRT, by = (c("ID")))
> str(dataRT_all)
'data.frame':   48 obs. of  6 variables:
 $ ID          : int  95 400 457 1138 1587 1805 1864 2006 2183 2318 ...
 $ WM          : int  47 45 42 41 54 52 57 44 55 51 ...
 $ IQ          : int  94 118 100 77 67 109 111 110 125 91 ...
 $ Comp        : int  19 18 22 18 21 19 19 19 23 21 ...
 $ Simple_Sentence : int  2154 1824 1857 1902 1844 2224 1880 2091 1926 1960 ...
 $ Complex_Sentence: int  2441 2456 2324 2341 2320 2256 2391 2456 2218 2440 ...
```

So we have created a new data frame of 48 participants consisting of their reading times and their individual difference measures from two separate (and different sized) data frames...with one line of code...

```
> head(dataRT_all)
   ID WM  IQ Comp Simple_Sentence Complex_Sentence
1  95 47  94  19         2154           2441
2 400 45 118  18         1824           2456
3 457 42 100  22         1857           2324
4 1138 41  77  18         1902           2341
5 1587 54  67  21         1844           2320
6 1805 52 109  19         2224           2256
```