STOPPD participant count by group

This script identifies the number of participants in olanzapine vs. placebo by scan timepoint, using the logic of group inclusion that Judy and Dielle provided, and Nick and Aristotle have agreed to.

Note: this script includes data from all participants with data in Judy's master log and our file system. It has not excluded participants on any other basis (e.g., QC fail, processing fail, post-hoc clinical trial ineligibility, etc.)

First - identify the number of baseline scans (i.e., scans completed at week 20).

```
#count the number of participants that have a 'yes' for 'completed' in "Scan.completed.1"
n_first_complete = sum(na.omit(df$first_complete == "Yes")) #88 participants completed week 20 scan

#for clarity, print the IDs of the N=88 participants that completed week 20 scans
df %>% filter(first_complete == "Yes") %>% select(STUDYID) %>%
  kable(caption = str_c("Subject ID's who completed first scan n = ", n_first_complete))
```

Table 1: Subject ID's who completed first scan n = 88

STUDYID
110008
110009
110013
110016
110022
110025
110028
110030
110031
110034
120011
120012
120015
120016
120017
120021
120026
210012
210013
210014
210017
210020
210022
210024
210026
210030
210033
210036

STUDYID
210038
210042
210048
210049
210051
220002
220003
220004
220006
220008
220009
310010
310015
310025
310037
310051
310070
320006
320013
320021
320022
320032
320041
320042
320043
320045
410004
410008
410009
410010
410011
410012
410013
410015
410019
410022
410023
410029
410030
410031
410037
410039
410040
410043
410045
410047
420005
420007
420013
420016
420018
420019

STUDYID
420020
420023
420029
420032
420039
420042
420043
420044

RANDOMIZATION - as expected, there's no difference in first scan completion between those randomized to O vs. P group

```
#RANDOMIZATION - as expected, there's no difference in first scan completion between those randomized t
(R <- addmargins(table(df$first_complete == 'Yes', df$randomization))) #O = 45; P = 43 (total = 88)
```

```
##
##         O  P Sum
##   TRUE 45 43  88
##   Sum  45 43  88
```

Second - identify the number of week 56 scans (i.e., 36 weeks after week 20).

```
#make sure that all the participants that completed week 56 scan also completed week 20
all_second_complete <- all((df$second_complete == "Yes") %in% (df$first_complete== "Yes")) #all TRUE

#count the number of participants that have a 'yes' for 'completed' in "Scan.completed" - but this incl
(n_second_complete <- sum(na.omit(df$second_complete == "Yes"))) #74 completed week 56 scan
```

```
## [1] 74
```

Subject ids of the n = 74 who completed their second scan. Note: it is TRUE that all participants who completed their second scan have baseline data.

```
#for clarity, print the IDs of the N=74 participants that completed week 56 scans
df %>% filter(second_complete == "Yes") %>% select(STUDYID)
```

```
## # A tibble: 74 x 1
##    STUDYID
##      <int>
##  1  110008
##  2  110009
##  3  110013
##  4  110022
##  5  110031
##  6  110034
##  7  120011
##  8  120012
##  9  120015
## 10  120016
```

```
## # ... with 64 more rows
```

```
#count how many participants that completed week 56 scan are classified as RCT
sum(na.omit(df$second_complete == 'Yes' & df$second_timepoint == 'RCT')) #RCT = 41
```

```
## [1] 41
```

```
  (as.vector(na.omit(df$STUDYID[df$second_complete == "Yes" & df$second_timepoint == 'RCT']))) #for cla
```

```
##  [1] 110008 110009 110013 110022 110031 110034 120011 120012 120015 210012
## [11] 210013 210014 210017 210020 210030 210051 220004 310051 320006 320021
## [21] 320032 320042 320043 320045 410004 410008 410010 410013 410022 410023
## [31] 410029 410030 410037 410039 410043 420013 420020 420029 420039 420042
## [41] 420043
```

```
sum(na.omit(df$second_complete == 'Yes' & df$second_timepoint == 'Relapse')) #Relapse = 28
```

```
## [1] 28
```

```
  (as.vector(na.omit(df$STUDYID[df$second_complete == "Yes" & df$second_timepoint == 'Relapse']))) #for
```

```
##  [1] 120016 120017 120021 120026 210022 210026 210033 210038 210042 210049
## [11] 220002 220003 220006 220009 310010 310015 310025 310037 320013 410009
## [21] 410011 410012 410031 410040 420007 420016 420023 420032
```

```
sum(na.omit(df$second_complete == 'Yes' & df$second_timepoint == 'Off protocol')) #Off protocol = 5
```

```
## [1] 5
```

```
  (as.vector(na.omit(df$STUDYID[df$second_complete == "Yes" & df$second_timepoint == 'Off protocol'])))
```

```
## [1] 320022 410015 410019 420018 420019
```

```
df %>%
  filter(second_complete == "Yes") %>%
  count(second_timepoint) %>%
  kable(caption = "breakdown of those who where scanned at two timepoints")
```

Table 2: breakdown of those who where scanned at two timepoints

second_timepoint	n
Off protocol	5
RCT	41
Relapse	28

```
#RANDOMIZATION- look at randomization info for those who completed a second timepoint RCT scan
(R <- addmargins(table(df$second_complete == 'Yes' & df$second_timepoint == 'RCT', df$randomization))) 
```

```
##
##          0  P Sum
##   FALSE 14 24  38
##   TRUE  27 14  41
##   Sum   41 38  79
```

```
df %>%
  filter(second_complete == "Yes") %>%
  count(second_timepoint, randomization) %>%
  kable(caption = "breakdown of those who where scanned at two timepoints, by arm")
```

Table 3: breakdown of those who where scanned at two timepoints, by arm

second_timepoint	randomization	n
Off protocol	O	4
Off protocol	P	1
RCT	O	27
RCT	P	14
Relapse	O	8
Relapse	P	20

Third - identify the number of "off label" scans also at week 56.

```
#make sure timepoint is a character
df$second_timepoint <- as.character(df$second_timepoint)

#count the number of scans completed at *third* timepoint, which are by definition "off label"
n_offlable <- sum(na.omit(df$third_complete == 'Yes')) #8 off-label scans

#for clarity, print the IDs of the N=8 participants that completed off-label scans
(as.vector(na.omit(df$STUDYID[df$third_complete == "Yes"])))
```

```
## [1] 110016 210033 210049 220006 310037 320022 410019 420032
```

```
#of these, determine how many "off protocol" vs. "relapse", based on second timepoint scan
sum(na.omit(df$third_complete == 'Yes' & df$second_timepoint  == 'Off protocol')) #2 "off protocol" sca
```

```
## [1] 2
```

```
  (as.vector(na.omit(df$STUDYID[df$third_complete == "Yes" & df$second_timepoint  == 'Off protocol'])))
```

```
## [1] 320022 410019
```

```
sum(na.omit(df$third_complete == 'Yes' & df$second_timepoint == 'Relapse')) #6 relapse scans
```

```
## [1] 6
```

```
  (as.vector(na.omit(df$STUDYID[df$third_complete == "Yes" & df$second_timepoint  == 'Relapse'])))
```

```
## [1] 110016 210033 210049 220006 310037 420032
```

```
#RANDOMIZATION
df %>%
  filter(third_complete == "Yes") %>%
  count(randomization) %>%
  kable(caption = str_c("Breakdown of thrid timepoint 'off-label' scans n = ", n_offlable))
```

Table 4: Breakdown of thrid timepoint 'off-label' scans n = 8

randomization	n
O	3
P	5

```
df %>%
  filter(df$second_timepoint == 'Off protocol') %>%
  count(randomization, third_complete) %>%
  kable(caption = str_c("Breakdown of 'off-protocol' scans by presence of third timepoint"))
```

Table 5: Breakdown of 'off-protocol' scans by presence of third timepoint

randomization	third_complete	n
O	Yes	2
O	NA	2
P	NA	1

```
df %>%
  filter(df$second_timepoint == 'Relapse') %>%
  count(randomization, third_complete) %>%
  kable(caption = str_c("Breakdown of thrid timepoint 'Relapse' scans by presence of third timepoint"))
```

Table 6: Breakdown of thrid timepoint 'Relapse' scans by presence of third timepoint

randomization	third_complete	n
O	Yes	1
O	NA	9
P	Yes	5
P	NA	18

Identify the scans completed between week 20 and week 56 which are the relapse scans (and in a small minority of cases may be a scan when somebody is moving or wants out of the study despite being well).

```
#count relapse - note: both 'relapse' and 'off protocol' is included here (everything other than 'RCT')
sum(na.omit((df$second_timepoint == 'Relapse' | df$second_timepoint == 'Off protocol') & df$second_compl
```

```
## [1] 33
```

```
#of these, count how many were "relapse" and how many were "off protocol"
sum(na.omit(df$second_timepoint == 'Relapse' & df$second_complete == 'Yes'))# 28 relapse
```

```
## [1] 28
```

```
sum(na.omit(df$second_timepoint == 'Off protocol' & df$second_complete == 'Yes'))#5 off protocol
```

```
## [1] 5
```

```
#RANDOMIZATION
(R <- addmargins(table((df$second_timepoint == 'Relapse' | df$second_timepoint == 'Off protocol') & df$s
```

```
##
##           O  P Sum
##   FALSE 28 15  43
##   TRUE  12 21  33
```

```
##   Sum   40 36  76
```

```
(R <- addmargins(table(df$second_timepoint == 'Relapse' & df$second_complete == 'Yes', df$randomization
```

```
##
##          O  P Sum
##   FALSE 32 16  48
##   TRUE   8 20  28
##   Sum   40 36  76
```

```
(R <- addmargins(table(df$second_timepoint == 'Off protocol' & df$second_complete == 'Yes', df$randomiza
```

```
##
##          O  P Sum
##   FALSE 38 38  76
##   TRUE   4  1   5
##   Sum   42 39  81
```

```
rm(df, R)
```