assignment2

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyr)
AI2 <-read.table("assignment2", header = TRUE)
AI2 <- as_tibble(AI2)
AI2_ID <- tibble::rowid_to_column(AI2, "ID")
which(AI2_ID$value == 1)
```

```
## [1] 3 8 9
```

```
AI2t <- as_tibble(AI2_ID)
df <- mutate(AI2_ID, diffthree = abs(ID - 3))
df <- mutate(df, diffeight = abs(ID - 8))
df <- mutate(df, diffnine = abs(ID - 9))
dft <- as_tibble(df)
dft
```

```
## # A tibble: 14 x 5
##       ID value diffthree diffeight diffnine
##    <int> <int>     <dbl>     <dbl>    <dbl>
##  1     1     0         2         7        8
##  2     2     0         1         6        7
##  3     3     1         0         5        6
##  4     4     0         1         4        5
##  5     5     0         2         3        4
##  6     6     0         3         2        3
##  7     7     0         4         1        2
##  8     8     1         5         0        1
##  9     9     1         6         1        0
## 10    10     0         7         2        1
## 11    11     0         8         3        2
## 12    12     0         9         4        3
## 13    13     0        10         5        4
## 14    14     0        11         6        5
```

```
dftest <- mutate(df, test = diffthree/14)
```

I was thinking to use subtraction to get the smallest difference and compare across the 3 rows to decide which one to keep. Kind of like a conditional statement, such as if column diffthree has lowest integer return 3 in a new column in that index row. Unfortunately, I can't seem to find the right functions to do so, perhaps I am not googling the correct words in order to find that function or conditonal statement to use. Perhaps I am also going down an incorrect path and there is another set of steps that need to be done to get to the final output. I just can't seem to figure out the first step neccessary if subtraction of the distances is not it.

```r
AI2r <-read.table("assignment2", header = TRUE)
AI2r$ID <- seq.int(nrow(AI2r))
which(AI2r$value == 1)
```

```
## [1] 3 8 9
```

```r
AI2r
```

```
##    value ID
## 1      0  1
## 2      0  2
## 3      1  3
## 4      0  4
## 5      0  5
## 6      0  6
## 7      0  7
## 8      1  8
## 9      1  9
## 10     0 10
## 11     0 11
## 12     0 12
## 13     0 13
## 14     0 14
```