Class 1

Libraries Used

```
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(stringr)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

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

```
library(DataCombine)
```

Upload data into R Studio

```
knitr::opts_chunk$set(echo=T)
# getwd()
# setwd(dir = "E:/MSc DSc/Sem 1/Business Analytics/Ch1_bike_sharing_data.csv")
datch1= read.csv("E:/MSc DSc/Sem 1/Business Analytics/Ch1_bike_sharing_data.csv")
head(datch1,10)
```

```
##         datetime season holiday workingday weather  temp  atemp humidity
## 1  1/1/2011 0:00      1       0          0       1  9.84 14.395       81
## 2  1/1/2011 1:00      1       0          0       1  9.02 13.635       80
## 3  1/1/2011 2:00      1       0          0       1  9.02 13.635       80
```

```
## 4  1/1/2011 3:00      1       0          0       1  9.84 14.395       75
## 5  1/1/2011 4:00      1       0          0       1  9.84 14.395       75
## 6  1/1/2011 5:00      1       0          0       2  9.84 12.880       75
## 7  1/1/2011 6:00      1       0          0       1  9.02 13.635       80
## 8  1/1/2011 7:00      1       0          0       1  8.20 12.880       86
## 9  1/1/2011 8:00      1       0          0       1  9.84 14.395       75
## 10 1/1/2011 9:00      1       0          0       1 13.12 17.425       76
##    windspeed casual registered count
## 1     0.0000      3         13    16
## 2     0.0000      8         32    40
## 3     0.0000      5         27    32
## 4     0.0000      3         10    13
## 5     0.0000      0          1     1
## 6     6.0032      0          1     1
## 7     0.0000      2          0     2
## 8     0.0000      1          2     3
## 9     0.0000      1          7     8
## 10    0.0000      8          6    14
```

```
str(datch1)
```

```
## 'data.frame':    17379 obs. of  12 variables:
##  $ datetime  : chr  "1/1/2011 0:00" "1/1/2011 1:00" "1/1/2011 2:00" "1/1/2011 3:00" ...
##  $ season    : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ holiday   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ workingday: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ weather   : int  1 1 1 1 1 2 1 1 1 1 ...
##  $ temp      : num  9.84 9.02 9.02 9.84 9.84 ...
##  $ atemp     : num  14.4 13.6 13.6 14.4 14.4 ...
##  $ humidity  : int  81 80 80 75 75 75 80 86 75 76 ...
##  $ windspeed : num  0 0 0 0 0 ...
##  $ casual    : int  3 8 5 3 0 0 2 1 1 8 ...
##  $ registered: int  13 32 27 10 1 1 0 2 7 6 ...
##  $ count     : int  16 40 32 13 1 1 2 3 8 14 ...
```

Transforming data

Using logical expression

extracted_rows gives info on no of rows and cols

```
extracted_rows = filter(datch1, registered == 0, season == 1 | season == 2)
dim(extracted_rows)
```

```
## [1] 10 12
```

```
using_membership = filter(datch1, registered == 0, season %in% c(1,2))
identical(extracted_rows, using_membership)
```

```
## [1] TRUE
```

Adding calculated column

```
add_revenue = mutate(extracted_rows, revenue = casual*5)

head(add_revenue,10)
```

```
##          datetime season holiday workingday weather  temp  atemp humidity
## 1   1/1/2011 6:00      1       0          0       1  9.02 13.635       80
## 2  1/10/2011 1:00      1       0          1       1  4.92  6.060       50
## 3   2/2/2011 2:00      1       0          1       3  9.02 11.365       93
## 4   3/2/2011 4:00      1       0          1       1  8.20 10.605       75
## 5   3/4/2011 4:00      1       0          1       2  7.38  9.090       74
## 6   3/7/2011 4:00      1       0          1       1  8.20  7.575       80
## 7   3/8/2011 2:00      1       0          1       1  9.84 12.120       52
## 8  3/10/2011 0:00      1       0          1       3 13.94 15.910        0
## 9   4/3/2011 4:00      2       0          0       1 11.48 15.150       70
## 10  4/4/2011 3:00      2       0          1       1 15.58 19.695       66
##    windspeed casual registered count revenue
## 1     0.0000      2          0     2      10
## 2    19.0012      1          0     1       5
## 3     8.9981      4          0     4      20
## 4     8.9981      1          0     1       5
## 5    12.9980      1          0     1       5
## 6    35.0008      1          0     1       5
## 7     8.9981      1          0     1       5
## 8    16.9979      3          0     3      15
## 9     6.0032      3          0     3      15
## 10   19.0012      1          0     1       5
```

Aggregate Data

```
grouped = group_by(add_revenue,season)
head(grouped,10)
```

```
## # A tibble: 10 x 13
## # Groups:   season [2]
##    datetime     season holiday workingday weather  temp atemp humidity windspeed
##    <chr>         <int>   <int>      <int>   <int> <dbl> <dbl>    <int>     <dbl>
##  1 1/1/2011 6:~      1       0          0       1  9.02 13.6        80      0
##  2 1/10/2011 1~      1       0          1       1  4.92  6.06       50     19.0
##  3 2/2/2011 2:~      1       0          1       3  9.02 11.4        93      9.00
##  4 3/2/2011 4:~      1       0          1       1  8.2  10.6        75      9.00
##  5 3/4/2011 4:~      1       0          1       2  7.38  9.09       74     13.0
##  6 3/7/2011 4:~      1       0          1       1  8.2   7.58       80     35.0
##  7 3/8/2011 2:~      1       0          1       1  9.84 12.1        52      9.00
##  8 3/10/2011 0~      1       0          1       3 13.9  15.9         0     17.0
##  9 4/3/2011 4:~      2       0          0       1 11.5  15.2        70      6.00
## 10 4/4/2011 3:~      2       0          1       1 15.6  19.7        66     19.0
## # i 4 more variables: casual <int>, registered <int>, count <int>,
## #   revenue <dbl>
```

Export Data

```
report = summarise(grouped, Casual = sum(casual), Revenue = sum(revenue))
report
write.csv(report, "revenue_report.csv", row.names = FALSE)
write.table(report, "revenue_report.txt", row.names = FALSE)
```

Exercise Chapter 1 Bike

Load the dataset

```
bike = read.csv("E:/MSc DSc/Sem 1/Business Analytics/Ch2_raw_bikeshare_data.csv")
head(bike,10)
```

```
##         datetime season holiday workingday weather  temp  atemp humidity
## 1  1/1/2011 0:00      1       0          0       1  9.84 14.395       81
## 2  1/1/2011 1:00      1       0          0       1  9.02 13.635       80
## 3  1/1/2011 2:00      1       0          0       1  9.02 13.635       80
## 4  1/1/2011 3:00      1       0          0       1  9.84 14.395       75
## 5  1/1/2011 4:00      1       0          0       1  9.84 14.395       75
## 6  1/1/2011 5:00      1       0          0       2  9.84 12.880       75
## 7  1/1/2011 6:00      1       0          0       1  9.02 13.635       80
## 8  1/1/2011 7:00      1       0          0       1  8.20 12.880       86
## 9  1/1/2011 8:00      1       0          0       1  9.84 14.395       75
## 10 1/1/2011 9:00      1       0          0       1 13.12 17.425       76
##    windspeed casual registered count       sources
## 1     0.0000      3         13    16   ad campaign
## 2     0.0000      8         32    40 www.yahoo.com
## 3     0.0000      5         27    32 www.google.fi
## 4     0.0000      3         10    13   AD campaign
## 5     0.0000      0          1     1       Twitter
## 6     6.0032      0          1     1  www.bing.com
## 7     0.0000      2          0     2   ad campaign
## 8     0.0000      1          2     3 www.yahoo.com
## 9     0.0000      1          7     8 www.yahoo.com
## 10    0.0000      8          6    14  www.bing.com
```

```
str(bike)
```

```
## 'data.frame':    17379 obs. of  13 variables:
##  $ datetime  : chr  "1/1/2011 0:00" "1/1/2011 1:00" "1/1/2011 2:00" "1/1/2011 3:00" ...
##  $ season    : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ holiday   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ workingday: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ weather   : int  1 1 1 1 1 2 1 1 1 1 ...
##  $ temp      : num  9.84 9.02 9.02 9.84 9.84 ...
##  $ atemp     : num  14.4 13.6 13.6 14.4 14.4 ...
##  $ humidity  : chr  "81" "80" "80" "75" ...
##  $ windspeed : num  0 0 0 0 0 ...
```

```
##  $ casual    : int  3 8 5 3 0 0 2 1 1 8 ...
##  $ registered: int  13 32 27 10 1 1 0 2 7 6 ...
##  $ count     : int  16 40 32 13 1 1 2 3 8 14 ...
##  $ sources   : chr  "ad campaign" "www.yahoo.com" "www.google.fi" "AD campaign" ...
```

Tabulate the null values

```
table(is.na(bike))
```

```
##
##  FALSE   TRUE
## 225373    554
```

Finding and fixing flawed data

```
bad_data = str_subset(bike$humidity, '[a-z A-Z]')
bad_data
```

```
## [1] "x61"
```

```
location = str_detect(bike$humidity, bad_data)
bike[location,]
```

```
##              datetime season holiday workingday weather  temp atemp humidity
## 14177 8/18/2012 21:00      3       0          0       1 27.06 31.06      x61
##       windspeed casual registered count      sources
## 14177         0     90        248   338 www.bing.com
```

```
bike$humidity = str_replace_all(bike$humidity, bad_data, "61")
table(is.na(bike))
```

```
##
##  FALSE   TRUE
## 225373    554
```

```
str(bike)
```

```
## 'data.frame':    17379 obs. of  13 variables:
##  $ datetime  : chr  "1/1/2011 0:00" "1/1/2011 1:00" "1/1/2011 2:00" "1/1/2011 3:00" ...
##  $ season    : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ holiday   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ workingday: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ weather   : int  1 1 1 1 1 2 1 1 1 1 ...
##  $ temp      : num  9.84 9.02 9.02 9.84 9.84 ...
##  $ atemp     : num  14.4 13.6 13.6 14.4 14.4 ...
##  $ humidity  : chr  "81" "80" "80" "75" ...
##  $ windspeed : num  0 0 0 0 0 ...
```

```
##  $ casual    : int  3 8 5 3 0 0 2 1 1 8 ...
##  $ registered: int  13 32 27 10 1 1 0 2 7 6 ...
##  $ count     : int  16 40 32 13 1 1 2 3 8 14 ...
##  $ sources   : chr  "ad campaign" "www.yahoo.com" "www.google.fi" "AD campaign" ...
```

```
bike$humidity = as.numeric(bike$humidity)
```

Transform and converting datatypes

```
bike$holiday = factor(bike$holiday, levels = c(0,1),
                      labels = c("no","yes"))
bike$workingday = factor(bike$workingday, levels = c(0,1),
                      labels = c("no","yes"))
bike$weather = factor(bike$weather, levels = c(1,2,3,4),
                      labels = c("clr_part_cloud",
                                 "mist_cloudy",
                                 "lt_rain_snow",
                                 "hcy_rain_snow"),
                      ordered = TRUE)
bike$season = factor(bike$season, levels = c(1,2,3,4),
                      labels = c("spring","summer",
                                 "fall","winter"),
                      ordered = TRUE)
bike$datetime = mdy_hm(bike$datetime)
```

Adapting data to standard

```
unique(bike$sources)
```

```
##  [1] "ad campaign"      "www.yahoo.com"    "www.google.fi"    "AD campaign"
##  [5] "Twitter"          "www.bing.com"     "www.google.co.uk" "facebook page"
##  [9] "Ad Campaign"      "Twitter    "      NA                 "www.google.com"
## [13] "direct"           "blog"
```

```
bike$sources = tolower(bike$sources)
bike$sources = str_trim(bike$sources)
na_loc = is.na(bike$sources)
bike$sources[na_loc] = "unknown"
```

Combining data to new categories

```
web_sites = "(www.[a-z]*.[a-z]*)"
current = unique(str_subset(bike$sources, web_sites))
current
```

```
## [1] "www.yahoo.com"    "www.google.fi"    "www.bing.com"     "www.google.co.uk"
## [5] "www.google.com"
```

```
replace = rep("web", length(current))
replace
```

```
## [1] "web" "web" "web" "web" "web"
```

```
replacements = data.frame(from = current, to = replace)
replacements
```

```
##               from  to
## 1    www.yahoo.com web
## 2    www.google.fi web
## 3     www.bing.com web
## 4 www.google.co.uk web
## 5   www.google.com web
```

```
bike = FindReplace(data = bike, Var = "sources", replacements, from = "from", to = "to", exact = FALSE)
unique(bike$sources)
```

```
## [1] "ad campaign"   "web"           "twitter"       "facebook page"
## [5] "unknown"       "direct"        "blog"
```

```
bike$sources = as.factor(bike$sources)
str(bike$sources)
```

```
##  Factor w/ 7 levels "ad campaign",..: 1 7 7 1 5 7 1 7 7 7 ...
```