README for Getting and Cleaning Data Course Project for Tidy Data

Toby Popenfoose

For this project I used the script *getdataAndUnzip.R* to download the data from Project Data to the local *data* directory.

To complete this project, I used the script *run_analysis.R* to clean and preprocess the data to produce a **independent tidy data set** called *IndependentTidyData.txt*. By sourcing and running the *run_analysis.R* script, you will end up with the tidy data set outputted to the current working directory.

README for run_analysis.R

first we download the data and unzip it into the local ./data directory

```
library(dplyr, warn.conflicts=FALSE)

## see what is there
## assume all the data is in the directory of "./data/UCI HAR Dataset/" from the working directory
list.files("./data", recursive=TRUE)
```

now we read in the feature labels

```
## read in the feature labels (will be used for both test and training data)
FeatureLabels <- read.table("./data/UCI HAR Dataset/features.txt")
names(FeatureLabels) <- c("FeatureID", "Feature")
```

now we read in the test data, its associated activity label codes and its associated subject id codes

```
## read in the obvservations for the test data
TestDataSet <- read.table("./data/UCI HAR Dataset/test/X_test.txt")
## now name the test data variables using the feature labels
names(TestDataSet) <- FeatureLabels$Feature

## read in the activity label for each observation of the test data
TestActivityLabelCode <- read.table("./data/UCI HAR Dataset/test/y_test.txt")
## name the ActivityID variable of the TestActivityLabels
names(TestActivityLabelCode) <- "ActivityID"

## read in the code for which subject was associated with each test observation
TestSubjectCode <- read.table("./data/UCI HAR Dataset/test/subject_test.txt")
## name the SubjectID variable of the TestSubjectsCodes
names(TestSubjectCode) <- "SubjectID"
```

now we read in the training data, its associated activity label codes and its associated subject id codes

```
## read in the obvservations for the training data
TrainDataSet <- read.table("./data/UCI HAR Dataset/train/X_train.txt")
## now name the training data variables using the feature labels
names(TrainDataSet) <- FeatureLabels$Feature

## read in the activity label for each observation of the training data
TrainActivityLabelCode <- read.table("./data/UCI HAR Dataset/train/y_train.txt")
## name the ActivityID variable of the TrainActivityLabels
names(TrainActivityLabelCode) <- "ActivityID"

## read in the code for which subject was associated with each training observation
TrainSubjectCode <- read.table("./data/UCI HAR Dataset/train/subject_train.txt")
## name the SubjectID variable of the TrainSubjectsCodes
names(TrainSubjectCode) <- "SubjectID"
```

we merge the test and training data sets

```
## 1. merge the test and traing data sets to make one data set
DataSet <- rbind(data.frame(TestSubjectCode, TestActivityLabelCode, TestDataSet),
                 data.frame(TrainSubjectCode, TrainActivityLabelCode, TrainDataSet))
```

we extract only the mean and standard deviation variables from the observations

```
## 2. extract only the mean and std of each variable observation
DataSet <- DataSet %>% select(SubjectID, ActivityID, matches("*\\.mean\\.|*\\.std\\."))
```

we read in the activity labels and merge to our main data set

```
## read in the activity labels
ActivityLabels <- read.table("./data/UCI HAR Dataset/activity_labels.txt")
## name the activity variable labels
names(ActivityLabels) <- c("ActivityID", "Activity")
## 3. merge the activity labels to the main data set
DataSet <- merge(DataSet, ActivityLabels)
```

we clean up the labels and use more descriptive variable names

```
## 4. appropriately label the data set with descriptive variable names
## first get the current feature variable names
varNames <- names(DataSet)[-(1:2)]
## change leading t to Time (for time based variables)
```

```
varNames <- sub("^t", "Time", varNames)
## change leading f to Freq (for frequency based variables)
varNames <- sub("^f", "Freq", varNames)
## clean up the doubled
varNames <- sub("BodyBody", "Body", varNames)
## finish cleaning
varNames <- sub(".mean..", "Mean", varNames)
varNames <- sub(".std..", "Std", varNames)
varNames <- sub(".X", "X", varNames)
varNames <- sub(".Y", "Y", varNames)
varNames <- sub(".Z", "Z", varNames)
## now rename the data set with descriptive variable names
names(DataSet)[-(1:2)] <- varNames
```

finally we create an independent tidy data set with the average of each variable for each activity and each subject

```
## 5. create independent tidy data set with average of each variable for each activity and each subject
IndTidyData <- DataSet %>% select(-matches("ActivityID")) %>%
    group_by(SubjectID, Activity) %>% summarise_each(funs(mean))
```

now we write it out in a format for our coursera course

```
## 6. write table of the independent tidy data to text file
write.table(IndTidyData, "IndependentTidyData.txt", row.name=FALSE)
```

Thank you for reading the README!