EXPLANATION

- Language used for analysis: r
- Experiment output file: one csv file “MTMmice.csv”.
  a) Each row records the information of an individual subject.
  b) Columns = {“slot”, “subjectnum”, “group”, “CPP”, “CPA”, “hab-paired”, “hab-unpaired”, “test-paired”, “test-unpaired”}
  c) “group” column records the group and condition they are in.
     There are two groups: CPA and CPP
     There are four conditions in each group: CT4-ON, CT4-OFF, CT11-ON, CT11-OFF.
- First, I will recode variables and clean unwanted columns, creating a new cleaned dataframe containing clean data of variables of interest stored in columns = {“teststim”, “testON”, “testtime”, “prefchange”}.
  This new cleaned dataframe will be used to generate two dataframes based on the variable “teststim”, one for the CPA-stimulus group, the other for the CPP-stimulus group.
- Then I will apply two-way ANOVA to the CPA group and the CPP group of data.
     Independent variable: test time (“4”/”11”), training-testing time match (“ON”/”OFF”)
     Dependent variable: change in preference, which is measured by the change of how many seconds more did the mice spent in the stimulus-paired chamber
  I will apply independent t-test to the cleaned data to compare between groups for the subjects that are test-ON.
     Independent variable: test stimulus (“CPA”/”CPP”)
     Dependent variable: change in preference.
- Lastly, I’ll draw a bar graph for each comparison to visualize the data.

PSEUDOCODE

```
## import required libraries
    library(magrittr)
    library(tidyverse)


## read in csv file containing data
```

```
    data = read_csv("MTMmice.csv")


## Reorganize and recode the data to new columns "teststim" "testON" and "testtime"
# Extract the data stored in data$group and create data$teststim and data$testON based on it
    data$teststim <- sapply(strsplit(as.character(data$group), "[- ]+"), `[`, 1)
    data$testON <- sapply(strsplit(as.character(data$group), "[- ]+"), `[`, 3)
# Recode the data stored in data$CPA and data$CPP to data$CPAtime and data$CPPtime
    data$CPAtime <- sapply(strsplit(as.character(data$CPA), " "), `[`, 3)
    data$CPPtime <- sapply(strsplit(as.character(data$CPP), " "), `[`, 3)
# Create a new column "testtime" and set it equal to either data$CPAtime or data$CPPtime for each
# subject based on the stimulus it is tested for
    data$testtime <- NA
    data$testtime[df$teststim=="CPA"] <- data$CPA[data$teststim=="CPA"]
    data$testtime[df$teststime=="CPP"] <- data$CPP[data$teststim=="CPP"]
# Record how many seconds more did the animal spent in the paired chamber during testing
    data$testpref = data$test-paired – data$test-unpaired
# Record how many seconds more did the animal spent in paired chamber during habituation
    data$habpref = data$hab-paired – data$hab-unpaired
# Record the preference change by subtracting the preference during habituation from the preference
# during testing for each subject.
    data$prefchange = data$testpref – data$habpref


## Generate the new dataframes that will be used for statistical analysis
# Generate a cleaned dataframe that contains all subjects with only the columns storing the variables
    dataclean <- select(df, "teststim", "testON", "testtime", "prefchange")
# Create two new dataframes from the cleaned dataframe containing only the subjects that are tested
# for CPA/CPP
    dataCPA <- select(subset(dataclean, dataclean$teststim == "CPA")
    dataCPP <- subset(dataclean, dataclean$teststim == "CPP")
```

```
## Perform statistical analysis
# Perform two-way ANOVA with no interaction effects on the CPA and CPP group.
    aovCPA <- aov(prefchange ~ testtime + testON, data = dataCPA)
    aovCPP <- aov(prefchange ~ testtime + testON, data = dataCPP)
# Perform independent t-test for all test-ON subjects within the cleaned dataframe.
    tstim <- t.test(prefchange~teststim, data=dataclean[dataclean$testON==”ON”, var.equal =
    TRUE)
# Print out the ANOVA and t-test results.
    summary(aovCPA)
    summary(aovCPP)
    summary(tstim)

## Plot the data
# Plot the CPA groups as bar graph
    ggplot(dataCPA, aes(x = testtime, y = prefchange, colour = testON)) +
    geom_bar() +
    geom_errorbar(aes(prefchange, mean, ymin = mean - sd, ymax = mean + sd)
# Plot the CPP group as bar graph
    ggplot(dataCPP, aes(x = testtime, y = prefchange, colour = testON)) +
    geom_bar +
    geom_errorbar(aes(prefchange, mean, ymin = mean - sd, ymax = mean + sd)
# Plot the the test-ON subjects within the cleaned dataframe as bar graph
    ggplot(dataclean[dataclean$testON==”ON”], aes(x = teststim, y = prefchange)) +
    geom_bar() +
    geom_errorbar(aes(prefchange, mean, ymin = mean - sd, ymax = mean + sd)
```