UP431 Lab2: Exploring NHTS Data (2)

Using NHTS Data (2)

This week, we will explore household data and vehicle data. Get ready with `tidyverse`, `haven` and NHTS SPSS dataset.

```
library(tidyverse)
```

```
## -- Attaching packages ------------------------------------------------------------------------------
```

```
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0
```

```
## Warning: package 'readr' was built under R version 4.0.3
```

```
## -- Conflicts ----------------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(haven)
```

```
## Warning: package 'haven' was built under R version 4.0.3
```

Using Household Data

```
hh <- read_sav("C:/Lab0/2021_UP431/Lab1/Data/spss (2)/hhpub.sav")
```

Begin with exploring what variables household data contains.

```
hh <- as_factor(hh)
names(hh)
```

```
##  [1] "houseid"    "travday"    "sampstrat"  "homeown"    "hhsize"
##  [6] "hhvehcnt"   "hhfaminc"   "pc"         "sphone"     "tab"
## [11] "walk"       "bike"       "car"        "taxi"       "bus"
## [16] "train"      "para"       "price"      "place"      "walk2save"
## [21] "bike2save"  "ptrans"     "hhrelatd"   "drvrcnt"    "cnttdhh"
## [26] "hhstate"    "hhstfips"   "numadlt"    "youngchild" "wrkcount"
## [31] "tdaydate"   "hhresp"     "lif_cyc"    "msacat"     "msasize"
## [36] "rail"       "urban"      "urbansize"  "urbrur"     "scresp"
## [41] "census_d"   "census_r"   "cdivmsar"   "hh_race"    "hh_hisp"
## [46] "hh_cbsa"    "resp_cnt"   "webuse17"   "smplsrce"   "wthhfin"
## [51] "hbhur"      "hthtnrnt"   "htppopdn"   "htresdn"    "hteempdn"
## [56] "hbhtnrnt"   "hbppopdn"   "hbresdn"
```

Filter rows that are from `Chicago-Naperville-Elgin, IL-IN-WI` CBSA.

```
chi_hh <- hh %>% filter(hh_cbsa == "Chicago-Naperville-Elgin, IL-IN-WI")
```

Task 1

What do you think would be the relationship between car ownership and income group? What variable would you use to analyze the relationship?

Like lab 1, use `levels` and `fct_collapse` to see how income group is recorded in the dataset and customize it.

```
levels(chi_hh$hhfaminc)
```

```
##  [1] "I prefer not to answer" "I don't know"           "Not ascertained"
##  [4] "Less than $10,000"      "$10,000 to $14,999"     "$15,000 to $24,999"
##  [7] "$25,000 to $34,999"     "$35,000 to $49,999"     "$50,000 to $74,999"
## [10] "$75,000 to $99,999"     "$100,000 to $124,999"   "$125,000 to $149,999"
## [13] "$150,000 to $199,999"   "$200,000 or more"
```

```
# Your code comes here
chi_hh <- chi_hh %>%
  mutate(
    hhincome_short = fct_collapse(
      hhfaminc,
      "lower" = c(
        "Less than $10,000",
        "$10,000 to $14,999",
        "$15,000 to $24,999",
        "$25,000 to $34,999"
      ),
      "moderate" = c("$35,000 to $49,999"),
      "middle" = c("$50,000 to $74,999"),
      "upper" = c(
        "$75,000 to $99,999",
        "$100,000 to $124,999",
        "$125,000 to $149,999",
        "$150,000 to $199,999",
        "$200,000 or more"
      ),
      Missing = c("I prefer not to answer", "I don't know", "Not ascertained")
    )
  ) %>% filter(hhincome_short != "Missing")
```

a) Calculate the percentage of household with no vehicle for each income group. Don't forget that every analysis should consider weight!

```
# Your code comes here
noVehicle_income <- chi_hh %>%
  count(hhincome_short, hhvehcnt, wt = wthhfin) %>%
  group_by(hhincome_short) %>%
  mutate(per = prop.table(n)*100) %>%
  filter(hhvehcnt == 0)

noVehicle_income
```

```
## # A tibble: 4 x 4
## # Groups:   hhincome_short [4]
##   hhincome_short hhvehcnt       n   per
##   <fct>             <dbl>   <dbl> <dbl>
## 1 lower                 0 292244. 32.7
## 2 moderate              0  24952.  7.64
## 3 middle                0  63686. 10.1
## 4 upper                 0  57768.  4.03
```

Is there any trend?

b) Calculate the mean of vehicle number for each income group. You can use `weighted.mean` to easily calculate the weighted mean.

```
# Your code comes here
meanVehicle_income <- chi_hh %>%
  group_by(hhincome_short) %>%
  summarise(weighted_veh = weighted.mean(hhvehcnt, wthhfin))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
meanVehicle_income
```

```
## # A tibble: 4 x 2
##   hhincome_short weighted_veh
##   <fct>                 <dbl>
## 1 lower                 0.910
## 2 moderate              1.62
## 3 middle                1.78
## 4 upper                 2.13
```

c) Calculate the mean number of vehicle to driver ratio (HHVEHCNT/WRKCOUNT), for each income group.

```
# Your code comes here
meanVehicle2employee_income <- chi_hh %>%
  group_by(hhincome_short) %>%
  summarise(ratio = weighted.mean(hhvehcnt/drvrcnt, wthhfin)) # This arouses Inf! Guess why.
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
meanVehicle2employee_income
```

```
## # A tibble: 4 x 2
##   hhincome_short ratio
##   <fct>          <dbl>
## 1 lower            NaN
## 2 moderate         NaN
## 3 middle           NaN
## 4 upper            NaN
```

Why can't we calculate the ratio with the previous code?

```
# Your code comes here
meanVehicle2employee_income <- chi_hh %>%
  mutate(ratio = ifelse(drvrcnt!=0, hhvehcnt/drvrcnt, hhvehcnt)) %>% # assumed that households with no
  group_by(hhincome_short) %>%
  summarise(meanRatio = weighted.mean(ratio, wthhfin))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

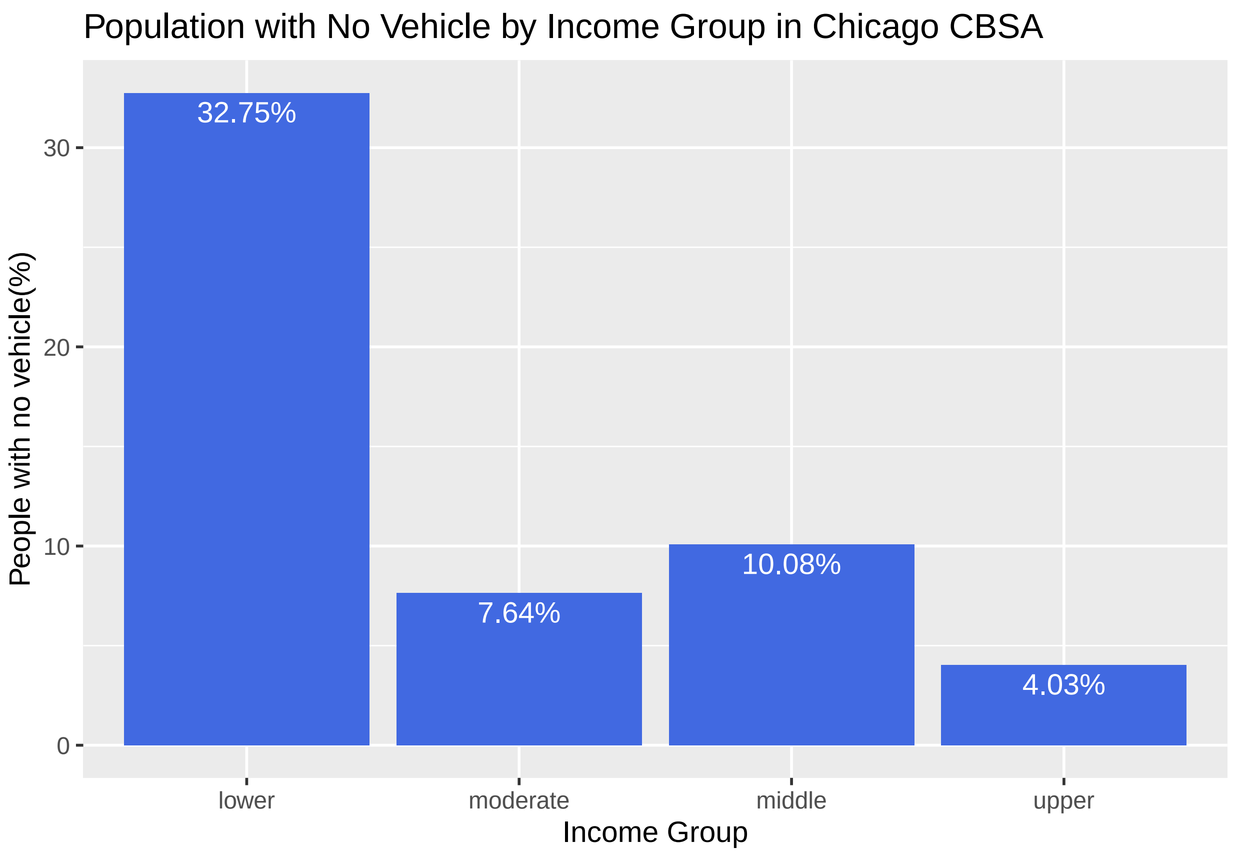
```
meanVehicle2employee_income
```

```
## # A tibble: 4 x 2
##   hhincome_short meanRatio
##   <fct>              <dbl>
## 1 lower              0.728
## 2 moderate           0.960
## 3 middle             0.971
## 4 upper              1.04
```

d) Visualze Task 1 (a) into a bar graph using `geom_bar`.

```
# Your code comes here
noVehicle_income$per <- round(noVehicle_income$per, 2)

ggplot(noVehicle_income, aes(hhincome_short, per)) +
  geom_bar(stat = "identity", fill = "royalblue") +
  labs(x = "Income Group", y = "People with no vehicle(%)", title = "Population with No Vehicle by Inco
  geom_text(
    aes(label = paste0(per, "%"), y = per),
    vjust = 1.4,
    size = 4,
    color = "white"
  )
```



d-1) Crosstab: bivariate frequency tables with percent numbers. Just another way of doing a similar task.

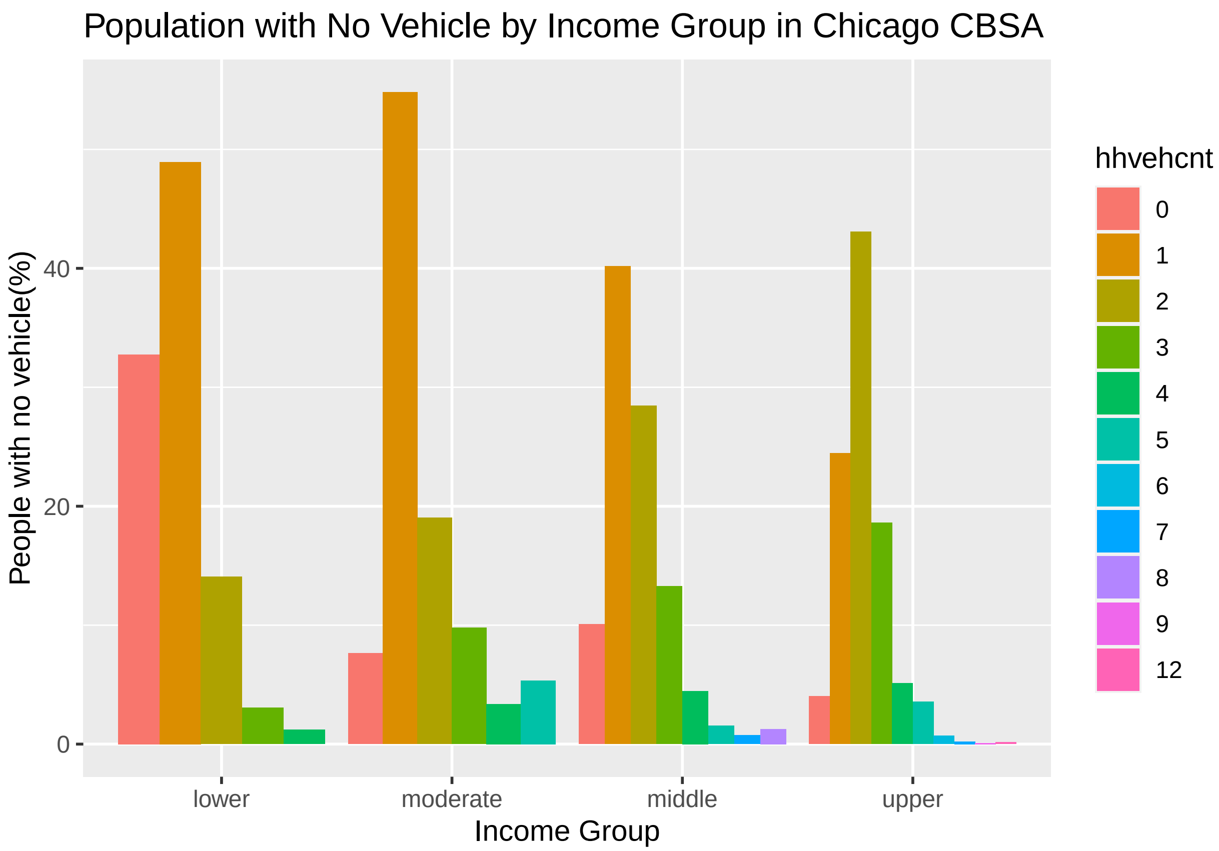
```
#install.packages("pollster")  #a package for survey analysis
library(pollster)
```

```
## Warning: package 'pollster' was built under R version 4.0.3
```

```
crosstab(df = chi_hh,
         x = hhincome_short,
         y = hhvehcnt,
         weight = wthhfin)
```

```
## # A tibble: 4 x 13
##   hhincome_short   `0`   `1`   `2`   `3`   `4`   `5`   `6`   `7`   `8`    `9`
##   <fct>          <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
## 1 lower          32.7   48.9  14.1  3.06  1.20  0    0     0      0    0
## 2 moderate        7.64  54.8  19.0  9.79  3.38  5.35 0     0      0    0
## 3 middle         10.1   40.2  28.4 13.3   4.46  1.55 0     0.755  1.26 0
## 4 upper           4.03  24.4  43.1 18.6   5.12  3.56 0.704 0.211  0    0.0918
## # ... with 2 more variables: `12` <dbl>, n <dbl>
```

```
crosstab(
  df = chi_hh,
  x = hhincome_short,
  y = hhvehcnt,
  weight = wthhfin,
  format = "long"
) %>%
  ggplot(aes(hhincome_short, pct, fill = hhvehcnt)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Income Group", y = "People with no vehicle(%)", title = "Population with No Vehicle by Inco
```



#Using Vehicle Data

Import vehicle data. Explore the variables and filter rows from Chicago-Naperville-Elgin, IL-IN-WI.

```
# Your code comes here
veh <- read_sav("C:/Lab0/2021_UP431/Lab1/Data/spss (2)/vehpub.sav")
veh <- as_factor(veh)
names(veh)
```

```
##  [1] "HOUSEID"   "VEHID"     "VEHYEAR"   "VEHAGE"    "MAKE"      "MODEL"
##  [7] "FUELTYPE"  "VEHTYPE"   "WHOMAIN"   "OD_READ"   "HFUEL"     "VEHOWNED"
## [13] "VEHOWNMO"  "ANNMILES"  "HYBRID"    "PERSONID"  "TRAVDAY"   "HOMEOWN"
## [19] "HHSIZE"    "HHVEHCNT"  "HHFAMINC"  "DRVRCNT"   "HHSTATE"   "HHSTFIPS"
## [25] "NUMADLT"   "WRKCOUNT"  "TDAYDATE"  "LIF_CYC"   "MSACAT"    "MSASIZE"
## [31] "RAIL"      "URBAN"     "URBANSIZE" "URBRUR"    "CENSUS_D"  "CENSUS_R"
## [37] "CDIVMSAR"  "HH_RACE"   "HH_HISP"   "HH_CBSA"   "SMPLSRCE"  "WTHHFIN"
## [43] "BESTMILE"  "BEST_FLG"  "BEST_EDT"  "BEST_OUT"  "HBHUR"     "HTHTNRNT"
## [49] "HTPPOPDN"  "HTRESDN"   "HTEEMPDN"  "HBHTNRNT"  "HBPPOPDN"  "HBRESDN"
## [55] "GSYRGAL"   "GSTOTCST"  "FEGEMPG"   "FEGEMPGA"  "GSCOST"    "FEGEMPGF"
```

```r
chi_veh <- veh %>% filter(HH_CBSA == "Chicago-Naperville-Elgin, IL-IN-WI")
```

Vehicle data consists one row for *each* vehicle. It means that a household with three vehicles will have three rows in the vehicle data.View the dataset and check `HOUSEID` to see what it means!

Task 2

Assume that you need a VMT value in a household level. You would need to aggregate the BESTMILE variable in the vehicle file to a household level using HOUSEID.Before that, make sure that there are no non-numeric values in BESTMILE.

```r
# Your code comes here
chi_veh <- chi_veh %>%
  mutate(BESTMILE_new = fct_collapse(BESTMILE, Missing = c("Not ascertained"))) %>%
  filter(BESTMILE != "Missing")
```

a) Use `aggregate` function. Let's leave HHFAMINC,WTHHFIN for the next task, and also HHSIZE, HBHUR, HHVEHCNT, DRVRCNT for linear regression in the last task.Name the aggregated VMT as `HHVMT`.

```r
# Your code comes here
agg_veh <- aggregate(as.numeric(as.character(BESTMILE_new))~HOUSEID + HHFAMINC + WTHHFIN + HBHUR + HHSI
```

```
## Warning in eval(predvars, data, env):          NA
```

```r
#agg_veh

agg_veh <- agg_veh %>% rename("HHVMT" = "as.numeric(as.character(BESTMILE_new))")
#agg_veh
```

b) Caculate the mean annual household VMT by four income group made in task 1.

```r
# Your code comes here
agg_veh <- agg_veh %>%
  mutate(
    HHINCOME_SHORT = fct_collapse(
      HHFAMINC,
      "lower" = c(
        "Less than $10,000",
        "$10,000 to $14,999",
        "$15,000 to $24,999",
        "$25,000 to $34,999"
      ),
      "moderate" = c("$35,000 to $49,999"),
      "middle" = c("$50,000 to $74,999"),
      "upper" = c(
        "$75,000 to $99,999",
        "$100,000 to $124,999",
        "$125,000 to $149,999",
        "$150,000 to $199,999",
        "$200,000 or more"
      ),
      Missing = c("I prefer not to answer", "I don't know", "Not ascertained")
    )
  ) %>%
  filter(HHINCOME_SHORT != "Missing")
```

```
# Your code comes here
meanHHVMT_income <- agg_veh %>%
  group_by(HHINCOME_SHORT) %>%
  summarise(HHVMT = weighted.mean(HHVMT, WTHHFIN))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
meanHHVMT_income
```

```
## # A tibble: 4 x 2
##   HHINCOME_SHORT  HHVMT
##   <fct>           <dbl>
## 1 lower          11770.
## 2 moderate       16239.
## 3 middle         18674.
## 4 upper          25479.
```

c) Make an ANOVA test to check the relationship between annual household VMT and income group.

```
# Your code comes here
library(car)
```

```
## Warning: package 'car' was built under R version 4.0.3

## Loading required package: carData

## Warning: package 'carData' was built under R version 4.0.3

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode

## The following object is masked from 'package:purrr':
##
##     some
```

```
leveneTest(agg_veh$HHVMT, agg_veh$HHINCOME_SHORT)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##        Df F value  Pr(>F)
## group   3  4.4121 0.00435 **
##       806
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
oneway.test(agg_veh$HHVMT~agg_veh$HHINCOME_SHORT, var.equal=F)
```

```
##
##  One-way analysis of means (not assuming equal variances)
##
## data:  agg_veh$HHVMT and agg_veh$HHINCOME_SHORT
## F = 28.28, num df = 3.00, denom df = 287.17, p-value = 4.731e-16
```

Task 3

Lastly, conduct a linear regression analysis at the household level.

HHVMT = f(HHSIZE, HHVEHCNT/WRKCOUNT, INCGROUP –> dummy, HBHUR –> dummy)

a) Create dummy variables (https://www.marsja.se/create-dummy-variables-in-r/). You can either use a libary to make it fast, or you can make dummy variables by yourself using `ifelse`.

```r
#install.packages("fastDummies")
library(fastDummies)
```

```
## Warning: package 'fastDummies' was built under R version 4.0.3
```

If you want to use `fastDummies`, rename any variable with an empty space(' ') in the name, since you will not be able to access the variable by name if there is a space.

```r
# Your code comes here
agg_veh <- agg_veh %>%
  mutate(HBHUR_new = fct_collapse(
    HBHUR,
    "SmallTown" = c("Small Town"),
    "SecondCity" = c("Second City"),
    "Missing" = c("Not ascertained")
  )) %>%
  filter(HBHUR_new != "Missing")
```

```r
# Your code comes here
agg_veh <- dummy_cols(agg_veh, select_columns = c("HHINCOME_SHORT","HBHUR_new"))
#agg_veh
```

a-2) IF you want to use `ifelse`, here is an example.

```r
#exampleData$Var_A <- ifelse(exampleData$Var == 'A', 1, 0)
#exampleData$Var_B <- ifelse(exampleData$Var == 'B', 1, 0)
```

b) Run a regression and print the result using `summary` function.

```r
# Your code comes here
HH_model <- lm(HHVMT ~ HHSIZE + HHVEHCNT/DRVRCNT + HHINCOME_SHORT_lower + HHINCOME_SHORT_moderate + HHI

summary(HH_model)
```

```
##
## Call:
## lm(formula = HHVMT ~ HHSIZE + HHVEHCNT/DRVRCNT + HHINCOME_SHORT_lower +
##     HHINCOME_SHORT_moderate + HHINCOME_SHORT_upper + HBHUR_new_Urban +
##     HBHUR_new_SmallTown + HBHUR_new_Suburban + HBHUR_new_Rural,
##     data = agg_veh, weights = WTHHFIN)
##
## Weighted Residuals:
##      Min       1Q   Median       3Q      Max
## -2390914  -352590   -74540   171146  7033839
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)               3938.2     1953.8   2.016   0.0442 *
## HHSIZE                     706.6      399.3   1.770   0.0772 .
## HHVEHCNT                  6343.8      840.9   7.544 1.24e-13 ***
## HHINCOME_SHORT_lower     -1708.8     1363.7  -1.253   0.2105
## HHINCOME_SHORT_moderate   -751.9     1632.4  -0.461   0.6452
## HHINCOME_SHORT_upper      5036.2     1138.8   4.422 1.11e-05 ***
## HBHUR_new_Urban          -2207.9     1281.4  -1.723   0.0853 .
## HBHUR_new_SmallTown        742.4     1681.2   0.442   0.6589
## HBHUR_new_Suburban        -427.6     1228.0  -0.348   0.7277
## HBHUR_new_Rural            670.0     3009.1   0.223   0.8238
## HHVEHCNT:DRVRCNT           234.8      184.8   1.271   0.2043
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 713500 on 799 degrees of freedom
## Multiple R-squared:  0.4338, Adjusted R-squared:  0.4267
## F-statistic: 61.22 on 10 and 799 DF,  p-value: < 2.2e-16
```

Do you see any significant relationship?

c) What additional variables do you want to include in the regression?

```
# Your code comes here
```