```
/* clearing log,output,results*/
dm 'clear log'; dm 'clear output'; dm 'odsresults; clear';
/* assigning library */
libname data "E:\pcg180000\final"; title;

/* DATA IMPORT */
proc import out = data.vehicles_raw
            datafile = "E:\pcg180000\final\craigslistVehiclesFull.csv"
            dbms = tab replace;
            delimiter = ",";
            getnames = yes;
            datarow = 2;
run;

/* copying data to work library */
proc datasets; copy in = data out = work; select vehicles_raw; run;

/* DATA CLEANING */
proc contents data = vehicles_raw varnum; run;

/* changing formats for numeric variables */
data vehicles_raw;
     set vehicles_raw;
     price_new = input(price,7.);
     year_new = input(year,6.);
     odometer_new = input(odometer,8.);
     drop price;
     drop year;
     drop odometer;
     rename price_new = price;
     rename year_new = year;
     rename odometer_new = odometer;
run;

/* numeric variables distribution */
proc univariate data = vehicles_raw; var price year odometer; histogram; inset n mean std min max; run

/* removing illogical values */
proc sql;
     create table vehicles_tmp as
     select price,year,fuel,title_status,transmission,manufacturer,
            odometer,condition,cylinders,drive,size,type,paint_color
     from vehicles_raw
     where (year > 1950 or year = .)
     and ((price > 100 and price < 70000) or (price = .))
     and ((odometer > 500 and odometer < 500000) or (odometer = .));
quit;

/* converting year into age */
data vehicles_tmp;
     set vehicles_tmp;
     age = 2020-year;
     drop year;
run;

/* numeric variable logarithms */
```

```
data vehicles_tmp;
     set vehicles_tmp;
     log_price = log(price);
     log_age = log(age);
     log_odometer = log(odometer);
run;

/* numeric variables distribution */
proc univariate data = vehicles_tmp; var price age odometer; histogram; inset n mean std min max; run;
proc univariate data = vehicles_tmp; var log_price log_age log_odometer; histogram; inset n mean std m

/* categorical variables frequency */
proc freq data = vehicles_tmp;
          tables fuel title_status transmission manufacturer condition cylinders drive size type paint_
run;

/* correcting errors and categorizing less frequent manufacturers as other */
proc sql; update vehicles_tmp set manufacturer = 'chevrolet' where manufacturer = 'chev'; quit;
proc sql; update vehicles_tmp set manufacturer = 'chevrolet' where manufacturer = 'chevy'; quit;
proc sql; update vehicles_tmp set manufacturer = 'infiniti' where manufacturer = 'infinity'; quit;
proc sql; update vehicles_tmp set manufacturer = 'mercedes' where manufacturer = 'mercedes-be'; quit;
proc sql; update vehicles_tmp set manufacturer = 'mercedes' where manufacturer = 'mercedesben'; quit;
proc sql; update vehicles_tmp set manufacturer = 'volkswagen' where manufacturer = 'vw'; quit;
proc sql; update vehicles_tmp set manufacturer = 'other'
          where manufacturer in ('alfa','alfa-romeo','aston','aston-marti','datsun','ferrari','harley'
                                 'harley-davi','hennessey','landrover','land rover','morgan','noble','
quit;

/* dataset with price,age and 4 categorical variables */
proc sql;
     create table vehicles_4cat as
     select price,age,log_price,log_age,fuel,title_status,transmission,manufacturer from vehicles_tmp;
quit;

/* dataset with price,age,odometer and 9 categorical variables */
proc sql;
     create table vehicles_9cat as
     select price,age,odometer,log_price,log_age,log_odometer,
            fuel,title_status,transmission,manufacturer,
            condition,cylinders,drive,type,paint_color
     from vehicles_tmp;
quit;

/* removing all rows with missing values */
data vehicles_4cat; set vehicles_4cat; if cmiss(of _all_) then delete; run;
data vehicles_9cat; set vehicles_9cat; if cmiss(of _all_) then delete; run;

/* backup data to own library */
proc datasets; copy in = work out = data; select vehicles_4cat vehicles_9cat; run;
proc export data = vehicles_4cat outfile = "E:\pcg180000\final\vehicles_4cat.csv" dbms = csv replace;
proc export data = vehicles_9cat outfile = "E:\pcg180000\final\vehicles_9cat.csv" dbms = csv replace;

/* EXPLORATORY DATA ANALYSIS */
proc univariate data = vehicles_4cat;
                var log_price log_age;
                histogram / normal kernel;
                qqplot / normal(mu=est sigma=est);
                inset n mean std; run;
proc univariate data = vehicles_9cat;
                var log_price log_age log_odometer;
                histogram / normal kernel;
                qqplot / normal(mu=est sigma=est);
                inset n mean std; run;

/* price distribution by categories */
proc univariate data = vehicles_4cat; class fuel; var log_price; histogram; run;
proc univariate data = vehicles_4cat; class title_status; var log_price; histogram; run;
proc univariate data = vehicles_4cat; class transmission; var log_price; histogram; run;

proc univariate data = vehicles_9cat; class condition; var log_price; histogram; run;
proc univariate data = vehicles_9cat; class cylinders; var log_price; histogram; run;
proc univariate data = vehicles_9cat; class drive; var log_price; histogram; run;

proc freq data = vehicles_tmp;
          tables fuel title_status transmission manufacturer condition cylinders drive type paint_colo
                 / plots(only)=freqplot(scale=percent);
run;

/* INDEPENDENT VARIABLE CORRELATIONS */
/* pearson correlations */
proc corr data = vehicles_4cat nosimple; var log_price log_age; run;
proc corr data = vehicles_9cat nosimple; var log_price log_age log_odometer; run;

/* chi square correlations */
proc freq data = vehicles_4cat; tables fuel*(title_status transmission manufacturer) / chisq; run;
proc freq data = vehicles_4cat; tables title_status*(transmission manufacturer) / chisq; run;
proc freq data = vehicles_4cat; tables transmission*manufacturer / chisq; run;

proc freq data = vehicles_9cat; tables condition*(cylinders drive type paint_color) / chisq; run;
proc freq data = vehicles_9cat; tables cylinders*(drive type paint_color) / chisq; run;
proc freq data = vehicles_9cat; tables drive*(type paint_color) / chisq; run;
proc freq data = vehicles_9cat; tables type*paint_color / chisq; run;

/* anova test between age,odometer and others */
proc glm data=vehicles_4cat; class fuel; model log_age=fuel; lsmeans fuel/adjust=tukey; run;
proc glm data=vehicles_4cat; class title_status; model log_age=title_status; lsmeans title_status/adju
proc glm data=vehicles_4cat; class transmission; model log_age=transmission; lsmeans transmission/adju
proc glm data=vehicles_4cat; class manufacturer; model log_age=manufacturer; lsmeans manufacturer/adju

proc glm data=vehicles_9cat; class condition; model log_age=condition; lsmeans condition/adjust=tukey;
proc glm data=vehicles_9cat; class cylinders; model log_age=cylinders; lsmeans cylinders/adjust=tukey;
proc glm data=vehicles_9cat; class drive; model log_age=drive; lsmeans drive/adjust=tukey; run;
proc glm data=vehicles_9cat; class type; model log_age=type; lsmeans type/adjust=tukey; run;
proc glm data=vehicles_9cat; class paint_color; model log_age=paint_color; lsmeans paint_color/adjust=

proc glm data=vehicles_9cat; class condition; model log_odometer=condition; lsmeans condition/adjust=t
proc glm data=vehicles_9cat; class cylinders; model log_odometer=cylinders; lsmeans cylinders/adjust=t
proc glm data=vehicles_9cat; class drive; model log_odometer=drive; lsmeans drive/adjust=tukey; run;
proc glm data=vehicles_9cat; class type; model log_odometer=type; lsmeans type/adjust=tukey; run;
proc glm data=vehicles_9cat; class paint_color; model log_odometer=paint_color; lsmeans paint_color/ad

/* GENERALIZED LINEAR REGRESSION */
```

```
ods graphics on;
proc glm data = vehicles_4cat plots(only)=(contourfit);
         class fuel(ref='other') title_status(ref='parts onl') transmission(ref='other');
         model log_price = log_age fuel title_status transmission / solution;
run;
proc glm data = vehicles_4cat plots(only)=(contourfit);
         class manufacturer(ref='other');
         model log_price = log_age manufacturer / solution;
run;
proc glm data = vehicles_9cat plots(only)=(contourfit);
         class condition(ref='salvage') cylinders(ref='other') drive(ref='4wd');
         model log_price = log_age log_odometer condition cylinders drive / solution;
run;
proc glm data = vehicles_9cat plots(only)=(contourfit);
         class type(ref='other') paint_color(ref='custom');
         model log_price = log_age log_odometer type paint_color / solution;
run;
ods graphics off;
quit;
```