```
/* This code was used to group the data by weekday to find the average number of
    available bikes and changes to be used in the demand calculation. */

/* Connect to the CAS Server - to access the files in the public folder */
cas mySession2 host="localhost" port=5570 sessopts=(caslib=casuser timeout=18000);
caslib _all_ assign;

/* Increase system data limit to prevent restrictions */
options CASDATALIMIT=ALL;

/* Duplicating and sorting the hour data with added variables by address and then by update time */
proc sort data=PUBLIC.ZZZ_BIKE_HOUR_WEIGHTS2 out=work.WEEKDAY equals;
    by address Date_Time;
run;

/* Creating new column for weekday */
data work.WEEKDAY1;
    set work.WEEKDAY;
    by address Date_Time;
    /* Using weekday function to get weekday number from date column */
    DayofWeek = Weekday('Date'n);
run;

/* Creating new column for weekday names */
data work.WEEKDAY2;
    set work.WEEKDAY1;
    /* Creating column for weekday name based on number in weekday column */
    if DayofWeek = 1 then DayWeek_proc = "Sun";
    if DayofWeek = 2 then DayWeek_proc = "Mon";
    if DayofWeek = 3 then DayWeek_proc = "Tues";
    if DayofWeek = 4 then DayWeek_proc = "Wed";
    if DayofWeek = 5 then DayWeek_proc = "Thur";
    if DayofWeek = 6 then DayWeek_proc = "Fri";
    if DayofWeek = 7 then DayWeek_proc = "Sat";
run;

/* Renaming and promoting the weekday preparation data to the public server */
proc casutil;
    load data=work.WEEKDAY2 outcaslib="public" promote
    casout="ZZZ_Weekday_Prep2";
run;

/* Using SQL to group data by hour and day of week */
proc sql;
    /* Creating new table to be grouped */
    create table work.testGroup as
    /* Selecting only the columns that are needed */
    select Time_Hour, address,
    /* Calculating and saving average values by hour and weekday */
    AVG(Available_Bikes) AS Avg_Available_Bikes,
    AVG(Avail_Bike_Changes) AS Avg_Bike_Change,
    DayofWeek, DayWeek_proc
    /* Calling in the newly created weekday preparation dataset */
    from Public.ZZZ_Weekday_Prep2
    /* Grouping the data by address, hour and day of week */
    group by Time_Hour, address, DayofWeek, DayWeek_proc;
quit;

/* Re-sort grouped data by address and then by hour */
proc sort data=work.testGroup out=work.testGroup1 equals;
    by address Time_Hour;
```

```
run;

/* Create new integer rounded average values */
data WORK.testGroup2;
    set WORK.testGroup1;
    by address Time_Hour;
    /* Round average available bikes to an integer in new column */
    Avg_Avail_Bikes = round(Avg_Available_Bikes,1);
    /* Round average available bike changes to an integer in new column */
    Avg_Avail_Bike_Change = round(Avg_Bike_Change,1);
run;

/* Remove unnecessary columns */
data WORK.testGroup3;
    set WORK.testGroup2 (drop = Avg_Available_Bikes Avg_Bike_Change);
run;

/* Renaming and promoting the grouped weekday data to the public server */
proc casutil;
    load data=work.testGroup3 outcaslib="public" promote
    casout="ZZZ_Weekday_Data2";
run;
```