

```
/* This code was used to group the newly created API data file with added
   activity columns into hours and to calculate the availability percentage
   for each station at every hour. */

/* 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 activity data by the update time */
proc sort data=PUBLIC.ZZZ_BIKE_ACTIVITY_DATA out=work.hourDS equals;
  by last_update;
run;

/* Duplicating time column - taking only the hour value */
data WORK.HOURDS1;
  set WORK.HOURDS;
  Time_Hour = hms(HOUR('Time'n),0,0); /* Copying hour values - setting minutes and seconds to 0 */
  format Time_Hour time16.; /* Formatting new column as SAS time values */
run;

/* Sorting data by station name then by date and then by time */
proc sort data=WORK.HOURDS1 out=WORK.HOURDS2 equals;
  by 'name'n 'Date'n 'Time'n;
run;

/* Creating new columns of nulls for first update in an hour per station */
data WORK.HOURDS3;
  set WORK.HOURDS2;
  by name Date Time_Hour;
  /* First update per hour - change null value to actual value for bikes and stands */
  if First.Time_Hour then do;
    First_Available_Bikes = available_bikes;
    First_Bike_Stands = available_bike_stands;
  end;
run;

/* Renaming and promoting the hour preparation data to the public server */
proc casutil;
  load data=work.HOURDS3 outcaslib="public" promote
  casout="ZZZ_Bike_Hour_Prep";
run;

/* Using SQL to group data by hour */
proc sql;
  /* Creating new table to be grouped */
  create table work.testBike as
  /* Selecting only the columns that are needed */
  select 'Date'n, Time_Hour, 'name'n,
  MAX(bike_stands) AS bike_stands,
  /* Taking the largest or non-null value as available bikes and stands */
  MAX(First_Available_Bikes) AS Available_Bikes,
  MAX(First_Bike_Stands) AS Available_Bike_Stands,
  /* Sum changes in bikes in an hour for a station */
  SUM(X_Avail_Bike_Changes) AS Avail_Bike_Changes,
  SUM(Avail_Bikes_Decrease) AS Avail_Bikes_Decrease,
  SUM(Avail_Bikes_Increase) AS Avail_Bikes_Increase,
  /* Sum changes in stands in an hour for a station */
  SUM(X_Avail_Bike_Stand_Changes) AS Avail_Bike_Stand_Changes,
```

```
SUM(Avail_Stands_Decrease) AS Avail_Stands_Decrease,
SUM(Avail_Stands_Increase) AS Avail_Stands_Increase,
address, banking, Elevation, Latitude, Longitude, 'number'n,
CASE WHEN 'status'n = "OPEN" THEN "Open" ELSE "Closed" END AS Status
from PUBLIC.ZZZ_Bike_Hour_Prep /* Calling in the newly created hour preparation dataset */
/* Using the non-changing variables to group the data */
group by 'Date'n, Time_Hour, 'name'n, address, banking, Elevation,
Latitude, Longitude, 'number'n, 'status'n;
quit;

/* Adding availability and datetime columns */
data WORK.testBike2;
  set WORK.testBike;
  by Date Time_Hour;
  /* New column for availability */
  /* Percentage of available bikes per number of stands at a station */
  Availability = Available_Bikes/bike_stands;
  /* New combined column with hour and date */
  Date_Time = dhms(Date, 0, 0, Time_Hour);
  /* Reformatting new column to the SAS datetime format */
  format Date_Time NLDATM30.;
run;

/* Renaming and promoting the hour grouped data to the public server */
proc casutil;
  load data=work.testBike2 outcaslib="public" promote
  casout="ZZZ_Bike_Hour_Data";
run;
```