MTA Map:

<http://web.mta.info/maps/submap.html>

MTA Data:

<http://web.mta.info/developers/turnstile.html>

10 volunteers 4 hours/day

Summer, so assume June, July, and August

1 4-hour shift or 2 2-hour shifts

Seven days in a row

Target: Find the busiest, 7 day span during the summer, at a commuter stop

* Program - Cut out unwanted stops

def Stations(MTA):

MTA.columns = MTA.columns.str.strip()

monday = MTA[(MTA.STATION == '59 ST') | (MTA.STATION =='57 ST') | (MTA.STATION == '5 AVE') | (MTA.STATION == '21 ST-QNSBRIDGE') | (MTA.STATION == '23 ST') | (MTA.STATION =='42 ST-PORT AUTH') | (MTA.STATION == 'CYPRESS AV') | (MTA.STATION == 'YORK ST') | (MTA.STATION == 'BOROUGH HALL') | (MTA.STATION == 'HOYT ST') | (MTA.STATION == 'NEVINS ST')]

monday = monday.reset\_index(drop=True)

return(monday)

def givestations(MTA):

linenames=[]

for linename in MTA.STATION:

linenames.append(linename)

return set(linenames)

* Program - Cut time to 8a-8p

def time(MTA)

morning = MTA[(MTA.TIME == '05:00:00') | (MTA.TIME == '06:00:00') |

(MTA.TIME == '07:00:00') | (MTA.TIME == '08:00:00') |

(MTA.TIME == '09:00:00') | (MTA.TIME == '10:00:00') |

(MTA.TIME == '11:00:00') | (MTA.TIME == '12:00:00') |

(MTA.TIME == '13:00:00') | (MTA.TIME == '14:00:00') |

(MTA.TIME == '15:00:00') | (MTA.TIME == '16:00:00') |

(MTA.TIME == '17:00:00') | (MTA.TIME == '18:00:00') |

(MTA.TIME == '19:00:00') | (MTA.TIME == '20:00:00')]

morning = morning.reset\_index(drop=True)

return(morning)

* (C/A, UNIT, STATION) maps to an inner dictionary containing {SCP: Morning Exit Differences}

def morningdif(mordict):

morning\_differences = {}

for key,value in morning\_dict.items():

newkey = (key[0],key[1],key[3])

val = []

SCP = key[2]

for key1, val1 in value.items():

val.append(val1)

dif = val[1]-val[0]

if newkey not in morning\_differences:

morning\_differences[newkey] = {SCP : dif}

else:

morning\_differences[newkey][SCP] = dif

return(morning\_differences)

Total number of turnstiles (Q21):

def totalnum (morningdif):

total\_morning={}

for key1, value1 in morning\_differences.items():

sum1=0

for scp, diff in value.items():

sum1=sum1+diff

total\_morning[key2]=total\_morning.get(key1,0)+sum1

Return (total\_morning)

Sorted tuple（Q22）:

def rank(totalnum)

list1=[]

for key2,value2 in total\_morning.items():

if key2 not in list1:

list1.append((value2,key2))

list1.sort(reverse=True)

return(list1)

* Program - Count riders at a stop, sort by time (Tara)

import pandas as pd

def Stations(MTA):

MTA.columns = MTA.columns.str.strip()

monday = MTA[(MTA.STATION == '59 ST') | (MTA.STATION =='57 ST') | (MTA.STATION == '5 AVE') | (MTA.STATION == '21 ST-QNSBRIDGE') | (MTA.STATION == '23 ST') | (MTA.STATION =='42 ST-PORT AUTH') | (MTA.STATION == 'CYPRESS AV') | (MTA.STATION == 'YORK ST') | (MTA.STATION == 'BOROUGH HALL') | (MTA.STATION == 'HOYT ST') | (MTA.STATION == 'NEVINS ST')]

monday = monday.reset\_index(drop=True)

return(monday)

Stations(pd.read\_csv('turnstile.csv'))

x=Stations(pd.read\_csv('turnstile.csv')).groupby(['C/A', 'UNIT', 'SCP', 'STATION']).groups

flds = ['TIME','EXITS']

monday\_exits = {key:Stations(pd.read\_csv('turnstile.csv')).loc[val,flds] for key, val in x.items()}

monday\_exits[('A002', 'R051', '02-00-00','59 ST')] **NEEDS DEBUGGING**

* Program - Calculate stops and schedules for the most effective method (Map in a tuple as (number of riders, station, time))

Busiest stops close to large number of employee of tech company.

Colleges

Station:

59 ST

57 ST

5 AVE

21 ST-QNSBRIDGE

23 ST

42 ST-PORT AUTH

CYPRESS AV

YORK ST

BOROUGH HALL

HOYT ST

NEVINS ST

**Top 10 biggest tech companies in NYC:**

Bloomberg: - 59 ST.

Oath: Astor Place

CA Technologies: 57 Street Subway Station, 5 Av Subway Station

Vice Media: 21 St- Queensbridge

Blue Apron - 23 Street

E\*Trade - 42 st port authority bus terminal

BuzzFeed - 23 street

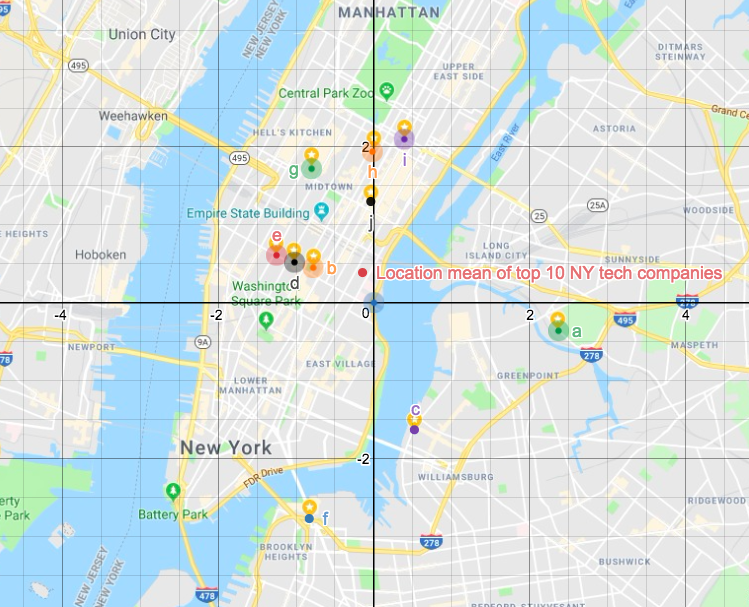
Yext - East 23rd

FreshDirect - Cypress Avenue

Etsy - York Street Station

New York College of Technology - Borough Hall, Hoyt St, Nevins St

NYU - West 4th Street



COMPLETE FUNC:

import pandas as pd

def Stations(MTA):

MTA.columns = MTA.columns.str.strip()

monday = MTA[(MTA.STATION == '59 ST') | (MTA.STATION =='57 ST') | (MTA.STATION == '5 AVE') | (MTA.STATION == '21 ST-QNSBRIDGE') | (MTA.STATION == '23 ST') | (MTA.STATION =='42 ST-PORT AUTH') | (MTA.STATION == 'CYPRESS AV') | (MTA.STATION == 'YORK ST') | (MTA.STATION == 'BOROUGH HALL') | (MTA.STATION == 'HOYT ST') | (MTA.STATION == 'NEVINS ST')]

monday = monday.reset\_index(drop=True)

return(monday)

def time(MTA):

morning=MTA[(MTA.TIME=='05:00:00')|(MTA.TIME=='12:00:00')|(MTA.TIME=='06:00:00')|

(MTA.TIME=='07:00:00')|(MTA.TIME=='08:00:00')|(MTA.TIME=='09:00:00')|

(MTA.TIME=='10:00:00')|(MTA.TIME=='11:00:00')|(MTA.TIME=='12:00:00')]

morning=morning.reset\_index(drop=True)

return(morning)

stations = Stations(pd.read\_csv('june17.csv'))

times = time(stations)

def time(MTA):

evening=MTA[(MTA.TIME=='12:00:00')|(MTA.TIME=='13:00:00')|(MTA.TIME=='14:00:00')|

(MTA.TIME=='15:00:00')|(MTA.TIME=='16:00:00')|(MTA.TIME=='17:00:00')|

(MTA.TIME=='18:00:00')|(MTA.TIME=='19:00:00')|(MTA.TIME=='20:00:00')]

times=time(stations)

def date(MTA):

MTA=MTA[MTA.DATE=='06/10/2017']

MTA=MTA.reset\_index(drop=True)

return(MTA)

droped=date(times)

def dic(MTA):

MTA\_dict={}

for i in range(len(MTA.index)):

tuple1 = ((MTA['C/A'][i], MTA.UNIT[i], MTA.SCP[i], MTA.STATION[i]))

MTA\_dict[tuple1]=MTA\_dict.get(tuple1,{})

MTA\_dict[tuple1].update({MTA.TIME[i]:MTA.EXITS[i]})

return(MTA\_dict)

dic1=dic(droped)

def delete(MTA):

to\_delete = []

for key,value in MTA.items():

if len(value)!=2:

to\_delete.append(key)

for i in to\_delete:

del MTA[i]

return(MTA)

deleted\_dict=delete(dic1)

def morningdif(mordict):

morning\_differences = {}

for key,value in mordict.items():

newkey = (key[0],key[1],key[3])

val = []

SCP = key[2]

for key1, val1 in value.items():

val.append(val1)

dif = val[1]-val[0]

if newkey not in morning\_differences:

morning\_differences[newkey] = {SCP : dif}

else:

morning\_differences[newkey][SCP] = dif

return(morning\_differences)

morningdiff=morningdif(deleted\_dict)

def totalnum (morningdif):

total\_morning\_diff={}

for key,value in morningdif.items():

total=[]

for key1,value1 in value.items():

total.append(value1)

sum1=sum(total)

total\_morning\_diff[key]=total\_morning\_diff.get(key,0)+sum1

return (total\_morning\_diff)

total=totalnum(morningdiff)

def rank(totalnum):

list1=[]

for key2,value2 in totalnum.items():

if key2 not in list1:

list1.append((value2,key2))

list1.sort(reverse=True)

return(list1)

ranked=rank(total)

print(ranked)

6/10: Sat

[(3401, ('N063A', 'R011', '42 ST-PORT AUTH')), (2088, ('R244', 'R050', '59 ST')), (1994, ('R601A', 'R108', 'BOROUGH HALL')), (1571, ('N062', 'R011', '42 ST-PORT AUTH')), (1512, ('N077', 'R111', '23 ST')), (1503, ('R501', 'R054', '5 AVE')), (1423, ('N062A', 'R010', '42 ST-PORT AUTH')), (1362, ('N600', 'R302', '57 ST')), (1290, ('R244A', 'R050', '59 ST')), (1290, ('A031', 'R083', '23 ST')), (1232, ('N076', 'R111', '23 ST')), (1232, ('A002', 'R051', '59 ST')), (1180, ('N060', 'R010', '42 ST-PORT AUTH')), (1100, ('A030', 'R083', '23 ST')), (1077, ('N509', 'R203', '23 ST')), (935, ('R605', 'R456', 'HOYT ST')), (872, ('R245A', 'R051', '59 ST')), (856, ('N603', 'R303', '21 ST-QNSBRIDGE')), (850, ('R226', 'R131', '23 ST')), (845, ('N063', 'R011', '42 ST-PORT AUTH')), (705, ('R609', 'R056', 'NEVINS ST')), (589, ('N530', 'R301', 'YORK ST')), (584, ('R245', 'R051', '59 ST')), (561, ('C021', 'R212', '59 ST')), (525, ('R132', 'R190', '23 ST')), (378, ('R606', 'R225', 'HOYT ST')), (332, ('R608', 'R056', 'NEVINS ST')), (244, ('N075', 'R111', '23 ST')), (240, ('C001', 'R108', 'BOROUGH HALL')), (229, ('R602', 'R108', 'BOROUGH HALL')), (213, ('R226A', 'R131', '23 ST')), (186, ('R405', 'R447', 'CYPRESS AV')), (167, ('C022', 'R212', '59 ST')), (155, ('R404', 'R447', 'CYPRESS AV')), (109, ('R227A', 'R131', '23 ST')), (57, ('R604', 'R108', 'BOROUGH HALL')), (33, ('R227', 'R131', '23 ST')), (5, ('N508', 'R453', '23 ST')), (0, ('R131', 'R190', '23 ST'))] **Evening**: [(3644, ('N062A', 'R010', '42 ST-PORT AUTH')), (2621, ('N060', 'R010', '42 ST-PORT AUTH')), (2310, ('N600', 'R302', '57 ST')), (2177, ('N530', 'R301', 'YORK ST')), (1662, ('R226', 'R131', '23 ST')), (1629, ('R132', 'R190', '23 ST')), (755, ('R226A', 'R131', '23 ST')), (593, ('R227A', 'R131', '23 ST')), (173, ('R404', 'R447', 'CYPRESS AV')), (164, ('R405', 'R447', 'CYPRESS AV')), (121, ('R227', 'R131', '23 ST')), (0, ('R131', 'R190', '23 ST'))]

6/11:Sun

[(2470, ('N063A', 'R011', '42 ST-PORT AUTH')), (1903, ('R244', 'R050', '59 ST')), (1203, ('N062', 'R011', '42 ST-PORT AUTH')), (1193, ('R501', 'R054', '5 AVE')), (1174, ('R244A', 'R050', '59 ST')), (1037, ('R601A', 'R108', 'BOROUGH HALL')), (970, ('N077', 'R111', '23 ST')), (954, ('N060', 'R010', '42 ST-PORT AUTH')), (941, ('R245A', 'R051', '59 ST')), (937, ('N062A', 'R010', '42 ST-PORT AUTH')), (934, ('N509', 'R203', '23 ST')), (924, ('N600', 'R302', '57 ST')), (910, ('A031', 'R083', '23 ST')), (890, ('A002', 'R051', '59 ST')), (810, ('R605', 'R456', 'HOYT ST')), (790, ('A030', 'R083', '23 ST')), (656, ('N076', 'R111', '23 ST')), (644, ('N603', 'R303', '21 ST-QNSBRIDGE')), (641, ('N063', 'R011', '42 ST-PORT AUTH')), (552, ('R245', 'R051', '59 ST')), (476, ('C021', 'R212', '59 ST')), (386, ('R609', 'R056', 'NEVINS ST')), (366, ('R602', 'R108', 'BOROUGH HALL')), (356, ('R226', 'R131', '23 ST')), (339, ('R606', 'R225', 'HOYT ST')), (315, ('R608', 'R056', 'NEVINS ST')), (275, ('R132', 'R190', '23 ST')), (183, ('C001', 'R108', 'BOROUGH HALL')), (153, ('N530', 'R301', 'YORK ST')), (148, ('C022', 'R212', '59 ST')), (147, ('N075', 'R111', '23 ST')), (112, ('R604', 'R108', 'BOROUGH HALL')), (81, ('R405', 'R447', 'CYPRESS AV')), (70, ('R404', 'R447', 'CYPRESS AV')), (67, ('R226A', 'R131', '23 ST')), (66, ('R131', 'R190', '23 ST')), (55, ('R227A', 'R131', '23 ST')), (38, ('R227', 'R131', '23 ST')), (2, ('N508', 'R453', '23 ST'))] **Evening**: [(3232, ('N062A', 'R010', '42 ST-PORT AUTH')), (2366, ('N600', 'R302', '57 ST')), (2348, ('N060', 'R010', '42 ST-PORT AUTH')), (1439, ('R132', 'R190', '23 ST')), (1339, ('R226', 'R131', '23 ST')), (1238, ('N530', 'R301', 'YORK ST')), (566, ('R131', 'R190', '23 ST')), (441, ('R227A', 'R131', '23 ST')), (408, ('R226A', 'R131', '23 ST')), (157, ('R227', 'R131', '23 ST')), (149, ('R404', 'R447', 'CYPRESS AV')), (129, ('R405', 'R447', 'CYPRESS AV'))]

6/12:Mon

[(5856, ('R245A', 'R051', '59 ST')), (5020, ('N509', 'R203', '23 ST')), (4751, ('N508', 'R453', '23 ST')), (4440, ('R244', 'R050', '59 ST')), (4108, ('R244A', 'R050', '59 ST')), (4094, ('N600', 'R302', '57 ST')), (3775, ('R501', 'R054', '5 AVE')), (3765, ('R601A', 'R108', 'BOROUGH HALL')), (3628, ('N062A', 'R010', '42 ST-PORT AUTH')), (3481, ('N060', 'R010', '42 ST-PORT AUTH')), (3040, ('A031', 'R083', '23 ST')), (2896, ('R602', 'R108', 'BOROUGH HALL')), (2828, ('A030', 'R083', '23 ST')), (2804, ('A002', 'R051', '59 ST')), (2692, ('R226', 'R131', '23 ST')), (2689, ('N063A', 'R011', '42 ST-PORT AUTH')), (2370, ('N076', 'R111', '23 ST')), (1878, ('N077', 'R111', '23 ST')), (1863, ('R132', 'R190', '23 ST')), (1661, ('R605', 'R456', 'HOYT ST')), (1612, ('N063', 'R011', '42 ST-PORT AUTH')), (1434, ('R609', 'R056', 'NEVINS ST')), (1390, ('N603', 'R303', '21 ST-QNSBRIDGE')), (1387, ('N062', 'R011', '42 ST-PORT AUTH')), (1134, ('R608', 'R056', 'NEVINS ST')), (1072, ('R245', 'R051', '59 ST')), (964, ('R604', 'R108', 'BOROUGH HALL')), (736, ('N530', 'R301', 'YORK ST')), (694, ('R131', 'R190', '23 ST')), (689, ('R227A', 'R131', '23 ST')), (634, ('R606', 'R225', 'HOYT ST')), (531, ('C021', 'R212', '59 ST')), (441, ('R226A', 'R131', '23 ST')), (415, ('C001', 'R108', 'BOROUGH HALL')), (408, ('R405', 'R447', 'CYPRESS AV')), (372, ('N075', 'R111', '23 ST')), (312, ('R227', 'R131', '23 ST')), (287, ('R404', 'R447', 'CYPRESS AV')), (152, ('C022', 'R212', '59 ST'))] **Evening:** [(3536, ('N062A', 'R010', '42 ST-PORT AUTH')), (2768, ('N060', 'R010', '42 ST-PORT AUTH')), (2025, ('R226', 'R131', '23 ST')), (2017, ('N600', 'R302', '57 ST')), (1669, ('R132', 'R190', '23 ST')), (1088, ('R501', 'R054', '5 AVE')), (872, ('N530', 'R301', 'YORK ST')), (834, ('R131', 'R190', '23 ST')), (728, ('R227A', 'R131', '23 ST')), (606, ('R226A', 'R131', '23 ST')), (326, ('R404', 'R447', 'CYPRESS AV')), (287, ('R227', 'R131', '23 ST')), (280, ('R405', 'R447', 'CYPRESS AV')), (157, ('R245', 'R051', '59 ST'))]

6/13:Tue

[(6131, ('R501', 'R054', '5 AVE')), (6020, ('R245A', 'R051', '59 ST')), (5341, ('N509', 'R203', '23 ST')), (5095, ('R244', 'R050', '59 ST')), (4801, ('N508', 'R453', '23 ST')), (4368, ('N600', 'R302', '57 ST')), (4342, ('R244A', 'R050', '59 ST')), (4048, ('R601A', 'R108', 'BOROUGH HALL')), (3881, ('N062A', 'R010', '42 ST-PORT AUTH')), (3423, ('A002', 'R051', '59 ST')), (3387, ('N060', 'R010', '42 ST-PORT AUTH')), (3257, ('A031', 'R083', '23 ST')), (3101, ('N063A', 'R011', '42 ST-PORT AUTH')), (2988, ('A030', 'R083', '23 ST')), (2905, ('R602', 'R108', 'BOROUGH HALL')), (2535, ('R226', 'R131', '23 ST')), (2450, ('N076', 'R111', '23 ST')), (2140, ('N077', 'R111', '23 ST')), (1837, ('N063', 'R011', '42 ST-PORT AUTH')), (1626, ('R605', 'R456', 'HOYT ST')), (1609, ('R609', 'R056', 'NEVINS ST')), (1589, ('R245', 'R051', '59 ST')), (1544, ('R132', 'R190', '23 ST')), (1494, ('N062', 'R011', '42 ST-PORT AUTH')), (1234, ('N603', 'R303', '21 ST-QNSBRIDGE')), (1128, ('R608', 'R056', 'NEVINS ST')), (1067, ('R604', 'R108', 'BOROUGH HALL')), (785, ('N530', 'R301', 'YORK ST')), (766, ('R227A', 'R131', '23 ST')), (703, ('R131', 'R190', '23 ST')), (589, ('R606', 'R225', 'HOYT ST')), (538, ('C021', 'R212', '59 ST')), (519, ('N075', 'R111', '23 ST')), (489, ('R405', 'R447', 'CYPRESS AV')), (451, ('C001', 'R108', 'BOROUGH HALL')), (443, ('R226A', 'R131', '23 ST')), (344, ('R227', 'R131', '23 ST')), (290, ('R404', 'R447', 'CYPRESS AV')), (134, ('C022', 'R212', '59 ST'))] **Evening**: [(4009, ('N062A', 'R010', '42 ST-PORT AUTH')), (3549, ('N060', 'R010', '42 ST-PORT AUTH')), (2302, ('N600', 'R302', '57 ST')), (1963, ('R226', 'R131', '23 ST')), (1685, ('R132', 'R190', '23 ST')), (1015, ('R131', 'R190', '23 ST')), (949, ('N530', 'R301', 'YORK ST')), (876, ('R227A', 'R131', '23 ST')), (692, ('R226A', 'R131', '23 ST')), (307, ('R405', 'R447', 'CYPRESS AV')), (284, ('R227', 'R131', '23 ST')), (235, ('R404', 'R447', 'CYPRESS AV'))]

6/14: Wed

[(6249, ('R501', 'R054', '5 AVE')), (5901, ('R245A', 'R051', '59 ST')), (5496, ('N509', 'R203', '23 ST')), (5476, ('R244', 'R050', '59 ST')), (5187, ('N508', 'R453', '23 ST')), (4868, ('R244A', 'R050', '59 ST')), (4395, ('N600', 'R302', '57 ST')), (3909, ('R601A', 'R108', 'BOROUGH HALL')), (3823, ('N062A', 'R010', '42 ST-PORT AUTH')), (3572, ('N060', 'R010', '42 ST-PORT AUTH')), (3500, ('A002', 'R051', '59 ST')), (3459, ('A031', 'R083', '23 ST')), (3242, ('A030', 'R083', '23 ST')), (3006, ('R602', 'R108', 'BOROUGH HALL')), (2796, ('N063A', 'R011', '42 ST-PORT AUTH')), (2741, ('N076', 'R111', '23 ST')), (2562, ('R226', 'R131', '23 ST')), (2454, ('N077', 'R111', '23 ST')), (1918, ('N063', 'R011', '42 ST-PORT AUTH')), (1775, ('R245', 'R051', '59 ST')), (1664, ('N062', 'R011', '42 ST-PORT AUTH')), (1617, ('R605', 'R456', 'HOYT ST')), (1575, ('R132', 'R190', '23 ST')), (1447, ('R609', 'R056', 'NEVINS ST')), (1295, ('N603', 'R303', '21 ST-QNSBRIDGE')), (1198, ('R608', 'R056', 'NEVINS ST')), (894, ('R604', 'R108', 'BOROUGH HALL')), (695, ('N530', 'R301', 'YORK ST')), (682, ('R131', 'R190', '23 ST')), (613, ('R606', 'R225', 'HOYT ST')), (545, ('N075', 'R111', '23 ST')), (543, ('R227A', 'R131', '23 ST')), (477, ('R405', 'R447', 'CYPRESS AV')), (456, ('C021', 'R212', '59 ST')), (437, ('R226A', 'R131', '23 ST')), (436, ('C001', 'R108', 'BOROUGH HALL')), (284, ('R227', 'R131', '23 ST')), (260, ('R404', 'R447', 'CYPRESS AV')), (167, ('C022', 'R212', '59 ST'))]

6/15: THU

[(6098, ('R245A', 'R051', '59 ST')), (6041, ('R501', 'R054', '5 AVE')), (5187, ('R244', 'R050', '59 ST')), (5171, ('N508', 'R453', '23 ST')), (5077, ('N509', 'R203', '23 ST')), (4420, ('R244A', 'R050', '59 ST')), (4214, ('N600', 'R302', '57 ST')), (3756, ('R601A', 'R108', 'BOROUGH HALL')), (3693, ('N062A', 'R010', '42 ST-PORT AUTH')), (3369, ('A002', 'R051', '59 ST')), (3362, ('N060', 'R010', '42 ST-PORT AUTH')), (3334, ('A031', 'R083', '23 ST')), (3112, ('R602', 'R108', 'BOROUGH HALL')), (3034, ('A030', 'R083', '23 ST')), (2891, ('N063A', 'R011', '42 ST-PORT AUTH')), (2754, ('R226', 'R131', '23 ST')), (2663, ('N076', 'R111', '23 ST')), (2186, ('N077', 'R111', '23 ST')), (2002, ('R245', 'R051', '59 ST')), (1820, ('R132', 'R190', '23 ST')), (1690, ('N063', 'R011', '42 ST-PORT AUTH')), (1681, ('R605', 'R456', 'HOYT ST')), (1468, ('N062', 'R011', '42 ST-PORT AUTH')), (1457, ('R609', 'R056', 'NEVINS ST')), (1300, ('N603', 'R303', '21 ST-QNSBRIDGE')), (1180, ('R608', 'R056', 'NEVINS ST')), (1042, ('R604', 'R108', 'BOROUGH HALL')), (847, ('N530', 'R301', 'YORK ST')), (782, ('R227A', 'R131', '23 ST')), (690, ('R131', 'R190', '23 ST')), (617, ('R606', 'R225', 'HOYT ST')), (525, ('C021', 'R212', '59 ST')), (513, ('N075', 'R111', '23 ST')), (500, ('R226A', 'R131', '23 ST')), (486, ('R405', 'R447', 'CYPRESS AV')), (445, ('C001', 'R108', 'BOROUGH HALL')), (351, ('R227', 'R131', '23 ST')), (287, ('R404', 'R447', 'CYPRESS AV')), (148, ('C022', 'R212', '59 ST'))]

6/16: FRI

[(5684, ('R245A', 'R051', '59 ST')), (5215, ('R501', 'R054', '5 AVE')), (4802, ('R244', 'R050', '59 ST')), (4778, ('N508', 'R453', '23 ST')), (4471, ('N509', 'R203', '23 ST')), (4095, ('R244A', 'R050', '59 ST')), (3721, ('N600', 'R302', '57 ST')), (3602, ('N062A', 'R010', '42 ST-PORT AUTH')), (3457, ('R601A', 'R108', 'BOROUGH HALL')), (3393, ('N063A', 'R011', '42 ST-PORT AUTH')), (3235, ('A002', 'R051', '59 ST')), (3082, ('N060', 'R010', '42 ST-PORT AUTH')), (2895, ('A030', 'R083', '23 ST')), (2889, ('R602', 'R108', 'BOROUGH HALL')), (2786, ('A031', 'R083', '23 ST')), (2554, ('N076', 'R111', '23 ST')), (2508, ('R226', 'R131', '23 ST')), (2156, ('N077', 'R111', '23 ST')), (1698, ('N063', 'R011', '42 ST-PORT AUTH')), (1686, ('N062', 'R011', '42 ST-PORT AUTH')), (1574, ('R605', 'R456', 'HOYT ST')), (1566, ('R132', 'R190', '23 ST')), (1550, ('R609', 'R056', 'NEVINS ST')), (1451, ('R245', 'R051', '59 ST')), (1340, ('N603', 'R303', '21 ST-QNSBRIDGE')), (1213, ('R608', 'R056', 'NEVINS ST')), (921, ('R604', 'R108', 'BOROUGH HALL')), (770, ('N530', 'R301', 'YORK ST')), (713, ('R227A', 'R131', '23 ST')), (636, ('R606', 'R225', 'HOYT ST')), (599, ('R131', 'R190', '23 ST')), (508, ('N075', 'R111', '23 ST')), (491, ('C021', 'R212', '59 ST')), (451, ('R405', 'R447', 'CYPRESS AV')), (435, ('C001', 'R108', 'BOROUGH HALL')), (414, ('R226A', 'R131', '23 ST')), (258, ('R404', 'R447', 'CYPRESS AV')), (234, ('R227', 'R131', '23 ST')), (155, ('C022', 'R212', '59 ST'))]

Morning Busiest Stops:

Mon:('R245A', 'R051', '59 ST') 8-12

('N509', 'R203', '23 ST') 8-12

('N600', 'R302', '57 ST') 5-9

Tue: ('R501', 'R054', '5 AVE') 8-12

('R245A', 'R051', '59 ST') 8-12

('N509', 'R203', '23 ST') 8-12

Wed: ('R501', 'R054', '5 AVE') 8-12

('R245A', 'R051', '59 ST') 8-12

('N509', 'R203', '23 ST') 8-12

Thu: ('R245A', 'R051', '59 ST') 8a - 12p

('R501', 'R054', '5 AVE') 8a - 12p

('N508', 'R453', '23 ST') 8a -12p

Fri: ('R245A', 'R051', '59 ST') 8a - 12p

('R501', 'R054', '5 AVE') 8a - 12p

('N508', 'R453', '23 ST') 8a - 12p

Sat: ('N063A', 'R011', '42 ST-PORT AUTH') 8-12

('R244', 'R050', '59 ST') 8a - 12p

('R601A', 'R108', 'BOROUGH HALL') 8-12

Sun: ('N063A', 'R011', '42 ST-PORT AUTH') 8a - 12p

('R244', 'R050', '59 ST') 8a - 12p

('R501', 'R054', '5 AVE') 8a - 12p

Evening Busiest Stops:

Mon:('N062A', 'R010', '42 ST-PORT AUTH') 2p - 6p

('R226', 'R131', '23 ST') 1p - 5p

Tue:('N062A', 'R010', '42 ST-PORT AUTH') 2p - 6p

('N600', 'R302', '57 ST') 1p - 5p

Wed:('N062A', 'R010', '42 ST-PORT AUTH') 2p - 6p

('N600', 'R302', '57 ST') 1p - 5p

Thu:('N062A', 'R010', '42 ST-PORT AUTH') 2p - 6p

('R226', 'R131', '23 ST') 1p - 5p

Fri:('N062A', 'R010', '42 ST-PORT AUTH') 2p - 6p

('N600', 'R302', '57 ST') 1p - 5p

Sat: ('N062A', 'R010', '42 ST-PORT AUTH') 2pm-6pm

('N600', 'R302', '57 ST') 1pm-5pm

Sun: ('N062A', 'R010', '42 ST-PORT AUTH') 2pm-6pm

('N600', 'R302', '57 ST') 1pm-5pm

14Wed

[(3918, ('N062A', 'R010', '42 ST-PORT AUTH')), (3173, ('N060', 'R010', '42 ST-PORT AUTH')), (2284, ('N600', 'R302', '57 ST')), (1751, ('R226', 'R131', '23 ST')), (1664, ('R132', 'R190', '23 ST')), (1025, ('N530', 'R301', 'YORK ST')), (886, ('R131', 'R190', '23 ST')), (834, ('R227A', 'R131', '23 ST')), (622, ('R226A', 'R131', '23 ST')), (489, ('R227', 'R131', '23 ST')), (270, ('R405', 'R447', 'CYPRESS AV')), (196, ('R404', 'R447', 'CYPRESS AV'))]

15Thu

[(4252, ('N062A', 'R010', '42 ST-PORT AUTH')), (3543, ('N060', 'R010', '42 ST-PORT AUTH')), (2313, ('R226', 'R131', '23 ST')), (2168, ('N600', 'R302', '57 ST')), (1776, ('R132', 'R190', '23 ST')), (1109, ('N530', 'R301', 'YORK ST')), (1036, ('R131', 'R190', '23 ST')), (797, ('R227A', 'R131', '23 ST')), (757, ('R226A', 'R131', '23 ST')), (297, ('R405', 'R447', 'CYPRESS AV')), (265, ('R404', 'R447', 'CYPRESS AV')), (150, ('R227', 'R131', '23 ST'))]

16Fri

[(4140, ('N062A', 'R010', '42 ST-PORT AUTH')), (3473, ('N060', 'R010', '42 ST-PORT AUTH')), (2190, ('N600', 'R302', '57 ST')), (1988, ('R132', 'R190', '23 ST')), (1890, ('R226', 'R131', '23 ST')), (1027, ('R131', 'R190', '23 ST')), (953, ('N530', 'R301', 'YORK ST')), (776, ('R227A', 'R131', '23 ST')), (692, ('R226A', 'R131', '23 ST')), (292, ('R404', 'R447', 'CYPRESS AV')), (287, ('R405', 'R447', 'CYPRESS AV')), (122, ('R227', 'R131', '23 ST'))]

Busiest stop evening: