Taller 7

Métodos Computacionales para Políticas Públicas - URosario

Entrega: viernes 3-abr-2020 11:59 PM

[Andrés Ramírez Vela]

[andrese.ramirez@urosario.edu.co]

Instrucciones:

- Guarde una copia de este *Jupyter Notebook* en su computador, idealmente en una carpeta destinada al material del curso.
- Modifique el nombre del archivo del *notebook*, agregando al final un guión inferior y su nombre y apellido, separados estos últimos por otro guión inferior. Por ejemplo, mi *notebook* se llamaría: mcpp_taller7_santiago_matallana
- Marque el *notebook* con su nombre y e-mail en el bloque verde arriba. Reemplace el texto "[Su nombre acá]" con su nombre y apellido. Similar para su e-mail.
- Desarrolle la totalidad del taller sobre este *notebook*, insertando las celdas que sea necesario debajo de cada pregunta. Haga buen uso de las celdas para código y de las celdas tipo *markdown* según el caso.
- Recuerde salvar periódicamente sus avances.
- Cuando termine el taller:
    1. Descárguelo en PDF. Si tiene algún problema con la conversión, descárguelo en HTML.
    2. Suba todos los archivos a su repositorio en GitHub, en una carpeta destinada exclusivamente para este taller, antes de la fecha y hora límites.

(Todos los ejercicios tienen el mismo valor.)

Este taller tiene dos partes. Una obligatoria, relativamente fácil, y otra voluntaria y más retadora. Los invito a intentar desarrollar el taller en su totalidad. (Buen plan para el aislamiento obligatorio.)

En este taller exploraremos los datos de crimen de Chicago.

Descargue los datos de crimen del Chicago Data Portal solo para el año 2015 (https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present/ijzp-q8t2/data (https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present/ijzp-q8t2/data)).

Parte obligatoria

1.

Calcule el número de crímenes en cada Community Area en 2015. Haga un gráfico de barras que lo ilustre.

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
plt.rcParams["figure.figsize"] = [38.0, 20.0]
plt.style.use('ggplot')
```

In [2]:
```
crimes = pd.read_csv('Crimes_-_2001_to_present.csv', parse_dates=['Date'
])
```

In [3]:
```
type(crimes)
```

Out[3]:
```
pandas.core.frame.DataFrame
```

In [4]:
```
crimes.head()
```

Out[4]:

	ID	Case Number	Date	Block	IUCR	Primary Type	Description	Location Description	A
0	12021303	JD198763	2015-07-01 09:35:00	055XX S RUTHERFORD AVE	1153	DECEPTIVE PRACTICE	FINANCIAL IDENTITY THEFT OVER $ 300	NaN	
1	11768614	JC361321	2015-01-01 00:00:00	030XX W 41ST ST	1751	OFFENSE INVOLVING CHILDREN	CRIMINAL SEXUAL ABUSE BY FAMILY MEMBER	APARTMENT	
2	11752913	JC342515	2015-01-01 00:00:00	030XX W 41ST ST	1752	OFFENSE INVOLVING CHILDREN	AGGRAVATED CRIMINAL SEXUAL ABUSE BY FAMILY MEMBER	APARTMENT	
3	12014151	JD191104	2015-01-20 13:50:00	057XX W LAKE ST	1153	DECEPTIVE PRACTICE	FINANCIAL IDENTITY THEFT OVER $ 300	NaN	
4	10509684	HZ251005	2015-12-25 08:00:00	0000X N WALLER AVE	1752	OFFENSE INVOLVING CHILDREN	AGGRAVATED CRIMINAL SEXUAL ABUSE BY FAMILY MEMBER	RESIDENCE	

5 rows × 22 columns

In [5]:
```
crimes['Description'].to_frame()
```

Out[5]:

	Description
0	FINANCIAL IDENTITY THEFT OVER $ 300
1	CRIMINAL SEXUAL ABUSE BY FAMILY MEMBER
2	AGGRAVATED CRIMINAL SEXUAL ABUSE BY FAMILY MEMBER
3	FINANCIAL IDENTITY THEFT OVER $ 300
4	AGGRAVATED CRIMINAL SEXUAL ABUSE BY FAMILY MEMBER
...	...
264344	AGGRAVATED: HANDGUN
264345	ARMED: HANDGUN
264346	DOMESTIC BATTERY SIMPLE
264347	SIMPLE
264348	SIMPLE

264349 rows × 1 columns

In [6]:
```
crimes_by_community = crimes.groupby("Community Area")
```

In [7]:
```
crimes_by_community.groups
```

Out[7]:
```
{0: Int64Index([158331, 232738], dtype='int64'),
 1: Int64Index([     8,    105,    300,    345,    767,    806,    838,
900,
                935,    957,
               ...
               264046, 264056, 264080, 264170, 264225, 264226, 264257, 26
4284,
               264300, 264319],
              dtype='int64', length=3592),
 2: Int64Index([    77,    104,    154,    194,    199,    306,    357,
442,
                508,    662,
               ...
               263317, 263447, 263465, 263608, 263904, 263952, 264168, 26
4169,
               264297, 264321],
              dtype='int64', length=3113),
 3: Int64Index([    54,     90,    391,    423,    448,    462,    651,
663,
                690,    715,
               ...
               263918, 263919, 263926, 264067, 264078, 264127, 264156, 26
4158,
               264213, 264302],
              dtype='int64', length=3665),
 4: Int64Index([   492,    672,    722,    842,   1750,   1776,   1831,
1924,
                 2361,   2678,
               ...
               263024, 263585, 263731, 263836, 263853, 263983, 264042, 26
4215,
               264228, 264286],
              dtype='int64', length=1770),
 5: Int64Index([   483,   1235,   1268,   1464,   1665,   2193,   2763,
2768,
                 3605,   4212,
               ...
               262671, 262795, 263161, 263196, 263355, 263454, 263566, 26
3883,
               263975, 264239],
              dtype='int64', length=1395),
 6: Int64Index([   109,    180,    284,    408,    464,    538,    574,
635,
                697,    713,
               ...
               263753, 263850, 263920, 263927, 264044, 264157, 264162, 26
4163,
               264164, 264203],
              dtype='int64', length=5591),
 7: Int64Index([   553,    564,    630,    636,    653,    655,    666,
673,
                695,    698,
               ...
               263812, 263894, 263901, 263909, 263922, 263925, 264033, 26
4159,
               264184, 264320],
              dtype='int64', length=3764),
```

```
 8: Int64Index([    66,    138,    214,    298,    355,    395,    425,
435,
               479,    491,
             ...
             264108, 264124, 264153, 264161, 264189, 264243, 264263, 26
4277,
             264280, 264281],
            dtype='int64', length=9079),
 9: Int64Index([   336,    825,   1941,   2253,   2271,   4652,   6783,
7393,
              7913,   8107,
             ...
             258520, 259140, 259375, 259556, 259675, 261122, 263630, 26
4039,
             264064, 264066],
            dtype='int64', length=258),
 10: Int64Index([   167,    480,    497,    541,    657,   1314,   202
6,   2201,
              2547,   2730,
             ...
             262233, 262235, 262237, 262736, 263175, 263452, 263646, 26
3656,
             263876, 264198],
            dtype='int64', length=1278),
 11: Int64Index([  1426,   1725,   1832,   2114,   2631,   2632,   291
0,   3095,
              4244,   4706,
             ...
             260928, 261068, 261118, 261260, 262844, 263635, 263636, 26
3671,
             264252, 264329],
            dtype='int64', length=1157),
 12: Int64Index([   509,   1215,   2297,   4385,   4699,   4731,   556
5,   5843,
              5954,   6034,
             ...
             256885, 256968, 259030, 260506, 260635, 260660, 262199, 26
3235,
             263631, 264231],
            dtype='int64', length=450),
 13: Int64Index([   169,    247,    324,    726,   1096,   2097,   217
2,   2233,
              2504,   3662,
             ...
             261803, 261949, 262193, 262574, 262583, 262768, 263010, 26
3020,
             264079, 264208],
            dtype='int64', length=844),
 14: Int64Index([    17,    111,    173,    294,    319,    413,    41
8,    562,
               595,   1421,
             ...
             263763, 263884, 263986, 264062, 264183, 264192, 264212, 26
4254,
             264259, 264325],
            dtype='int64', length=2537),
 15: Int64Index([   181,    282,    293,    328,    447,    580,    59
```

```
0,    894,
               981,    998,
            ...
            263628, 263629, 263754, 264016, 264053, 264063, 264094, 26
4118,
            264330, 264348],
           dtype='int64', length=3454),
 16: Int64Index([    81,     87,    155,    309,    354,    370,    42
4,    433,
               444,    481,
            ...
            263405, 263411, 263806, 263868, 263980, 264029, 264152, 26
4175,
            264190, 264191],
           dtype='int64', length=3022),
 17: Int64Index([   230,    320,   1049,   1307,   1674,   1851,   188
9,   1962,
             1979,   2170,
            ...
            261623, 261794, 261982, 262157, 262236, 263031, 263664, 26
4037,
            264178, 264332],
           dtype='int64', length=1660),
 18: Int64Index([    60,    687,    906,   1264,   1635,   2158,   263
3,   2852,
             2872,   2883,
            ...
            260214, 260474, 260478, 260507, 261573, 261806, 261995, 26
2122,
            262651, 264173],
           dtype='int64', length=587),
 19: Int64Index([    11,     14,     62,     70,    125,    151,    27
1,    273,
               312,    358,
            ...
            263729, 263733, 263846, 263981, 264086, 264105, 264174, 26
4188,
            264247, 264342],
           dtype='int64', length=4843),
 20: Int64Index([     5,     28,    141,    255,    458,    598,    61
6,    777,
               829,    952,
            ...
            262927, 262970, 262971, 263122, 263299, 263330, 263612, 26
3722,
            263775, 263963],
           dtype='int64', length=1734),
 21: Int64Index([   366,    514,   1087,   1118,   1277,   1560,   159
1,   1604,
             1721,   1734,
            ...
            262928, 263285, 263350, 263625, 263633, 263674, 263742, 26
3988,
            264240, 264347],
           dtype='int64', length=2261),
 22: Int64Index([    37,     46,    156,    204,    226,    269,    54
2,    589,
```

```
              644,    700,
             ...
             263688, 263769, 263770, 263772, 263915, 264032, 264071, 26
4205,
             264219, 264241],
            dtype='int64', length=4807),
 23: Int64Index([   163,    182,    192,    212,    239,    258,    29
5,    311,
                313,    321,
             ...
             264237, 264245, 264248, 264272, 264282, 264291, 264314, 26
4317,
             264333, 264346],
            dtype='int64', length=8202),
 24: Int64Index([    15,    153,    274,    292,    340,    365,    39
2,    488,
                494,    515,
             ...
             263790, 263804, 263805, 263821, 263857, 263940, 264031, 26
4102,
             264275, 264285],
            dtype='int64', length=7087),
 25: Int64Index([     3,      4,     29,     30,     47,     61,     9
3,    110,
                120,    148,
             ...
             264238, 264253, 264255, 264256, 264258, 264260, 264267, 26
4274,
             264289, 264338],
            dtype='int64', length=17410),
 26: Int64Index([    16,     33,    128,    159,    161,    172,    19
0,    289,
                367,    401,
             ...
             263803, 263814, 263899, 263950, 264007, 264121, 264138, 26
4236,
             264326, 264339],
            dtype='int64', length=6038),
 27: Int64Index([     7,     23,     59,    108,    130,    142,    17
9,    183,
                223,    307,
             ...
             263924, 263934, 264091, 264112, 264122, 264273, 264298, 26
4304,
             264323, 264343],
            dtype='int64', length=5459),
 28: Int64Index([    71,    121,    184,    220,    256,    288,    29
7,    349,
                377,    383,
             ...
             264018, 264022, 264060, 264072, 264107, 264110, 264128, 26
4181,
             264210, 264278],
            dtype='int64', length=7955),
 29: Int64Index([    25,     31,     67,     98,    160,    164,    17
4,    176,
                227,    234,
```

```
            ...
            264077, 264100, 264125, 264167, 264171, 264197, 264202, 26
4251,
            264293, 264341],
           dtype='int64', length=8429),
 30: Int64Index([    85,    114,    201,    360,    471,    528,    57
3,   1095,
              1104,   1181,
            ...
            263622, 263642, 263679, 263796, 263823, 263905, 263908, 26
4196,
            264306, 264318],
           dtype='int64', length=4455),
 31: Int64Index([   131,    207,    217,    229,    308,    482,    53
1,    575,
               683,    934,
            ...
            263602, 263708, 263756, 263882, 264028, 264214, 264234, 26
4270,
            264308, 264337],
           dtype='int64', length=2530),
 32: Int64Index([    26,     94,    301,    484,    522,    554,    57
6,    617,
               652,    679,
            ...
            264119, 264142, 264148, 264150, 264151, 264199, 264222, 26
4233,
            264283, 264312],
           dtype='int64', length=7596),
 33: Int64Index([    88,    335,    466,    899,   1084,   1210,   121
6,   1578,
              2047,   2162,
            ...
            262666, 263082, 263253, 263286, 263579, 263676, 263779, 26
3784,
            263864, 264221],
           dtype='int64', length=1883),
 34: Int64Index([   100,    677,   1025,   1473,   1511,   1730,   215
3,   2183,
              2227,   2257,
            ...
            262623, 262696, 263250, 263362, 263433, 264024, 264160, 26
4250,
            264264, 264296],
           dtype='int64', length=1085),
 35: Int64Index([    69,    210,    337,    411,    654,    828,    97
7,   1053,
              1091,   1099,
            ...
            262849, 263080, 263182, 263288, 263463, 263638, 263660, 26
3951,
            263985, 264120],
           dtype='int64', length=2525),
 36: Int64Index([    32,     97,    708,   1070,   1209,   1334,   192
8,   2286,
              2386,   2534,
            ...
```

```
            261737, 261939, 262992, 263040, 263394, 263518, 263860, 26
3953,
            264185, 264303],
           dtype='int64', length=635),
 37: Int64Index([   107,   1047,   1218,   1867,   2557,   2683,   292
8,   3350,
              4135,   4138,
            ...
            262031, 262402, 262807, 262932, 263439, 263730, 263798, 26
3838,
            263933, 264195],
           dtype='int64', length=843),
 38: Int64Index([   189,    196,    213,    216,    218,    267,    27
7,    286,
               373,    403,
            ...
            263002, 263015, 263016, 263060, 263212, 263213, 263381, 26
3424,
            264186, 264232],
           dtype='int64', length=3158),
 39: Int64Index([    38,     73,     84,    224,    272,    478,    58
6,    621,
               699,    958,
            ...
            262622, 262809, 262810, 263191, 263230, 263232, 263451, 26
3621,
            264043, 264345],
           dtype='int64', length=1377),
 40: Int64Index([    40,    113,    126,    166,    221,    253,    28
1,    440,
               551,    572,
            ...
            263604, 263619, 263653, 263698, 263855, 263862, 264104, 26
4113,
            264307, 264344],
           dtype='int64', length=2774),
 41: Int64Index([    41,     99,    988,   1624,   1779,   2083,   210
4,   2185,
              2717,   2725,
            ...
            262247, 262264, 262358, 262418, 262897, 263371, 263434, 26
3761,
            263941, 263955],
           dtype='int64', length=1518),
 42: Int64Index([    18,    149,    170,    205,    410,    414,    45
7,    527,
               594,    737,
            ...
            263739, 263746, 263800, 263827, 263856, 263998, 264041, 26
4154,
            264180, 264261],
           dtype='int64', length=3757),
 43: Int64Index([    43,     78,     86,    103,    117,    140,    15
8,    175,
               211,    241,
            ...
            264004, 264040, 264050, 264201, 264294, 264295, 264313, 26
```

```
4316,
            264322, 264336],
           dtype='int64', length=9111),
 44: Int64Index([    24,    115,    136,    233,    249,    334,    52
9,    549,
               568,    739,
            ...
            263791, 263916, 263923, 263999, 264057, 264074, 264109, 26
4200,
            264204, 264242],
           dtype='int64', length=6050),
 45: Int64Index([    49,    495,    552,    749,    927,   1298,   133
6,   1466,
             1495,   2385,
            ...
            263189, 263229, 263251, 263481, 263552, 263957, 263976, 26
3994,
            264130, 264134],
           dtype='int64', length=1391),
 46: Int64Index([    64,     65,    139,    265,    387,    445,    44
6,    540,
               623,    646,
            ...
            263611, 263657, 263681, 263869, 263917, 264014, 264047, 26
4049,
            264141, 264220],
           dtype='int64', length=5093),
 47: Int64Index([   504,    510,   1476,   1514,   2688,   2749,   285
7,   4169,
             5250,   6550,
            ...
            253738, 254315, 256859, 257015, 259031, 259099, 259838, 26
1768,
            263795, 264013],
           dtype='int64', length=389),
 48: Int64Index([    22,     50,     74,    200,    536,    893,   100
2,   1092,
             1171,   2062,
            ...
            261754, 262022, 262257, 262285, 262656, 263388, 263421, 26
3591,
            264216, 264292],
           dtype='int64', length=1487),
 49: Int64Index([     6,     76,     96,    102,    195,    259,    30
5,    361,
               470,    487,
            ...
            263802, 263810, 263818, 263837, 263912, 264005, 264089, 26
4093,
            264244, 264335],
           dtype='int64', length=6649),
 50: Int64Index([    80,    416,    604,    641,   1760,   2259,   229
9,   2459,
             2560,   4306,
            ...
            262626, 262638, 262901, 262936, 263606, 263809, 263992, 26
4084,
```

```
            264165, 264310],
           dtype='int64', length=1049),
 51: Int64Index([    12,     19,     95,    638,   1137,   1569,   239
7,   2804,
              2971,   3524,
             ...
             262861, 262919, 263397, 263484, 263578, 263582, 263979, 26
4011,
             264166, 264288],
           dtype='int64', length=1927),
 52: Int64Index([   106,    122,    132,    322,   1026,   1543,   229
4,   4395,
              4955,   5000,
             ...
             261964, 262073, 262103, 262313, 262316, 262426, 262778, 26
3086,
             263372, 264133],
           dtype='int64', length=1319),
 53: Int64Index([    39,    101,    123,    157,    278,    290,    34
6,    680,
               723,    863,
             ...
             263609, 263641, 263654, 263973, 263997, 264081, 264082, 26
4085,
             264090, 264129],
           dtype='int64', length=4018),
 54: Int64Index([   624,    634,   1322,   1854,   2457,   2556,   261
5,   3047,
              4362,   4413,
             ...
             262750, 263811, 263840, 263854, 263974, 263989, 264023, 26
4207,
             264217, 264218],
           dtype='int64', length=1189),
 55: Int64Index([  1206,   2134,   2307,   2390,   4353,   4387,   447
2,   4994,
              5662,   5915,
             ...
             259331, 259453, 259454, 259776, 260604, 260734, 262197, 26
3159,
             263293, 264132],
           dtype='int64', length=516),
 56: Int64Index([     0,    116,    168,    270,    304,    326,    36
8,    443,
               493,    854,
             ...
             263320, 263391, 263392, 263670, 263757, 263781, 263794, 26
4000,
             264027, 264223],
           dtype='int64', length=1857),
 57: Int64Index([   250,    359,    612,   1105,   2133,   2322,   243
1,   2559,
              2645,   2892,
             ...
             263225, 263256, 263393, 263492, 263519, 263568, 263570, 26
3658,
             263740, 263741],
```

```
           dtype='int64', length=998),
 58: Int64Index([     1,      2,    119,    310,    344,    452,    51
3,    519,
                557,    588,
             ...
             263491, 263569, 263610, 263615, 263652, 263808, 263932, 26
4145,
             264177, 264301],
            dtype='int64', length=2760),
 59: Int64Index([    35,    257,    583,   1128,   1286,   1701,   191
8,   2231,
               2242,   2336,
             ...
             260822, 261028, 261319, 262045, 262706, 262855, 263361, 26
3537,
             263945, 264331],
            dtype='int64', length=1083),
 60: Int64Index([   566,    643,    656,    805,   1147,   1350,   164
7,   1691,
               1810,   1864,
             ...
             263009, 263119, 263271, 263389, 263655, 263830, 263958, 26
4305,
             264328, 264334],
            dtype='int64', length=1615),
 61: Int64Index([    27,    124,    188,    222,    236,    268,    28
0,    353,
                363,    369,
             ...
             263936, 264001, 264038, 264051, 264070, 264144, 264147, 26
4265,
             264324, 264327],
            dtype='int64', length=5004),
 62: Int64Index([   135,   1741,   1975,   2024,   5881,   6064,   629
5,   6975,
               7146,   7831,
             ...
             260800, 260983, 261681, 261828, 261833, 262447, 262603, 26
3448,
             263797, 263972],
            dtype='int64', length=1037),
 63: Int64Index([    83,    112,    165,    219,    356,    378,    42
7,    436,
                459,    615,
             ...
             262643, 262884, 262887, 262902, 263074, 263192, 263359, 26
3911,
             264017, 264249],
            dtype='int64', length=2367),
 64: Int64Index([    63,    225,    260,    477,    559,    585,   235
4,   2535,
               2640,   3255,
             ...
             261607, 261692, 261749, 261788, 262505, 262728, 262798, 26
2965,
             263170, 263841],
            dtype='int64', length=942),
```

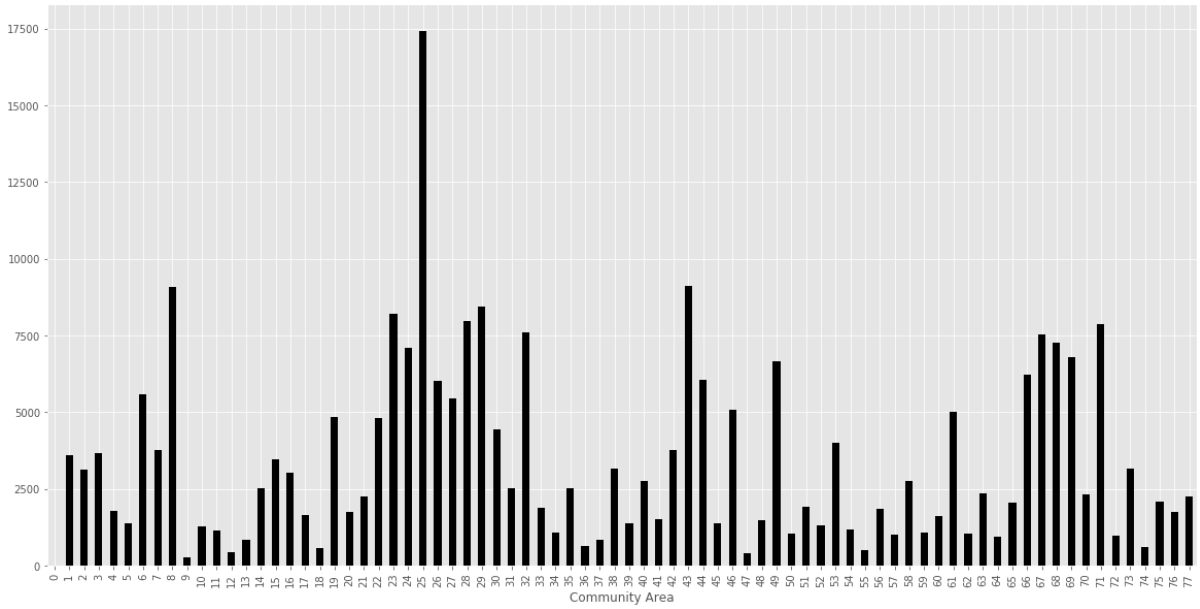
```
 65: Int64Index([    36,     52,    137,    251,    261,    262,    45
6,    745,
                945,   1196,
             ...
             262617, 262664, 262824, 262958, 263488, 263526, 263565, 26
3613,
             263768, 263777],
            dtype='int64', length=2056),
 66: Int64Index([    13,     42,     44,     48,     51,    118,    13
4,    231,
                237,    248,
             ...
             263863, 263865, 263881, 263937, 263971, 263991, 264068, 26
4211,
             264287, 264340],
            dtype='int64', length=6225),
 67: Int64Index([    72,     75,    127,    146,    152,    171,    20
6,    299,
                393,    501,
             ...
             263987, 263990, 264003, 264019, 264076, 264092, 264101, 26
4116,
             264224, 264276],
            dtype='int64', length=7547),
 68: Int64Index([    10,     34,     58,     79,    145,    177,    19
7,    238,
                242,    245,
             ...
             263935, 263938, 263943, 263946, 263970, 264140, 264146, 26
4206,
             264227, 264315],
            dtype='int64', length=7255),
 69: Int64Index([    20,     82,    129,    147,    191,    198,    21
5,    263,
                264,    296,
             ...
             264052, 264061, 264083, 264087, 264117, 264136, 264137, 26
4262,
             264309, 264311],
            dtype='int64', length=6783),
 70: Int64Index([    45,    162,    348,    502,    877,    880,   127
3,   2059,
               2122,   2195,
             ...
             263323, 263333, 263383, 263461, 263528, 263564, 263617, 26
3678,
             263842, 264179],
            dtype='int64', length=2319),
 71: Int64Index([    53,     55,     57,     92,    133,    143,    14
4,    186,
                209,    243,
             ...
             263874, 263880, 263886, 263888, 263984, 264065, 264149, 26
4209,
             264290, 264299],
            dtype='int64', length=7855),
 72: Int64Index([   276,    318,    631,   1818,   2109,   2164,   286
```

```
1,   4558,
              4607,   4668,
             ...
             261953, 261986, 262508, 263313, 263472, 263473, 263493, 26
3839,
             263965, 264269],
            dtype='int64', length=991),
 73: Int64Index([    68,     89,     91,    202,    287,    303,    36
4,    385,
              428,    473,
             ...
             263474, 263725, 263726, 263820, 263847, 263849, 264123, 26
4131,
             264268, 264271],
            dtype='int64', length=3170),
 74: Int64Index([   591,    661,    738,   1619,   1922,   2274,   253
1,   4304,
              4313,   4403,
             ...
             258895, 258897, 259826, 260886, 261034, 261592, 262286, 26
2660,
             262779, 262898],
            dtype='int64', length=619),
 75: Int64Index([    21,     56,    275,    291,    376,    648,    67
4,    685,
              742,    931,
             ...
             263307, 263348, 263643, 263727, 263848, 263851, 263966, 26
4155,
             264230, 264279],
            dtype='int64', length=2088),
 76: Int64Index([   240,    362,    379,    398,    399,    468,    62
8,    658,
              676,    678,
             ...
             263893, 263907, 263913, 263929, 263930, 263939, 264045, 26
4048,
             264055, 264229],
            dtype='int64', length=1749),
 77: Int64Index([     9,    232,    323,    431,    512,    819,    91
7,    974,
              979,   1057,
             ...
             263314, 263404, 263469, 263547, 263549, 263614, 263645, 26
4139,
             264246, 264266],
            dtype='int64', length=2262)}
```

In [18]:
```
community_crime_count = crimes_by_community['Community Area'].agg('count')
community_crime_count
```

Out[18]:
```
Community Area
0         2
1      3592
2      3113
3      3665
4      1770
       ...
73     3170
74      619
75     2088
76     1749
77     2262
Name: Community Area, Length: 78, dtype: int64
```

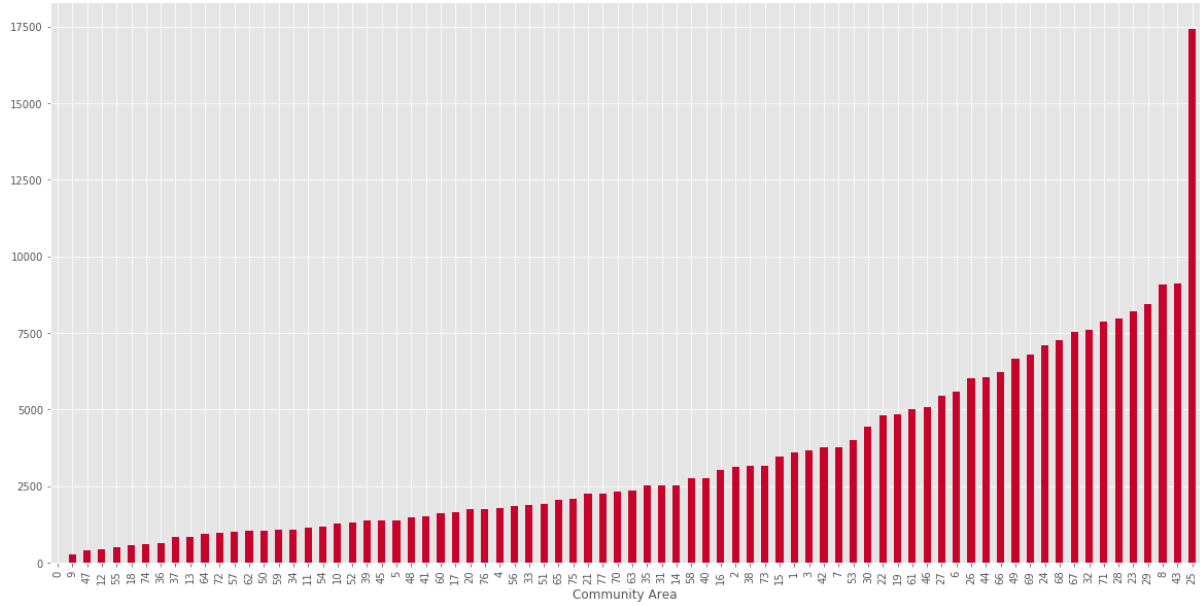
In [24]:
```
community_crime_count.plot(kind='bar', color='Black', figsize=(20,10));
```



2.

Ordene las Community Areas de acuerdo con el número de crímenes. ¿Qué Community Area (por nombre, idealmente) presenta el mayor número de crímenes? ¿El menor?

In [25]:
```
organize = crimes_by_community['Community Area'].agg('count')
organize.sort_values()
x =organize.sort_values()
x.plot(kind='bar', color="#C8032B", figsize=(20,10));
```



La Community Area que presenta el menor número de crimenes es Edison Park ubicado al norte de la ciudad y representado en la gráfica con el número 9. Por el contrario, la Community Area con mayor número de crimenes es Austin ubicado al occidente de la ciudad y representado por el número 25 en la gráfica.

3.

Cree una tabla cuyas filas sean días del año (yyyy-mm-dd) y las columnas las 77 Community Areas. En cada campo de la tabla deberá haber el correspondiente número de crímenes. Seleccione algunas Community Areas que le llamen la atención y haga un gráfico de serie de tiempo.

Pista: El siguiente código puede serle útil.

In [11]:
```
# Create function to strip time from date field, and use it to create an
other column
def to_day(timestamp):
    return timestamp.replace(minute=0,hour=0, second=0)

crimes['Day'] = crimes['Date'].apply(to_day)
```

In [12]:
```
crimes_by_community_day = crimes.groupby(["Community Area","Day"])
```

In [13]:
```
community_crime_count1 = crimes_by_community_day['ID'].agg('count')
community_crime_count1.to_frame()
```

Out[13]:

		ID
Community Area	**Day**	
0	2015-01-14	1
	2015-10-08	1
	2015-01-01	18
1	2015-01-02	5
	2015-01-03	7
...	...	...
	2015-12-27	2
	2015-12-28	9
77	2015-12-29	6
	2015-12-30	4
	2015-12-31	4

26915 rows × 1 columns

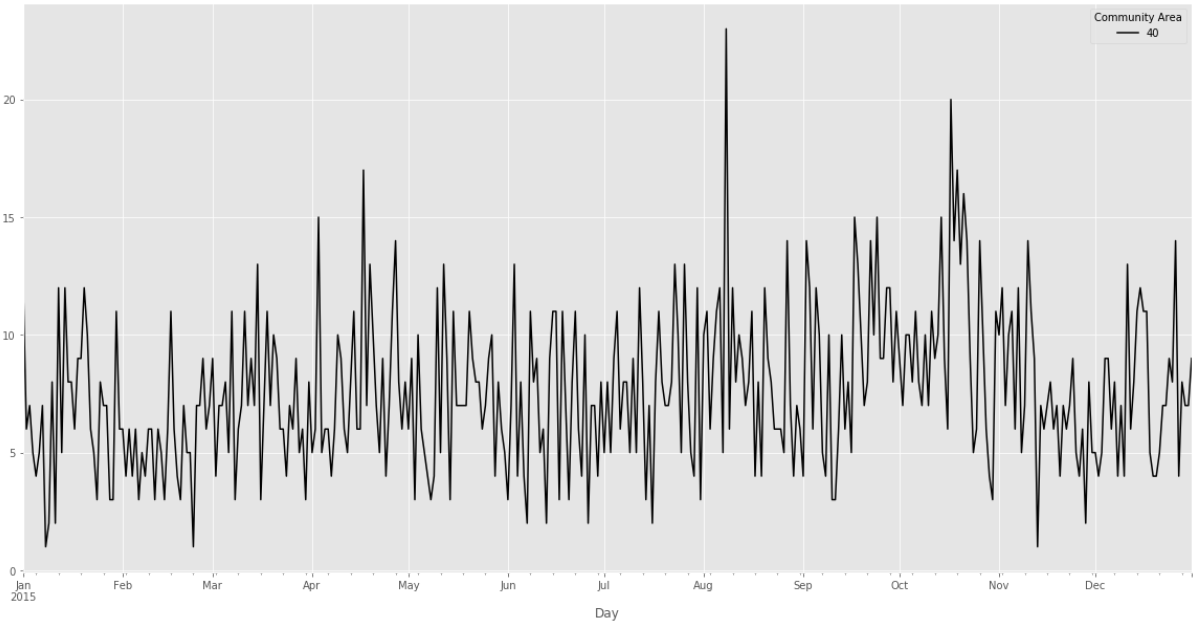
In [14]:
```
community_crime_count2 =community_crime_count1.unstack("Community Area")
community_crime_count2
```

Out[14]:

Community Area	0	1	2	3	4	5	6	7	8	9	...	68	69	70	71	
Day																
2015-01-01	NaN	18.0	13.0	15.0	6.0	5.0	25.0	15.0	48.0	1.0	...	31.0	31.0	9.0	53.0	3
2015-01-02	NaN	5.0	9.0	8.0	3.0	2.0	10.0	9.0	27.0	NaN	...	12.0	22.0	6.0	17.0	1
2015-01-03	NaN	7.0	11.0	9.0	7.0	4.0	6.0	11.0	27.0	1.0	...	23.0	12.0	8.0	18.0	Na
2015-01-04	NaN	12.0	7.0	9.0	10.0	3.0	15.0	5.0	16.0	1.0	...	13.0	15.0	9.0	12.0	1
2015-01-05	NaN	6.0	7.0	5.0	4.0	5.0	15.0	7.0	11.0	1.0	...	16.0	12.0	8.0	17.0	Na
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
2015-12-27	NaN	13.0	8.0	6.0	4.0	1.0	16.0	10.0	35.0	NaN	...	13.0	19.0	4.0	26.0	2
2015-12-28	NaN	7.0	8.0	6.0	2.0	2.0	14.0	9.0	19.0	NaN	...	12.0	23.0	9.0	14.0	2
2015-12-29	NaN	6.0	8.0	14.0	8.0	4.0	10.0	5.0	24.0	NaN	...	19.0	16.0	7.0	19.0	Na
2015-12-30	NaN	5.0	9.0	8.0	4.0	1.0	12.0	17.0	28.0	1.0	...	11.0	23.0	6.0	14.0	2
2015-12-31	NaN	8.0	4.0	9.0	4.0	3.0	19.0	5.0	27.0	NaN	...	19.0	17.0	4.0	19.0	1

365 rows × 78 columns

In [26]:
```
community_crime_count2[[40]].plot(color= "Black" , figsize=(20,10));
```



Parte voluntaria

Descargue la base de datos de información socioeconómica (https://data.cityofchicago.org/Health-Human-Services/Census-Data-Selected-socioeconomic-indicators-in-C/kn9c-c2s2 (https://data.cityofchicago.org/Health-Human-Services/Census-Data-Selected-socioeconomic-indicators-in-C/kn9c-c2s2)).

4.

Cree una tabla que agregue el número de crímenes por Community Area. Una esa tabla con la de datos socioeconómicos y cree un "scatter plot" de número de crímenes vs ingreso per cápita. Explique la relación en palabras.