```
Timer unit: 1e-06 s

Total time: 1863.43 s
File: <ipython-input-24-f2c30edaec5d>
Function: get_id_data at line 1

Line #      Hits         Time  Per Hit   % Time  Line Contents
==============================================================
     1                                           def get_id_data():
     2                                               # URL to search venues example with offset
     3         1          2.0      2.0      0.0      url = 'https://api.foursquare.com/v2/venues/search'
     4         1       4719.0   4719.0      0.0      last_run_values = pd.read_csv('files/last_run_values.csv')
     5                                               # Creating a temp list from the categories starting from the last used category
     6         1        142.0    142.0      0.0      new_value = last_run_values['values'].values[-1]
     7         1       3473.0   3473.0      0.0      position = category['category'][category['category'] == new_value].index.to_list()
     8         1        190.0    190.0      0.0      category_temp_list = category[(position[-1]):]
     9                                               # Create the lat_long generator
    10         1         16.0     16.0      0.0      ll_iter = ll_generator(lat_long['lat_long'])
    11                                               # value to be saved in the last run values file
    12         1          1.0      1.0      0.0      category_id = None
    13         1          1.0      1.0      0.0      end_of_search = False
    14         1       1091.0   1091.0      0.0      with open('files/id_data_test.csv', 'a') as csv_file:
    15                                                 # creating a csv writer object
    16         1          6.0      6.0      0.0        csvwriter = csv.writer(csv_file)
    17         1          0.0      0.0      0.0        headers = 5000
    18                                                 # Going over the 5000 requests limit per hour
    19         2        178.0     89.0      0.0        while headers > 1:
    20        12        280.0     23.3      0.0          for i in category_temp_list['category']:
    21        12         15.0      1.2      0.0            category_id = i
    22        12        138.0     11.5      0.0            logging.info(f"Runs category ID: {category_id}")
    23        12         14.0      1.2      0.0            if headers < 1:
    24         1          2.0      2.0      0.0              break
    25        11         10.0      0.9      0.0            if end_of_search:
    26                                                       break
    27      5010       7384.0      1.5      0.0            for a in range(1, len(lat_long)):
    28      5000       5163.0      1.0      0.0              try:
    29      5000      23961.0      4.8      0.0                ll = next(ll_iter)
    30        10         17.0      1.7      0.0              except StopIteration:
    31                                                         # Reset the lat_long generator
    32        10        238.0     23.8      0.0                ll_iter = ll_generator(lat_long['lat_long'])
    33        10        112.0     11.2      0.0                logging.info('The lat_long generator was reset')
    34        10        272.0     27.2      0.0                ll = next(ll_iter)
    35                                           
    36      5000       8318.0      1.7      0.0              params['ll'] = ll
    37      5000       5698.0      1.1      0.0              params['categoryId'] = category_id
    38      5000       5534.0      1.1      0.0              params['limit'] = 50
    39                                           
    40      5000 1858558512.0 371711.7     99.7              resp = requests.get(url=url, params=params)
    41                                           
    42      5000      14569.0      2.9      0.0              try:
    43      5000     300363.0     60.1      0.0                headers = int(dict(resp.headers)['X-RateLimit-Remaining'])
    44                                                       except:
    45                                                         logging.warning('The headers did not return a X-RateLimit-Remaining value')
    46                                           
    47      5000       8402.0      1.7      0.0              if headers < 1:
    48         1          1.0      1.0      0.0                break
    49      4999    1793886.0    358.8      0.1              if category_id == category['category'].tail(1).item() and ll == lat_long['lat_lor
    50                                                         end_of_search = True
    51                                                         break
    52                                           
    53      4999    1609750.0    322.0      0.1              data = json.loads(resp.text)
    54      4999      12283.0      2.5      0.0              if data['meta']['code'] != 200:
    55                                                         logging.warning(f"Error code in response: {data['meta']['code']}")
    56                                           
    57      4999       5643.0      1.1      0.0              try:
    58     92658     114983.0      1.2      0.0                for venue in data['response']['venues']:
    59                                                           # writing the data rows to the csv file
    60    175318     443038.0      2.5      0.0                  csvwriter.writerows(
    61     87659      92342.0      1.1      0.0                    [
    62     87659     411210.0      4.7      0.0                      [venue['id'], str(datetime.now())]
    63                                                             ]
    64                                                           )
    65                                                       except:
    66                                                         logging.critical("Did not write into the id_data.csv file")
```