G. Michael Fitzgerald (gmf5de)
CS 6501 Smart and Healthy Buildings - Lab 3

Deliverables:

1. A write up of the plans to execute the task and how it was divided.

   I worked alone on this lab assignment because I had already made significant progress towards the deliverable when we were asked to split into groups. Oops :|
   For this function I implemented it in the following stages:

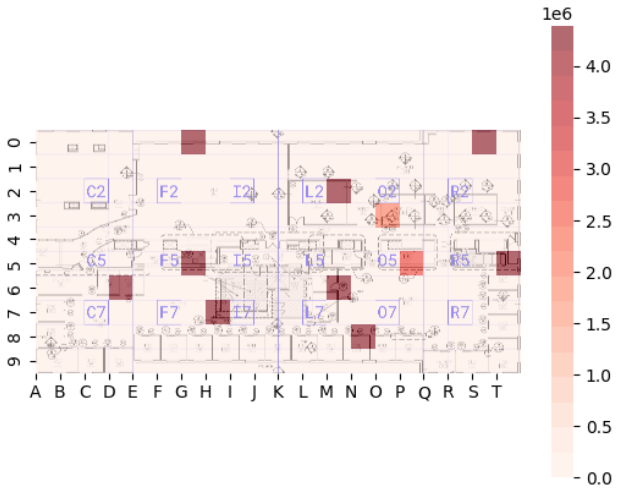
   i. Start with listing existing sensors.
   ii. For each grid square, list the sensors contained in that square.
   iii. For each sensor in a given grid square, find the intersection between the fields it has and the fields parameter passed into the function.
   iv. For each relevant field, get the data from the given sensor in the time range specified in the function parameters. Add the number of datapoints to a running total on a per-grid-square basis.
   v. After this process has completed for each grid square, create the heatmap using the number of data points per grid square.

2. A function with signatures shown above that, if called, generates a heatmap of the Link Lab that shows how much data is detected relative to each grid.

```
def generate_linklab_heatmap(start_datetime, end_datetime, fields, export_path):
    fields_set = set(fields)

    # get sensor information
    print('Reading in sensor registration information...')
    df = pd.read_csv('book_with_grids.csv')

    # grid range 0-199
    datapoints = [
        [0] * 20,
        [0] * 20,
        [0] * 20,
        [0] * 20,
        [0] * 20,
        [0] * 20,
        [0] * 20,
        [0] * 20,
        [0] * 20,
        [0] * 20
    ]

    print('Iterating across the grid to gather data points...')
    for grid in range(200):
        print(f'[GRID {grid}]')
        sensors_in_grid = df[df['grid'] == grid]

        # for each sensor in the grid
        dps = 0
        for row_id, sensor in sensors_in_grid.iterrows():
            print(f'\tsensor {row_id}')
            sensor_fields = sensor['fields'].split(',')
            device_id_list = [sensor['device_id']]
            # print(sensor_fields)
            # print(fields)
            relevant_fields = list(set(sensor_fields).intersection(fields_set))
            # print(relevant_fields)
            for field in relevant_fields:
```

```
                print(f'\t\t{field}')
                with suppress_stdout_stderr():
                    ldf = util.get_ldf(field, start_datetime, end_datetime,
device_id_list)
                if ldf is not None:
                    dps += ldf.shape[0]
                else:
                    print('\t\t\tNo data in time range.')
                # dps += len(ldf) ??
            datapoints[int(grid / 20)][grid % 20] = dps
        print(datapoints)
        dpi_scale_trans = plt.figure().dpi_scale_trans
        plt.clf()

        print('Plotting data on the heatmap...')
        with suppress_stdout_stderr():
            img = cv2.imread('grid.png')
            colors = sns.color_palette("light:#F00", 24, as_cmap=True)
            heatmap = sns.heatmap(
                data=datapoints,
                square=True,
                cmap=colors,
                zorder=2,
                alpha=0.6
            )
            plt.imshow(
                img,
                aspect=heatmap.get_aspect(),
                extent=heatmap.get_xlim() + heatmap.get_ylim(),
                zorder=1,
            )
            plt.xticks(range(20), list("ABCDEFGHIJKLMNOPQRST"))

        print('Exporting heatmap to image...')
        plt.savefig(export_path)
        plt.clf()
        print('Done!')
        return export_path
```

3. An annual aggregated heatmap from 2021/1/1 to 2021/09/21, where the plan of the Link Lab is plotted behind the heatmap.



4. Create a public repository that can show this work.
   https://github.com/tic/smart_and_healthy_buildings/tree/master/lab_3

Extra credit: Generate a video of the Link Lab heatmap across its history and upload it as a mp4.

I have a small disclaimer for this extra credit. Running the program to generate the heatmap was pretty time consuming, so I made two executive decisions that I hope are reasonable in your mind. First, the date range is essentially the same as from deliverable #3: 2021/1/1-2021/9/23, **not** a date range which covers most of the Link Lab's existence. Second, each image shown in the video shows data points from a single day in the lab, and the heatmap isn't aggregated over time. Again, this was because running the annual aggregation took a long time, and I didn't want to tie up my laptop all weekend to generate the heatmaps for this. Finally, not all sensor fields are included. I chose the fields which appeared to be the most popular among sensors. Similarly to the previous items, I opted to do this to save time. Disclaimers aside, here's a link to the video:
https://github.com/tic/smart_and_healthy_buildings/blob/master/lab_3/visualization.mp4