

Creating final data for crop and ferti recommendation system

```
In [ ]: #pandas is used for analyzing and cleaning the data
#matplotlib and seaborn are used for visualizing the data
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
In [ ]: #Reading the updated fertilizer dataset
fertilizer_data_path = '../Data-raw/FertilizerData.csv'
merge_fert = pd.read_csv(fertilizer_data_path)
```

```
In [ ]: merge_fert.head()
```

```
Out[ ]:   Unnamed: 0  Crop  N  P  K  pH
0          0   rice  80  40  40  5.5
1          3  maize  80  40  20  5.5
2          5 chickpea  40  60  80  5.5
3         12 kidneybeans  20  60  20  5.5
4         13 pigeonpeas  20  60  20  5.5
```

```
In [ ]: del merge_fert['Unnamed: 0']
```

```
In [ ]: #Describing the fertilizer data
#count-total no. of rows, mean ,std,min,max for Nitrogen,Phosphorou, Pottasium, pH
merge_fert.describe()
```

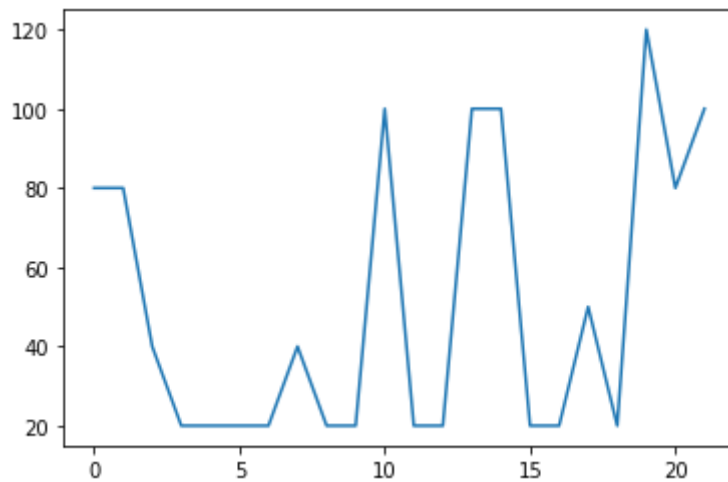
```
Out[ ]:      N      P      K      pH
count  22.000000  22.000000  22.000000  22.000000
mean    50.454545  45.681818  48.181818   5.409091
std     36.315715  32.634172  51.698426   0.590326
min     20.000000  10.000000  10.000000   4.000000
25%     20.000000  20.000000  20.000000   5.500000
50%     30.000000  40.000000  30.000000   5.500000
75%     80.000000  60.000000  50.000000   5.500000
max    120.000000 125.000000 200.000000   6.500000
```

```
In [ ]: #There are 22 unique values in crop dataset
merge_fert['Crop'].unique()
```

```
Out[ ]: array(['rice', 'maize', 'chickpea', 'kidneybeans', 'pigeonpeas',
        'mothbeans', 'mungbean', 'blackgram', 'lentil', 'pomegranate',
        'banana', 'mango', 'grapes', 'watermelon', 'muskmelon', 'apple',
        'orange', 'papaya', 'coconut', 'cotton', 'jute', 'coffee'],
        dtype=object)
```

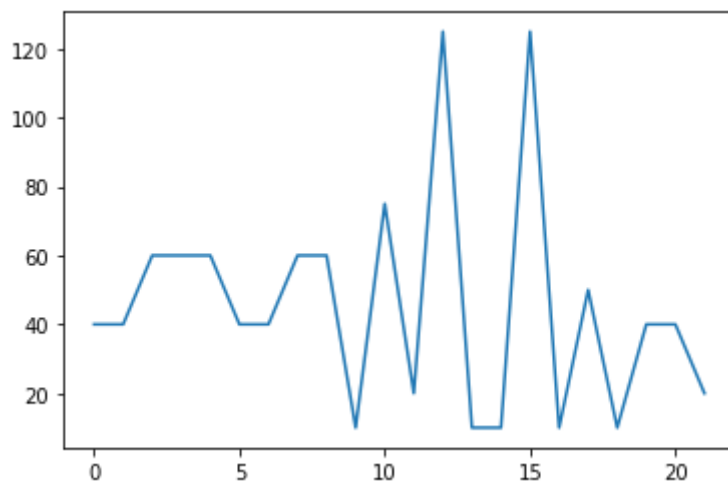
```
In [ ]: #plotted the graph for nitrogen data-we can see min is 20 and max is 120  
plt.plot(merge_fert["N"])
```

```
Out[ ]: [matplotlib.lines.Line2D at 0x25ad149efb0>]
```



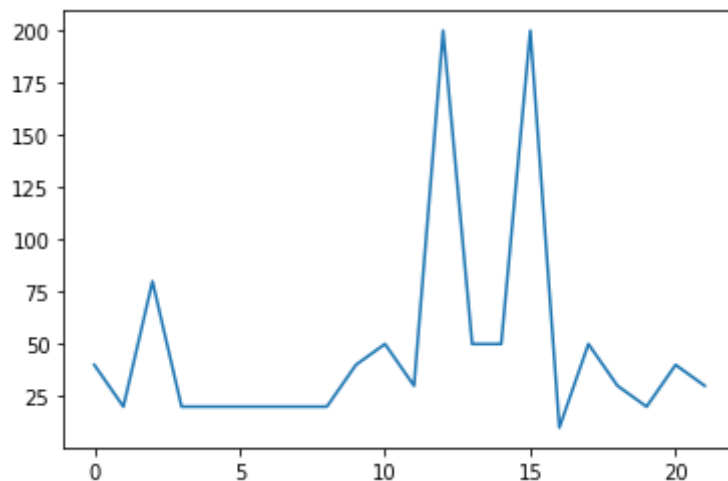
```
In [ ]: #plotted the graph for Phosphorous data-we can see min is 10 and max is 125  
plt.plot(merge_fert["P"])
```

```
Out[ ]: [matplotlib.lines.Line2D at 0x25ad25b9120>]
```



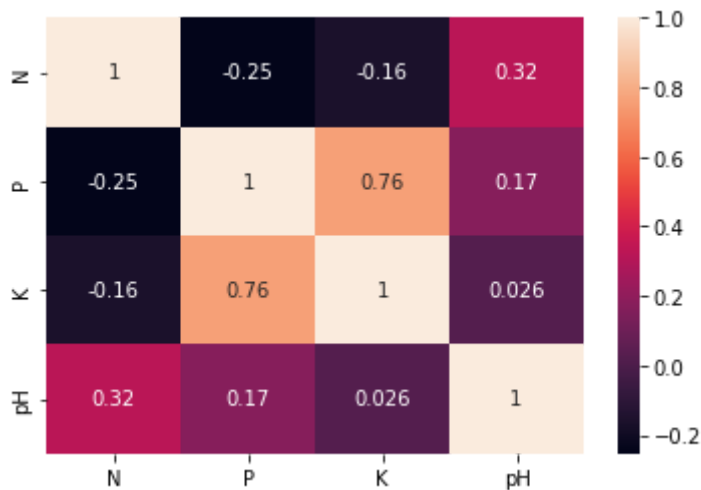
```
In [ ]: ##plotted the graph for pottasium data-we can see min is 10 and max is 200  
plt.plot(merge_fert["K"])
```

```
Out[ ]: [matplotlib.lines.Line2D at 0x25ad263b010>]
```



```
In [ ]: #we can see positive and negative correlation between each feature
sns.heatmap(merge_fert.corr(),annot=True)
```

```
Out[ ]: <AxesSubplot:>
```



```
In [ ]: #reading the updated crop file
merge_crop = pd.read_csv('../Data-raw/MergeFileCrop.csv')
reco_fert = merge_fert
```

```
In [ ]: #Stored the column names in temp variable
#We have assigned the fertilizer NPKpH values to temp by adding some random values
import random
temp = pd.DataFrame(columns = ['N','P','K'])
for i in range(0,merge_crop.shape[0]):
    crop = merge_crop.label.iloc[i]
    # print(crop)
    N = reco_fert[reco_fert['Crop'] == crop]["N"].iloc[0] + random.randint(-20,20)
    P = reco_fert[reco_fert['Crop'] == crop]["P"].iloc[0] + random.randint(-5,20)
    K = reco_fert[reco_fert['Crop'] == crop]["K"].iloc[0] + random.randint(-5,5)
    d = {"N":N,"P":P,"K":K}
    #print(d)
    temp = temp.append(d,ignore_index = True)
```

```
In [ ]: temp
```

Out[]:

| | N | P | K |
|------|-----|-----|-----|
| 0 | 77 | 49 | 35 |
| 1 | 60 | 56 | 41 |
| 2 | 62 | 51 | 35 |
| 3 | 88 | 47 | 38 |
| 4 | 65 | 45 | 38 |
| ... | ... | ... | ... |
| 2195 | 95 | 18 | 35 |
| 2196 | 80 | 40 | 34 |
| 2197 | 88 | 22 | 31 |
| 2198 | 93 | 20 | 28 |
| 2199 | 100 | 32 | 29 |

2200 rows × 3 columns

In []:

```
#N,P,K columns are merged to crop dataset
merge_crop['N'] = temp['N']
merge_crop['P'] = temp['P']
merge_crop['K'] = temp['K']
```

In []:

```
merge_crop
```

Out[]:

| | Unnamed: 0 | temperature | humidity | ph | rainfall | label | N | P | K |
|------|------------|-------------|-----------|----------|------------|--------|-----|-----|-----|
| 0 | 0 | 20.879744 | 82.002744 | 6.502985 | 202.935536 | rice | 77 | 49 | 35 |
| 1 | 1 | 21.770462 | 80.319644 | 7.038096 | 226.655537 | rice | 60 | 56 | 41 |
| 2 | 2 | 23.004459 | 82.320763 | 7.840207 | 263.964248 | rice | 62 | 51 | 35 |
| 3 | 3 | 26.491096 | 80.158363 | 6.980401 | 242.864034 | rice | 88 | 47 | 38 |
| 4 | 4 | 20.130175 | 81.604873 | 7.628473 | 262.717340 | rice | 65 | 45 | 38 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 2195 | 895 | 26.774637 | 66.413269 | 6.780064 | 177.774507 | coffee | 95 | 18 | 35 |
| 2196 | 896 | 27.417112 | 56.636362 | 6.086922 | 127.924610 | coffee | 80 | 40 | 34 |
| 2197 | 897 | 24.131797 | 67.225123 | 6.362608 | 173.322839 | coffee | 88 | 22 | 31 |
| 2198 | 898 | 26.272418 | 52.127394 | 6.758793 | 127.175293 | coffee | 93 | 20 | 28 |
| 2199 | 899 | 23.603016 | 60.396475 | 6.779833 | 140.937041 | coffee | 100 | 32 | 29 |

2200 rows × 9 columns

In []:

```
del merge_crop['Unnamed: 0']
```

In []:

```
#The columns are increased 5 to 8 IN THIS CROP DATASEET
merge_crop
```

Out []:

| | temperature | humidity | ph | rainfall | label | N | P | K |
|------|-------------|-----------|----------|------------|--------|-----|-----|-----|
| 0 | 20.879744 | 82.002744 | 6.502985 | 202.935536 | rice | 77 | 49 | 35 |
| 1 | 21.770462 | 80.319644 | 7.038096 | 226.655537 | rice | 60 | 56 | 41 |
| 2 | 23.004459 | 82.320763 | 7.840207 | 263.964248 | rice | 62 | 51 | 35 |
| 3 | 26.491096 | 80.158363 | 6.980401 | 242.864034 | rice | 88 | 47 | 38 |
| 4 | 20.130175 | 81.604873 | 7.628473 | 262.717340 | rice | 65 | 45 | 38 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 2195 | 26.774637 | 66.413269 | 6.780064 | 177.774507 | coffee | 95 | 18 | 35 |
| 2196 | 27.417112 | 56.636362 | 6.086922 | 127.924610 | coffee | 80 | 40 | 34 |
| 2197 | 24.131797 | 67.225123 | 6.362608 | 173.322839 | coffee | 88 | 22 | 31 |
| 2198 | 26.272418 | 52.127394 | 6.758793 | 127.175293 | coffee | 93 | 20 | 28 |
| 2199 | 23.603016 | 60.396475 | 6.779833 | 140.937041 | coffee | 100 | 32 | 29 |

2200 rows × 8 columns

In []:

```
#The column are kept in ordered
merge_crop = merge_crop[[ 'N', 'P', 'K','temperature', 'humidity', 'ph', 'rainfall
```

In []:

```
#The final mergecrop is saved as csv file
merge_crop.to_csv("../Data-processed/crop_recommendation.csv",index=False)
```

In []:

```
df = pd.read_csv("../Data-processed/crop_recommendation.csv")
```

In []:

```
df.head()
```

Out []:

| | N | P | K | temperature | humidity | ph | rainfall | label |
|---|----|----|----|-------------|-----------|----------|------------|-------|
| 0 | 77 | 49 | 35 | 20.879744 | 82.002744 | 6.502985 | 202.935536 | rice |
| 1 | 60 | 56 | 41 | 21.770462 | 80.319644 | 7.038096 | 226.655537 | rice |
| 2 | 62 | 51 | 35 | 23.004459 | 82.320763 | 7.840207 | 263.964248 | rice |
| 3 | 88 | 47 | 38 | 26.491096 | 80.158363 | 6.980401 | 242.864034 | rice |
| 4 | 65 | 45 | 38 | 20.130175 | 81.604873 | 7.628473 | 262.717340 | rice |

In []:

```
df.shape
```

Out []:

(2200, 8)