IDS PROJECT REPORT

FOOD GRAINS PRODUCTION IN 10 DISTRICTS OF HIMACHAL PRADESH

By

Uggirala Rudhvi(20UCS213)

Pre-Requisites:

Data Science, Information Visualization, Python, and its libraries.

Libraries Used:

Pandas, Numpy, Seaborn, matplot.lib **Abstract:**

The objective of this project is to conduct data pre-processing and data visualization using the concepts taught during the course. This project helps us in understanding the various types of data we deal with in handling various datasets. We also try to get inferences from the data by using statistical analysis taught to us during the course.

We would like to thank Dr. Shakti Balan sir, Dr. Subrat Dash and Dr. Aloke Dutta sir for their valuable input and guidance they have provided during the entire process of completing this project.

Data set Source:

We have collected data from https://data.gov.in/ . The title of the data set was Production under different crops during 2019-20

Preliminary Data analysis:

The given dataset has 13 rows and 11 columns (this is the number after eliminating an entire column as it indicates the name of the state where all the districts lie in, which is Himachal Pradesh). The rows indicate the name of the district, and the columns indicate the name of the Crop. An entire row and an entire column give us the total values respectively**.**

Features Description:

(i) District Name: (Nominal Data)

- Bilaspur
- Chamba
- Hamirpur
- Kangra
- Kinnaur
- Kullu
- Lahaulspiti
- Mandi
- Shimla
- Sirmaur
- Solan
- Una

(ii) Crop name: (Nominal Data)

- Wheat
- Maize
- Rice
- Barley
- Ragi
- Pulses
- Common Millets
- Chilies
- Ginger
- Common Millets

(iii) The crop production values are given in Metric tonnes. The numerical data in the dataset are quantitative attributes (numerical).

Importing the data set:

In [20]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline

data_set = pd.read_csv(r"C:\Users\Rudhvi\OneDrive\Desktop\ids\Prodution-Under-Different-Crops_during-2019-20.csv")
data_set
```

Renaming the Columns for easing further analysis:

2
```
rename = {
    "Wheat ( in Metric Tonnes)":'Wheat',
    "Maize ( in Metric Tonnes)": "Maize",
    "Rice ( in Metric Tonnes)": "Rice",
    "Barley ( in Metric Tonnes)":"Barley",
    "Ragi ( in Metric Tonnes)": "Ragi",
    "Pulses ( in Metric Tonnes)": "Pulses",
    "Common millets ( in Metric Tonnes)": "Common millets",
    "Chillies ( in Metric Tonnes)": "Chillies",
    "Ginger ( in Metric Tonnes)": "Ginger",
    "Oil  seeds ( in Metric Tonnes)": "Oil  seeds",
    "Total Food grains ( in Metric Tonnes)":"Total Food grains"
}
data = data_set.rename(columns=(rename))
data
```

Checking for Null Values:

In [29]:
```
print(data.isnull().sum())
```

```
Wheat             0
Maize             0
Rice              0
Barley            0
Ragi              0
Pulses            0
Common millets    0
Chillies          0
Ginger            0
Oil  seeds        0
dtype: int64
```

The data has nil null values.

hence no further elimination of data points is required.

```
In [2]: data_set.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 13 entries, 0 to 12
Data columns (total 13 columns):
 #   Column                                Non-Null Count  Dtype
---  ------                                --------------  -----
 0   State                                 13 non-null     object
 1   District                              13 non-null     object
 2   Wheat ( in Metric Tonnes)             13 non-null     int64
 3   Maize ( in Metric Tonnes)             13 non-null     int64
 4   Rice ( in Metric Tonnes)              13 non-null     int64
 5   Barley ( in Metric Tonnes)            13 non-null     int64
 6   Ragi ( in Metric Tonnes)              13 non-null     int64
 7   Pulses ( in Metric Tonnes)            13 non-null     int64
 8   Common millets ( in Metric Tonnes)    13 non-null     int64
 9   Total Food grains ( in Metric Tonnes) 13 non-null     int64
 10  Chillies ( in Metric Tonnes)          13 non-null     int64
 11  Ginger ( in Metric Tonnes)            13 non-null     int64
 12  Oil  seeds ( in Metric Tonnes)        13 non-null     int64
dtypes: int64(11), object(2)
memory usage: 1.4+ KB
```

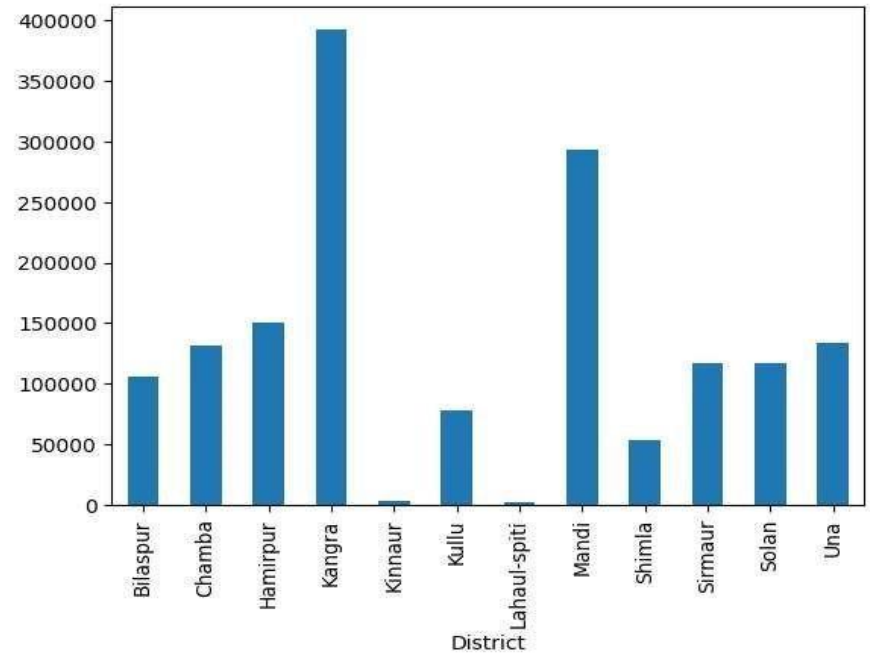
The above-mentioned output mentions the number of non-null values**.**

Bar plot depicting the total production of various crops in all the districts:

Since the numerical data in the dataset is numerical and categorical, we have opted for Bar graphs to represent the data in majority of cases.

```
In [24]: data[:12]["Total Food grains"].plot.bar()
```

```
Out[24]: <AxesSubplot: xlabel='District'>
```



The above bar plot gives us the total production of all the food crops combined. We can infer from the visualization that Kangra and Mandi districts were the highest producers while Kinnaur and Lahaul-spiti were the lowest producers.

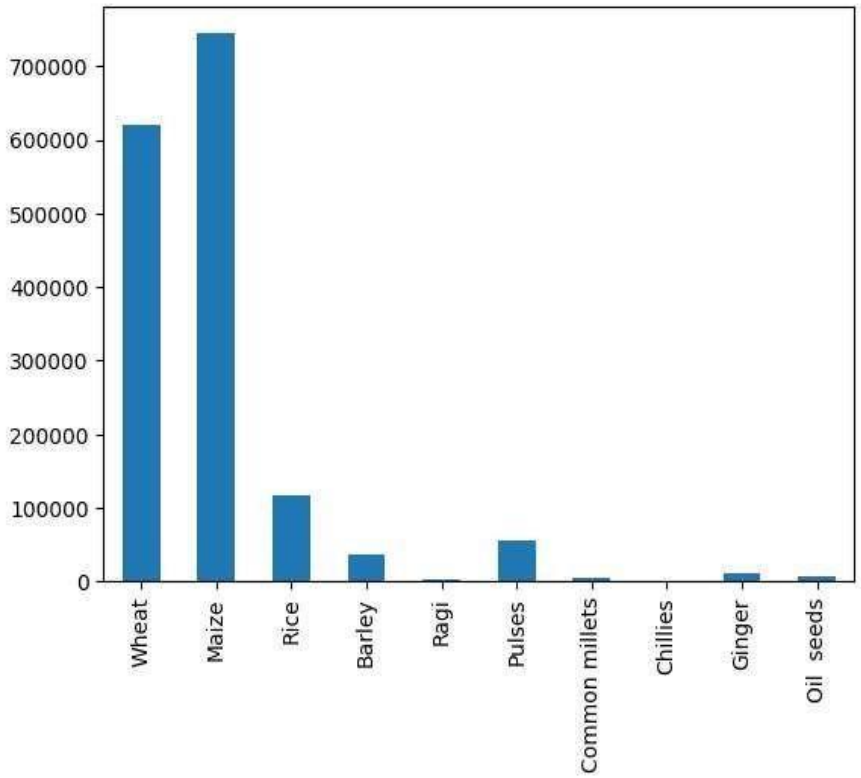
Mean of Each District and Total Districts Crop Production:

```
In [93]: mean_district_production = (data[:12]["Total Food grains"]/10)
         round(mean_district_production, 2)

Out[93]: District
         Bilaspur        10595.9
         Chamba          13180.2
         Hamirpur        15003.5
         Kangra          39191.6
         Kinnaur           335.3
         Kullu            7838.4
         Lahaul-spiti      213.4
         Mandi           29269.6
         Shimla           5360.1
         Sirmaur         11706.2
         Solan           11725.6
         Una             13433.1
         Name: Total Food grains, dtype: float64

In [95]: mean_hp = data[:12]["Total Food grains"].mean()
         round(mean_hp, 2)

Out[95]: 131544.08
```

Total production of each crop in all the districts:

```
7    _ df["Total"].plot.bar()
```

```
7    <AxesSubplot: >
```



Among all the districts, Wheat and Maize were the most farmed crops. Their production values are marginally higher in comparison to the rest of the crops.

Mean of Each Crop in all the Districts:

```
In [96]: mean_crop_production = (d_tdf["Total"]/12)
         round(mean_crop_production, 2)
```

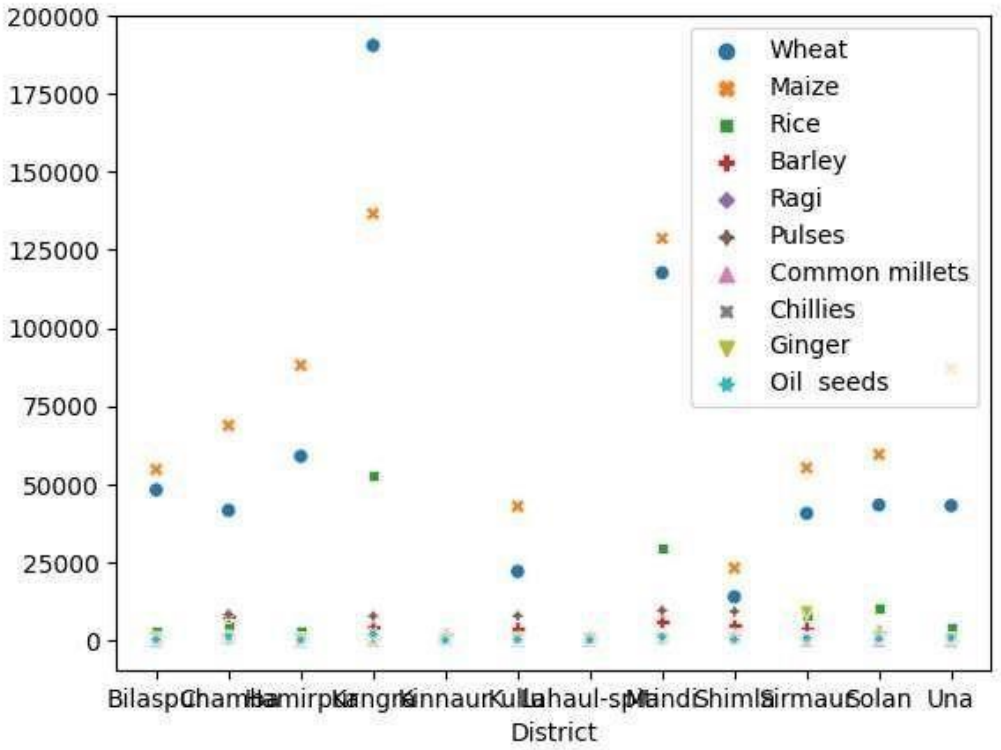
```
Out[96]: Wheat             51640.75
         Maize             62050.25
         Rice               9739.92
         Barley             2951.08
         Ragi                171.67
         Pulses             4602.33
         Common millets      388.08
         Chillies             23.00
         Ginger              889.58
         Oil  seeds          537.00
         Name: Total, dtype: float64
```

Scatter plot depicting production of different crops in each district:

The scatter plot below, gives us an overall understanding of the entire dataset. It provides us the relationship between the crop type, its production in each district. An alternative scatter plot has also been depicted below the Scatter plot.

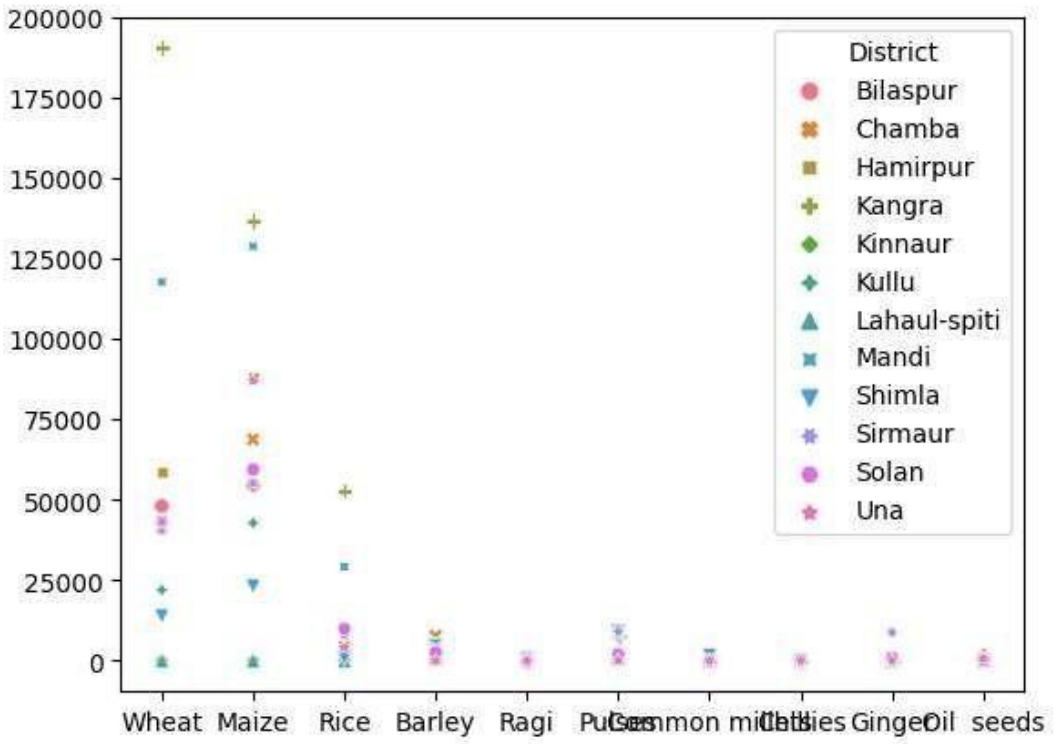
```
[30]: sns.scatterplot(data)
```

```
Out[30]: <AxesSubplot: xlabel='District'>
```



```
1   sns.scatterplot(tdf)
```

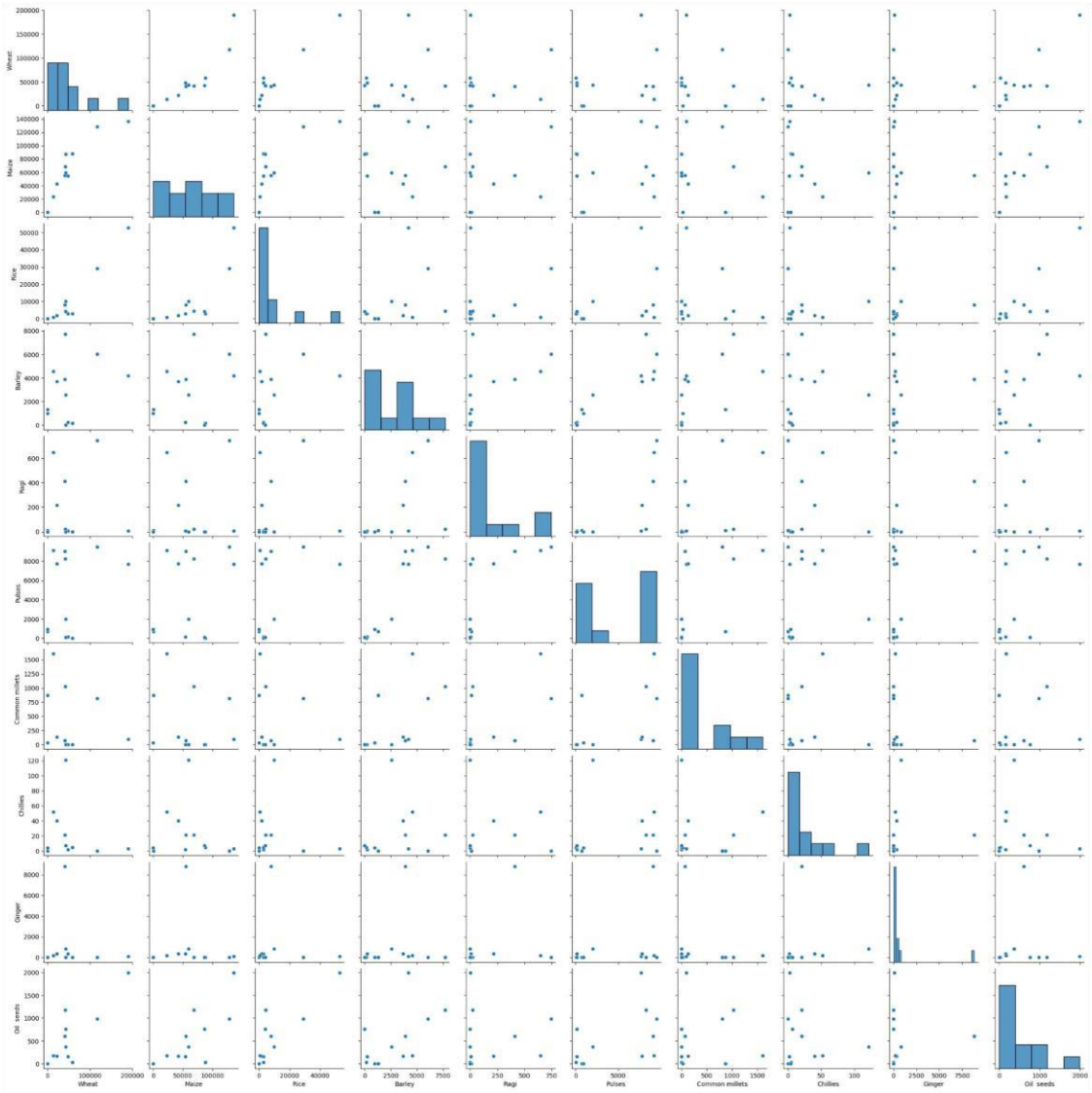
```
Out[61  <\ esSubpl›t: >
```



Detailed plots of relationship between production of different crops:

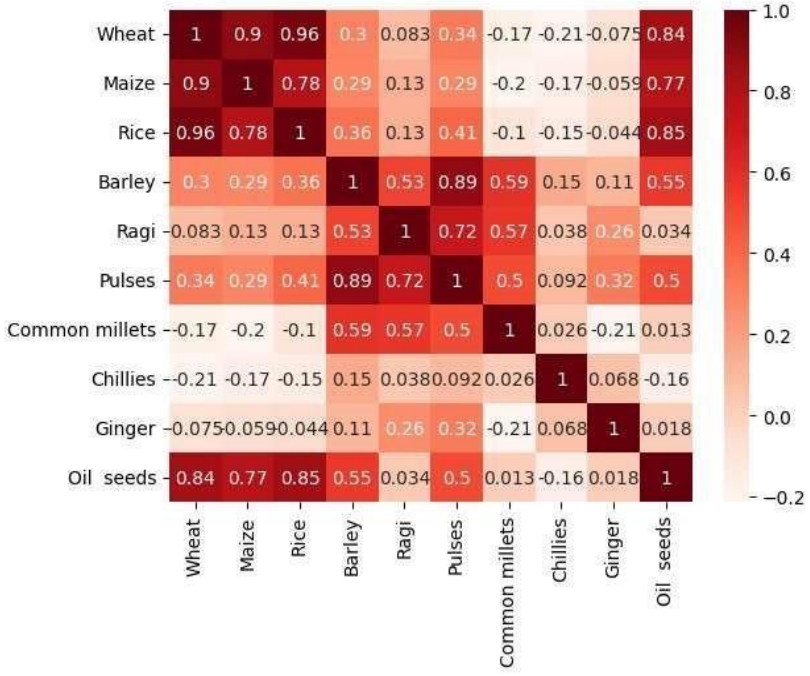
A pair plot is a data representation where pairwise representation can be made easier. The below given pair plot gives a collection of scatter plots and bar plots. The Scatter plots gives the relationship between the production rates of different crops, while the bar plots give us the total production values of the common crop.

```
1  arr = ['Wheat', 'Maize', "Rice", "Barley", "Ragi", "Pulses", "Common millets",
   "Chillies", "Ginger", "Oil  seeds"]
   sns.pairplot(data = data, vars=arr)
   plt.show()
```



A heat map for the same data representation above: (Production rates)

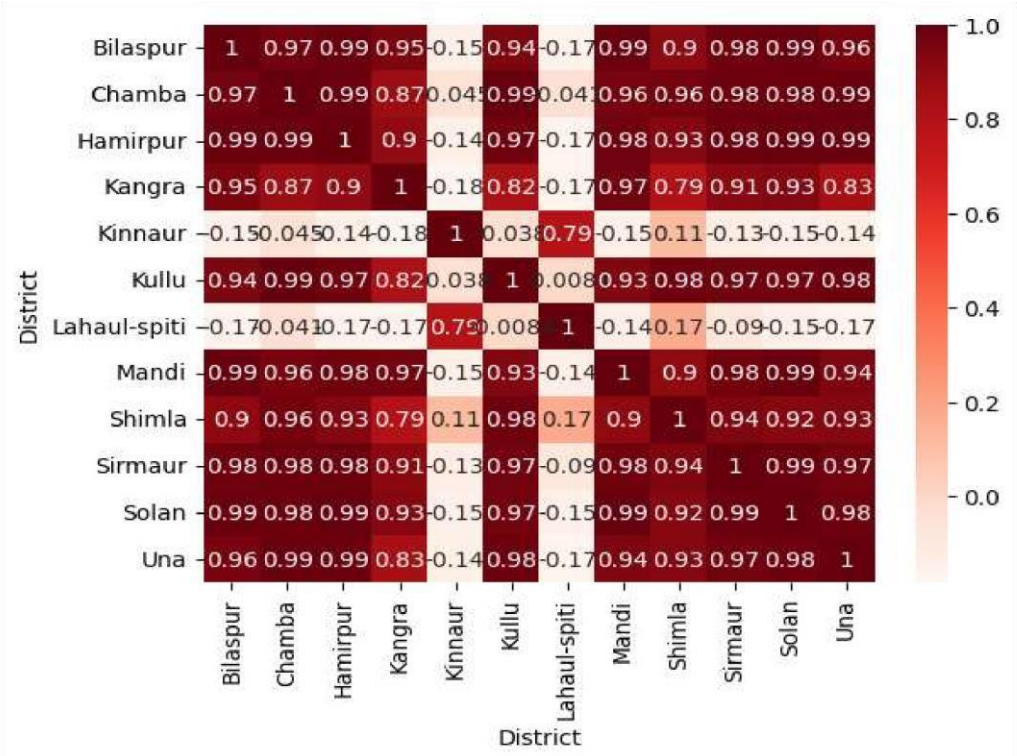
```
2  sns.heatmap(data[arr].corr(), annot=True, cmap = 'Reds')
   plt.show()
```



Comparison of total production between different districts using Heat map:

Like the heat map given above, we can also make a heat map to the relationship between two different districts, like the one shown below

```
3  dist_arr = ['Bilaspur', 'Chamba', "Hamirpur", "Kangra", "Kinnaur", "Kullu", "L
   ahaul-spiti", "Mandi", "Shimla", "Sirmaur", "Solan", "Una"]
   sns.heatmap(tdf[dist_arr].corr(), annot=True, cmap = 'Reds')
   plt.show()
```



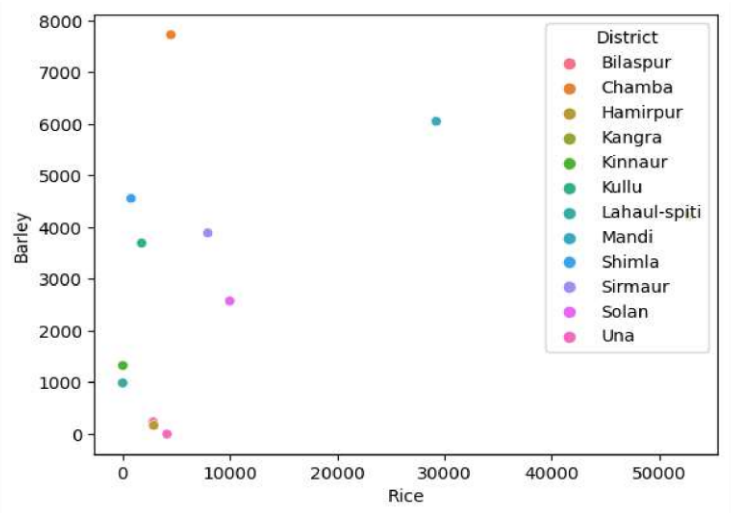
An individual scatter plot can be implemented on Python to understand the relationship between two different crops. An example has been shown below.

A scatter plot showing the relationship between Rice production and Barley production:

The given scatterplot gives us the relationship between the total production of Rice and Barley in all the districts. If we look at the production of Mandi, you can observe that it has the total highest production for Rice and Barley combined. (36000 metric tonnes approx.), while Lahaul-spiti has the lowest.

```
4  sns.scatterplot(x=data["Rice"],y=data["Barley"],hue=data.index)
```

```
Out[34]: <AxesSubplot: xlabel='Rice', ylabel='Barley'>
```

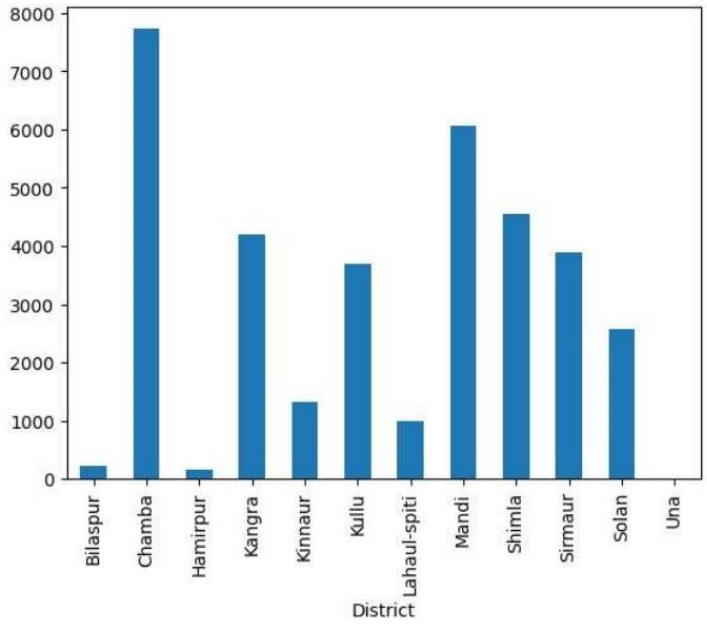


In the above scatter plot, you can also observe that the more left the data points are on the X-axis, the lesser the rice production and vice-versa. Also, higher the data points are on the Y-axis, more the production of Barley and vice -versa. Similarly, each scatter plot in the pair plot can be studied individually to understand the relationship between the production of any two crops in all the districts. A similar relationship study can also be conducted between any of the two districts.

The bar plots below give us an idea about the production of each crop in different districts**.**

Production of Barley:

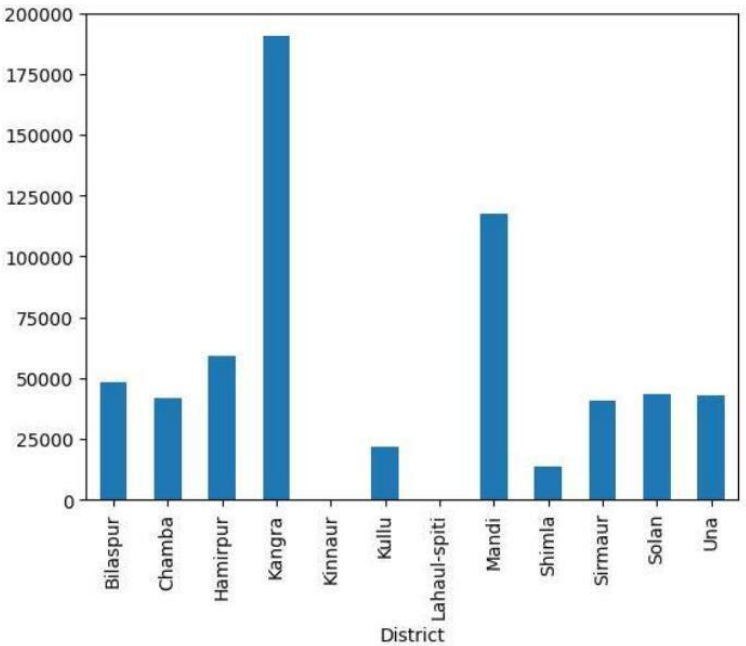
```
In [42]: data["Barley"].plot.bar()
Out[42]: <AxesSubplot: xlabel='District'>
```



From the above graph, we can deduce that Chamba is the largest producer of Barley while Una is the lowest, which is almost negligible.

Production of Wheat:

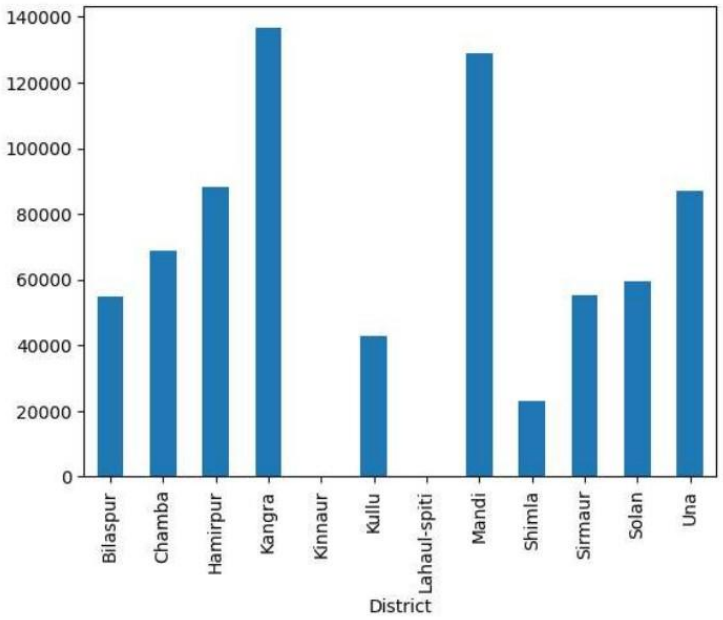
```
In [39]: data["Wheat"].plot.bar()
Out[39]: <AxesSubplot: xlabel='District'>
```



While Kangra is the largest producer of wheat, Kinnaur and Lahaul-Spiti has no production.

Production of Maize:

```
In [40]: data["Maize"].plot.bar()
Out[40]: <AxesSubplot: xlabel='District'>
```

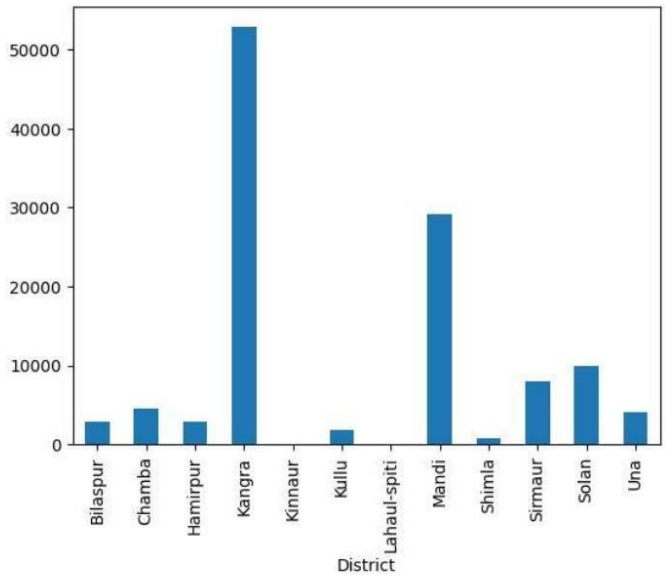


While Mandi had the largest production, Kinnaur and Lahaul-spiti have nil production values.

Production of Rice:

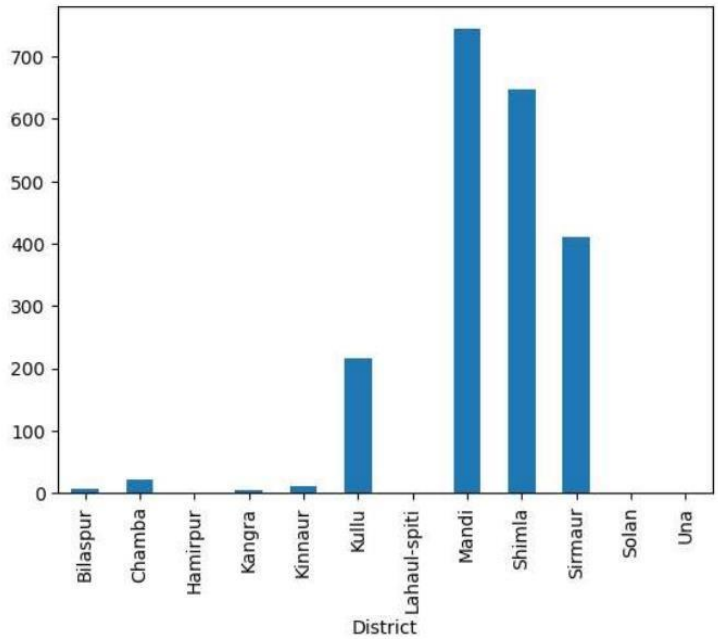
Kangra and Mandi have marginally high production values of rice when compared to the rest of the districts.

```
In [41]: data["Rice"].plot.bar()
Out[41]: <AxesSubplot: xlabel='District'>
```



Production of Ragi:

```
In [43]: data["Ragi"].plot.bar()
Out[43]: <AxesSubplot: xlabel='District'>
```

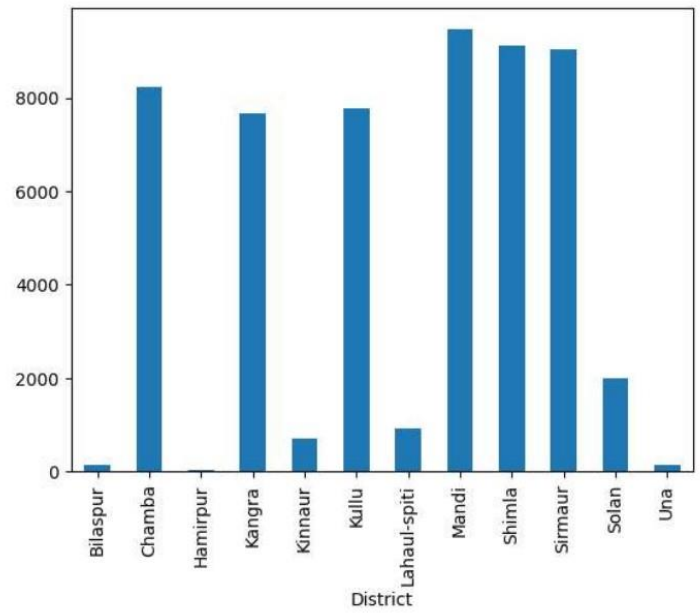


Only four districts, namely- Mandi, Shimla, Sirmaur and Kullu have sizable production values of Ragi while the rest of the districts have very low or negligible production values.

Production of Millets:

Almost all districts produce millets. However, Mandi, Shimla and Sirmaur are the largest producers. The production values of Pulses are very low in comparison to Rice, wheat, or maize.

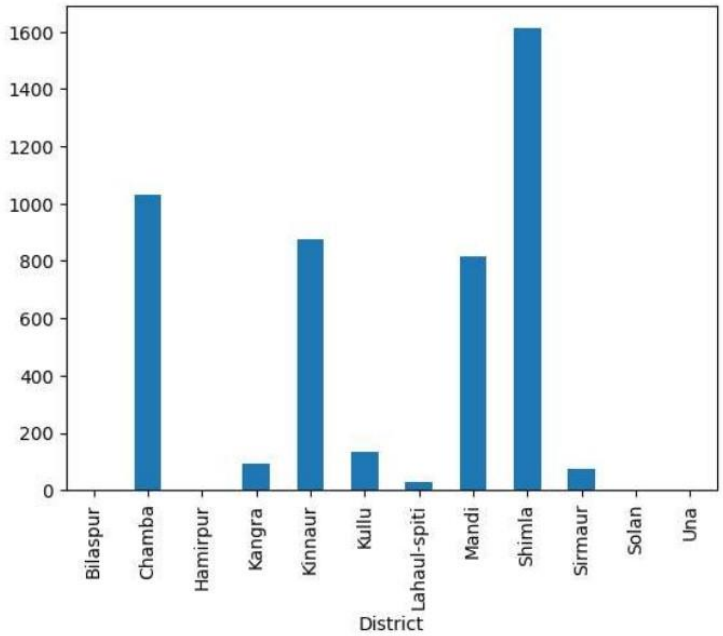
```
In [44]: data["Pulses"].plot.bar()
Out[44]: <AxesSubplot: xlabel='District'>
```



Production of Common Millets:

Millets have very low production in the districts. Shimla is the largest producer. While more than 4 districts have negligible or nil production

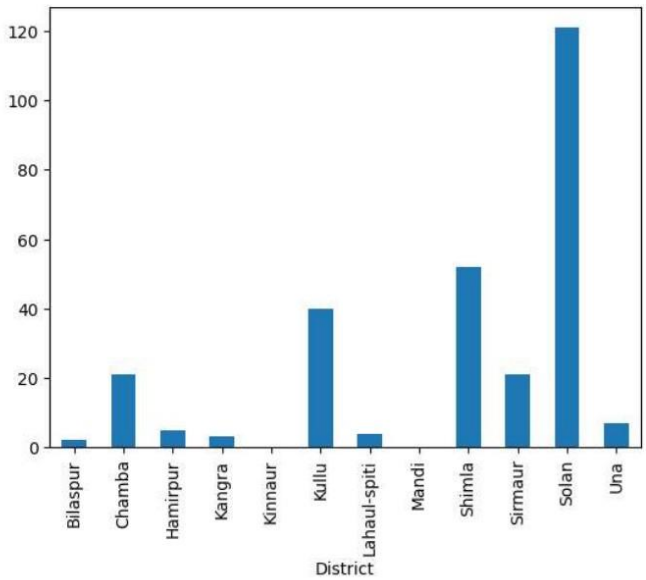
```
In [45]: data["Common millets"].plot.bar()
Out[45]: <AxesSubplot: xlabel='District'>
```



Production of Chillies:

Solan is the largest producer of chilies while mandi and Kinnaur have nil production.

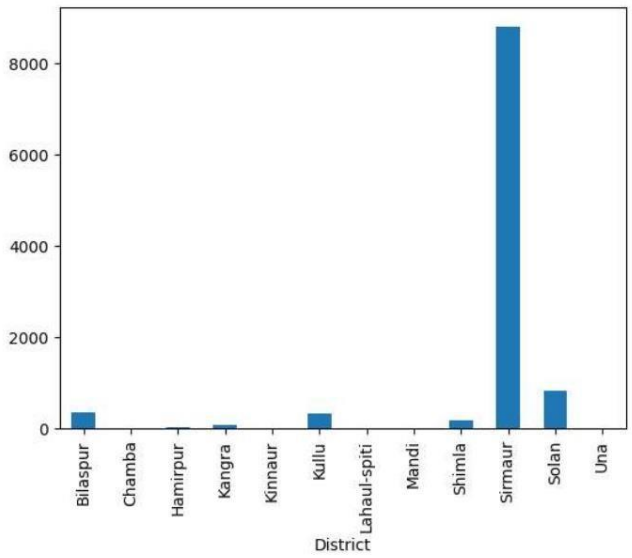
```
In [46]: data["Chillies"].plot.bar()
Out[46]: <AxesSubplot: xlabel='District'>
```



Production of Ginger:

Very few districts produce Ginger. Sirmaur is the largest producer.

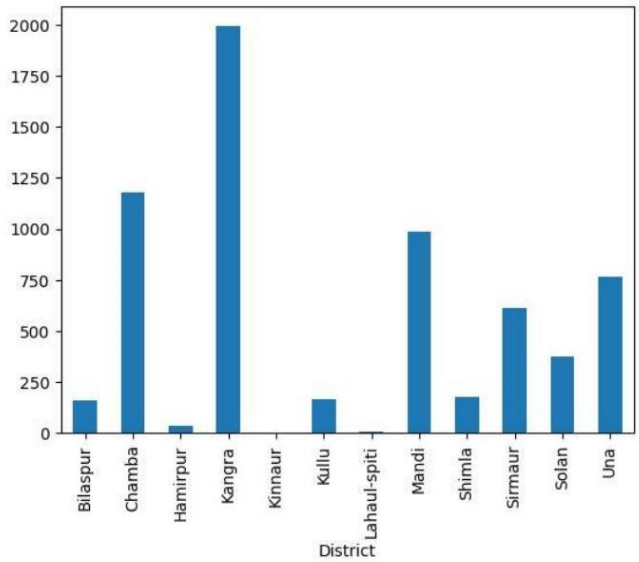
```
In [47]: data["Ginger"].plot.bar()
Out[47]: <AxesSubplot: xlabel='District'>
```



Production of Oil seeds:

Kangra is the largest producer of Oil seeds.

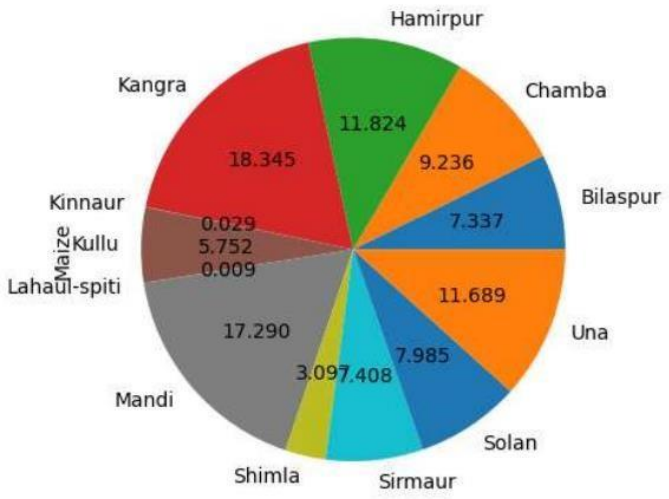
```
In [203]: data["Oil  seeds"].plot.bar()
Out[203]: <AxesSubplot: xlabel='District'>
```



Alternative representation of Maize production:

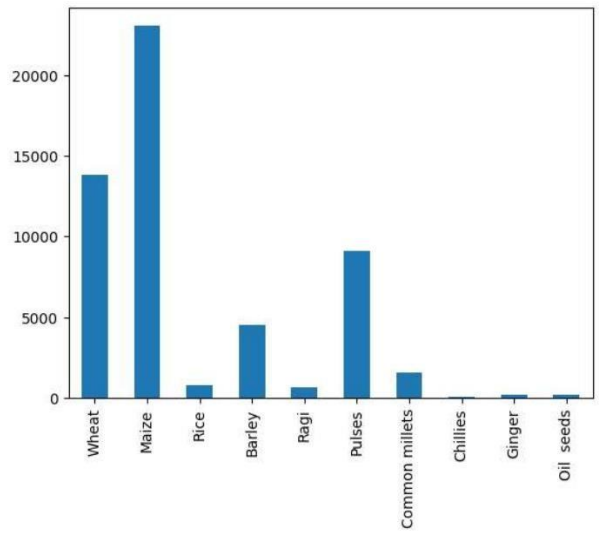
The above bar plots can also be represented using pie charts, For example, a pie chart depicting the production of maize in all the districts is given below.

```
In [204]: data["Maize"].plot.pie(autopct='%.3f')
Out[204]: <AxesSubplot: ylabel='Maize'>
```



An alternative for the above bar plots can be crop wise production study in each district. For example:

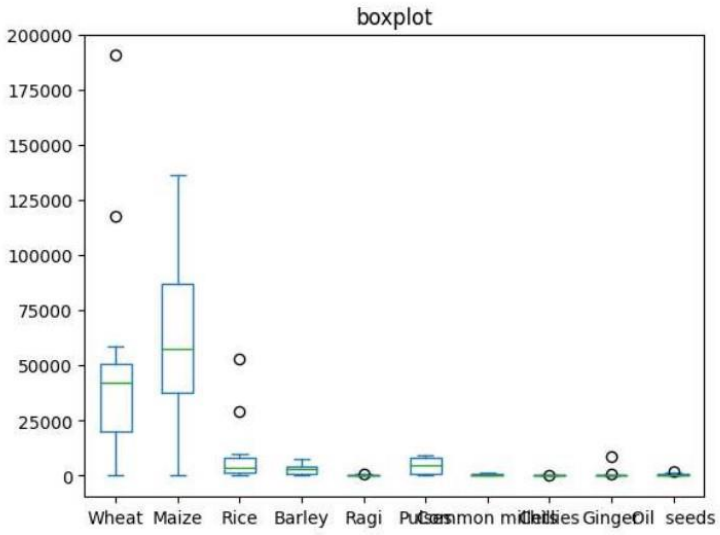
```
In [76]: tdf["Shimla"].plot.bar()
Out[76]: <AxesSubplot: >
```



The above bar plot is a representation of production values of different crops in Shimla. Wheat and Maize are primarily grown here.

Box Plot:

```
In [36]: arr = ['Wheat', 'Maize', "Rice", "Barley", "Ragi", "Pulses", "Common millets",
         "Chillies", "Ginger", "Oil  seeds"]
         boxplot = data[arr].plot(kind='box', title='boxplot')
```



A box plot is a great visualization tool to understand the median production values of each crop. From the above box plot we can infer that Maize has the highest median value, while crops like Ragi, Millets, Ginger etc. have very negligible production values in comparison.

Conclusions:

From the above detailed study, we come to the following conclusions:

(i) Wheat and Maize are the primarily grown crops in the given districts of Himachal Pradesh.

(ii) Kinnaur and Lahaul-spiti have very low agricultural produce.

(iii) The median production values of Maize and Wheat are the highest.

(iv) We understood the various objects and data types used in the dataset and have successfully inferred various conclusions using different visualizations.

Bibliography:

1. https://towardsdatascience.com/exploratory-data-analysis-edapython-87178e35b14
2. https://github.com/dphi-official/Data-

Preprocessing/blob/master/Data_Pre_processing.ipynb
3. https://www.analyticsvidhya.com/blog/2020/07/univariate-analysisvisualization-with-illustrations-in-python/
4. https://matplotlib.org/2.0.2/users/pyplot_tutorial.html

Source Code and Dataset:

https://drive.google.com/drive/folders/1nNXXekfpvlxp3Tu3Zk6Y_9bAgHFelyTD?usp=share_link