```
import pandas as pd
import numpy as np
import seaborn as sns                       #visualisation
import matplotlib.pyplot as plt             #visualisation
%matplotlib inline
sns.set(color_codes=True)
```

```
df = pd.read_csv("Top_1000_Companies_Dataset.csv")
# To display the top 5 rows
df.head(5)
```

	company_name	url	city	state	country	employees	
0	OpenAI	openai.com	San Francisco	CA	United States	655	http://\
1	Alchemy	alchemy.com	San Francisco	CA	United States	201	http://www.
2	dbt Labs	getdbt.com	Philadelphia	PA	United States	511	http://w
3	Wasabi Technologies	wasabi.com	Boston	MA	United States	355	http://www.linked
4	Whatnot	whatnot.com	Los Angeles	CA	United States	551	http://www.

5 rows × 23 columns

```
df.tail(5)                      # To display the botton 5 rows
```

	company_name	url	city	state	country	employees	
995	Bond Vet	bondvet.com	New York	NY	United States	292	http://ww
996	CompStak	compstak.com	New York	NY	United States	153	http://\
997	Quantum Metric	quantummetric.com	Monument	CO	United States	528	http://
998	Fathom (YC W21)	fathom.video	San Francisco	CA	USA	96	http://ww
999	Hone	honehq.com	San Francisco	CA	United States	179	http

5 rows × 23 columns

```
df.dtypes
```

```
company_name           object
url                    object
city                   object
state                  object
country                object
employees               int64
linkedin_url           object
founded               float64
Industry               object
GrowjoRanking           int64
Previous Ranking        int64
estimated_revenues    float64
job_openings          float64
keywords               object
LeadInvestors          object
Accelerator            object
btype                  object
valuation             float64
total_funding          object
product_url            object
indeed_url             object
growth_percentage      object
contact_info           object
dtype: object
```

```
df = df.drop(['founded', 'LeadInvestors', 'Accelerator', 'btype', 'valuation', 'total_funding'], axis=1)
df.head(5)
```

	company_name	url	city	state	country	employees	
0	OpenAI	openai.com	San Francisco	CA	United States	655	http://\
1	Alchemy	alchemy.com	San Francisco	CA	United States	201	http://www.
2	dbt Labs	getdbt.com	Philadelphia	PA	United States	511	http://w
3	Wasabi Technologies	wasabi.com	Boston	MA	United States	355	http://www.linked
4	Whatnot	whatnot.com	Los Angeles	CA	United States	551	http://www.

```
df = df.rename(columns={"company_name": "NAME", "url": "company_url", "city": "Place", "employees": "List_employees" })
df.head(5)
```

	NAME	company_url	Place	state	country	List_employees	
0	OpenAI	openai.com	San Francisco	CA	United States	655	h
1	Alchemy	alchemy.com	San Francisco	CA	United States	201	http://
2	dbt Labs	getdbt.com	Philadelphia	PA	United States	511	ht

```
df.shape
```

```
(1000, 17)
```

```
duplicate_rows_df = df[df.duplicated()]
print("number of duplicate rows: ", duplicate_rows_df.shape)
```

```
number of duplicate rows:  (0, 17)
```

```
df.count()      # Used to count the number of rows
```

```
NAME                 1000
company_url           999
Place                 999
state                 812
country               994
List_employees       1000
linkedin_url          997
Industry              997
GrowjoRanking        1000
Previous Ranking     1000
estimated_revenues    972
job_openings          938
keywords              368
product_url          1000
indeed_url           1000
growth_percentage    1000
contact_info         1000
dtype: int64
```

```
df = df.drop_duplicates()
df.head(5)
```

```
NAME  company_url      Place  state  country  List_employees
```

```
df.count()
```

```
NAME                 1000
company_url           999
Place                 999
state                 812
country               994
List_employees       1000
linkedin_url          997
Industry              997
GrowjoRanking        1000
Previous Ranking     1000
estimated_revenues    972
job_openings          938
keywords              368
product_url          1000
indeed_url           1000
growth_percentage    1000
contact_info         1000
dtype: int64
```

```
print(df.isnull().sum())
```

```
NAME                   0
company_url            1
Place                  1
state                188
country                6
List_employees         0
linkedin_url           3
Industry               3
GrowjoRanking          0
Previous Ranking       0
estimated_revenues    28
job_openings          62
keywords             632
product_url            0
indeed_url             0
growth_percentage      0
contact_info           0
dtype: int64
```

```
df = df.dropna()    # Dropping the missing values.
df.count()
```

```
NAME                 335
company_url          335
Place                335
state                335
country              335
List_employees       335
linkedin_url         335
Industry             335
GrowjoRanking        335
Previous Ranking     335
estimated_revenues   335
job_openings         335
keywords             335
product_url          335
indeed_url           335
growth_percentage    335
contact_info         335
dtype: int64
```

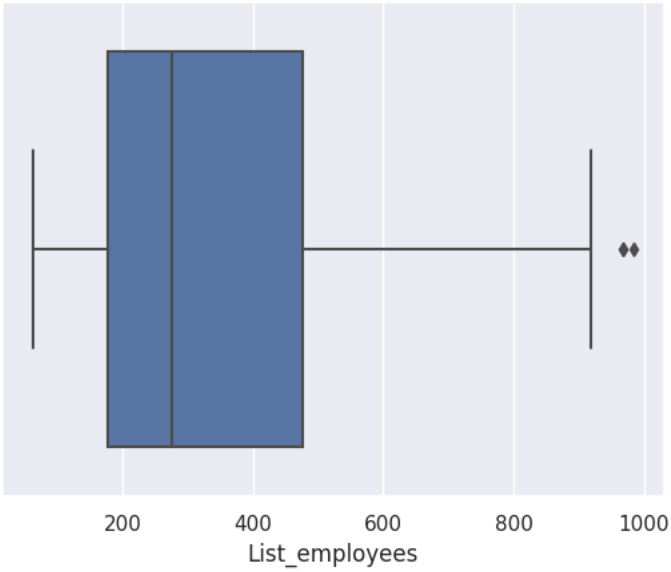
```
print(df.isnull().sum())   # After dropping the values
```

```
NAME                 0
company_url          0
Place                0
state                0
country              0
List_employees       0
linkedin_url         0
Industry             0
GrowjoRanking        0
```

```
Previous Ranking      0
estimated_revenues    0
job_openings          0
keywords              0
product_url           0
indeed_url            0
growth_percentage     0
contact_info          0
dtype: int64
```

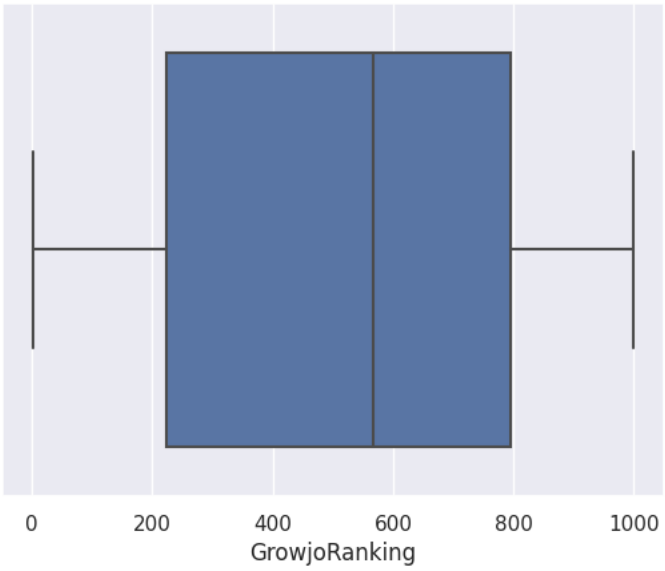
```
sns.boxplot(x=df['List_employees'])
```

```
<Axes: xlabel='List_employees'>
```



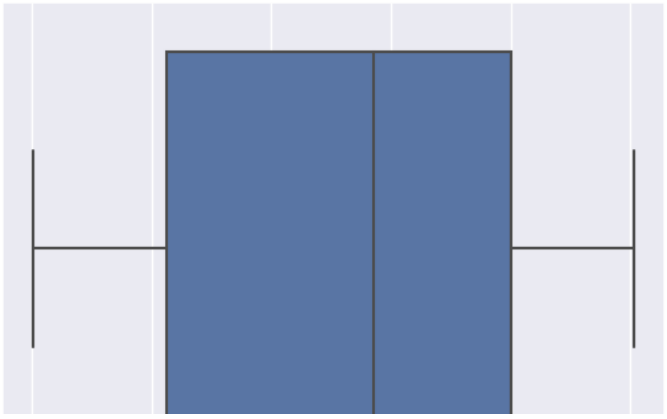

```
sns.boxplot(x=df['GrowjoRanking'])
```

```
<Axes: xlabel='GrowjoRanking'>
```




```
sns.boxplot(x=df['Previous Ranking'])
```

```
<Axes: xlabel='Previous Ranking'>
```



```
Q1 = df.quantile(0.25)
Q3 = df.quantile(0.75)
IQR = Q3 - Q1
print(IQR)
```

```
List_employees             298.0
GrowjoRanking              573.0
Previous Ranking           575.0
estimated_revenues    68424375.0
job_openings                22.5
dtype: float64
<ipython-input-21-d7397e803310>:1: FutureWarning: The default value of numeric_only in DataFrame.quantile is deprecated. In a future ver
  Q1 = df.quantile(0.25)
<ipython-input-21-d7397e803310>:2: FutureWarning: The default value of numeric_only in DataFrame.quantile is deprecated. In a future ver
  Q3 = df.quantile(0.75)
```

```
df = df[~((df < (Q1 - 1.5 * IQR)) |(df > (Q3 + 1.5 * IQR))).any(axis=1)]
df.shape
```

```
<ipython-input-22-f4e1682787c4>:1: FutureWarning: Automatic reindexing on DataFrame vs Series comparisons is deprecated and will raise \
  df = df[~((df < (Q1 - 1.5 * IQR)) |(df > (Q3 + 1.5 * IQR))).any(axis=1)]
(300, 17)
```

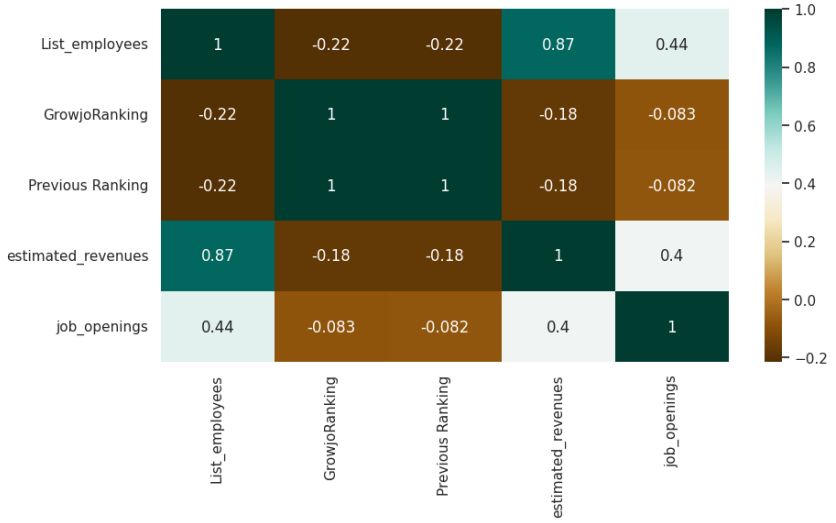
```
df.List_employees.value_counts().nlargest(40).plot(kind='bar', figsize=(10,5))
plt.title("companies growth")
plt.ylabel('List_employees')
plt.xlabel('GrowjoRanking');
```

companies growth

```
plt.figure(figsize=(10,5))
c= df.corr()
sns.heatmap(c,cmap="BrBG",annot=True)
c
```

```
<ipython-input-24-a44b43776930>:2: FutureWarning: The default value of numeric_onl
  c= df.corr()
```

	List_employees	GrowjoRanking	Previous Ranking	estimated_revenues	j
List_employees	1.000000	-0.215649	-0.215511	0.872642	
GrowjoRanking	-0.215649	1.000000	0.999998	-0.179031	
Previous Ranking	-0.215511	0.999998	1.000000	-0.178917	
estimated_revenues	0.872642	-0.179031	-0.178917	1.000000	
job_openings	0.443652	-0.082724	-0.082419	0.404168	




```
fig, ax = plt.subplots(figsize=(10,6))
ax.scatter(df['GrowjoRanking'], df['List_employees'])
ax.set_xlabel('GrowjoRanking')
ax.set_ylabel('List_employees')
plt.show()
```

We have used matplotlib for visua;llisatrion of data set . the csv file is to be read and displayed the top five rows use pandas and numpy library in python .

we have used top 1000 companies data set for exploratory data analysis . all the components imported list from EDA has loaded the data in a data frame by treading the csv file and uploaded with in the notebook, then we have to check the type of data s of the data set as library takes only integer data type and then we going to make change by dropping unknown colounns and reaming the used coloumns. Duplicate rows removes the unused data or the common data in the rows and dropping the missing null values and matplotlib uses the data and makes it remove the null values and then plot different features againest one another through histogram and scatter plot,