In [1]: import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

In [3]: file_path = "SAMPANN_statewise_dataset_0.csv" # Update the correct path
 df = pd.read_csv(file_path)

In [5]: **df**

Out[5]:

0			Pensioner	Pensioner	Pensioner	Pensioner	Female Pensioner	IDA Pensioner	IDA Male Pensioner
	Andaman and Nicobar	185	111	74	31	29	2	154	137
1	Andhra Pradesh	20007	13158	6849	2494	2426	60	17513	16074
2	Assam	7521	4528	2993	1140	1098	41	6381	5904
3	Bihar	12427	8510	3917	3941	3248	63	8486	7441
4	Chhattisgarh	3478	2048	1430	595	592	3	2883	2667
5	Delhi Region	23708	15900	7808	16595	14338	2088	7113	6568
6	Gujarat	30690	21460	9230	4142	3748	385	26548	23823
7	Himachal Pradesh	3049	2187	862	345	331	12	2704	2536
8	Haryana	6372	4360	2012	827	803	24	5545	5124
9	Jammu and Kashmir	3109	2312	797	440	435	5	2669	2442
10	Jharkhand	4475	3087	1388	889	870	17	3586	3440
11	Karnataka	30259	20563	9696	5544	5047	480	24715	20141
12	Kerala	26974	19031	7943	4880	4351	443	22094	18179
13	Kolkota	17359	11756	5603	3229	2981	221	14130	12269
14	Madhya Pradesh	20125	12670	7455	4150	3973	174	15975	14893
15	Maharashtra	42966	29822	13144	8007	7443	557	34959	30129
16	Mumbai	8159	4191	3968	8158	6500	1064	1	1
17	North East II	482	424	58	5	5	0	477	436
18	North East	2597	1489	1108	372	354	17	2225	1905
19	Orissa	7108	4702	2406	916	907	4	6192	5973
20	Punjab	12008	8486	3522	1597	1522	61	10411	9441
21	Rajasthan	16607	11611	4996	2918	2879	37	13689	13343
22	Tamil Nadu	52032	35409	16623	10643	8908	945	41389	32721
23	Telangana	17235	11633	5602	2375	2152	203	14860	12793
24	UP East	23055	15365	7690	5619	5345	266	17436	16235
25	UP West	12351	8062	4289	2817	2758	57	9534	9109
26	Uttarakhand	3072	2043	1029	440	425	10	2632	2462
27	West Bengal	19039	11393	7646	4627	4456	166	14412	13362

In [7]: df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 28 entries, 0 to 27
Data columns (total 10 columns):

#	Column	Non-Null Count	Dtype		
0	State Name	28 non-null	object		
1	Total Pensioner	28 non-null	int64		
2	Normal Pensioner	28 non-null	int64		
3	Family Pensioner	28 non-null	int64		
4	CDA Pensioner	28 non-null	int64		
5	CDA Male Pensioner	28 non-null	int64		
6	CDA Female Pensioner	28 non-null	int64		
7	IDA Pensioner	28 non-null	int64		
8	IDA Male Pensioner	28 non-null	int64		
9	IDA Female Pensioner	28 non-null	int64		

dtypes: int64(9), object(1) memory usage: 2.3+ KB

In [9]: df.head

Out[9]:		und method NDFrame.head o	of	State Name Tot	al Pensioner Norma	l Pensioner Fam
	-		105	111	7.4	
	0	Andaman and Nicobar	185	111	74	
	1	Andhra Pradesh	20007	13158	6849	
	2	Assam	7521	4528	2993	
	3	Bihar	12427	8510	3917	
	4	Chhattisgarh	3478	2048	1430	
	5	Delhi Region	23708	15900	7808	
	6	Gujarat	30690	21460	9230	
	7	Himachal Pradesh	3049	2187	862	
	8	Haryana	6372	4360	2012	
	9	Jammu and Kashmir	3109	2312	797	
	10	Jharkhand	4475	3087	1388	
	11	Karnataka		20563	9696	
			30259			
	12	Kerala	26974	19031	7943	
	13	Kolkota	17359	11756	5603	
	14	Madhya Pradesh	20125	12670	7455	
	15	Maharashtra	42966	29822	13144	
	16	Mumbai	8159	4191	3968	
	17	North East II	482	424	58	
	18	North East	2597	1489	1108	
	19	Orissa	7108	4702	2406	
	20	Punjab	12008	8486	3522	
	21	Rajasthan	16607	11611	4996	
		-				
	22	Tamil Nadu	52032	35409	16623	
	23	Telangana	17235	11633	5602	
	24	UP East	23055	15365	7690	
	25	UP West	12351	8062	4289	
	26	Uttarakhand	3072	2043	1029	
	27	West Bengal	19039	11393	7646	
	21	west bengat	19039	11393	7040	
		CDA Pensioner CDA Male	Pensioner CDA	Female Pensioner	IDA Pensioner \	
	0	31	29	2	154	
	1	2494	2426	60	17513	
	2	1140	1098	41	6381	
	3	3941	3248	63	8486	
	4	595	592	3	2883	
	5	16595	14338	2088	7113	
	6	4142	3748	385	26548	
	7					
		345	331	12	2704	
	8	827	803	24	5545	
	9	440	435	5	2669	
	10	889	870	17	3586	
	11	5544	5047	480	24715	
	12	4880	4351	443	22094	
	13	3229	2981	221	14130	
	14	4150	3973	174	15975	
	15	8007	7443	557	34959	
	16	8158	6500	1064	1	
	17	5	5	0	477	
	18	372	354	17	2225	
	19	916	907	4	6192	
	20	1597	1522	61	10411	
	21	2918	2879	37	13689	
	22	10643	8908	945	41389	
	23	2375	2152	203	14860	
	24	5619	5345	266	17436	
	25	2817	2758	57	9534	
	26	440	425	10	2632	
	27	4627	4456	166	14412	
		IDA Male Pensioner IDA	Female Pensione	r		
	0	137	1	5		
	1	16074	143			
	2	5904	46			
	3	7441	33			
	4	2667	21			
	5	6568	52	9		
	6	23823	267			
	7	2536	16			
	8	5124	41			
	9	2442	22			
	10	3440	14	2		
	11	20141	453			
	12	18179	382			
			302	-		

12269	1785
14893	1078
30129	4820
1	0
436	41
1905	307
5973	199
9441	960
13343	329
32721	7297
12793	2021
16235	1158
9109	421
2462	146
13362	1021 >
	14893 30129

In [11]: df.describe

0+[11].	-ho	und mothod NDEr	sama dasan	ibo of		C+-	ata Nama	Total	Donsioner	No rmo 1	Doncionor		
Out[II]:		und method NDFr ily Pensioner	The or		Sta	ate Name	Totat	Pensioner	Normat	Pensioner			
	0	Andaman and Ni	185			111		7	4				
	1	Andhra Pr		20		13158		6849					
	2		7		4528		299						
	3		12		8510		3917						
	4	Chhatt	3		2048		1430						
	5	Delhi	23		15900		7808						
	6	G	30		21460		9230						
	7	Himachal P	3		2187		862						
	8	H	6372			4360		201	12				
	9	Jammu and K	3109			2312		797					
	10	Jha	4475			3087		1388					
	11		nataka		259		20563		969				
	12		Kerala		974		19031		794				
	13		Kolkota		7359	11756			5603				
	14 15	Madhya P	radesn rashtra)125 2966		12670 29822		745 1314				
	16		Mumbai		3159		4191		396				
	17	North E			482		424						
	18		h East		2597		1489		110	8 8			
	19		Orissa		108		4702						
	20		Punjab		2008		8486		240 352				
	21		astĥan		6607		11611		499				
	22	_	l Nadu	52	2032		35409		1662				
	23	Tel	angana	17	235		11633		560	2			
	24	U	IP East		8055		15365		769	0			
	25	U	JP West	12	2351		8062		428				
	26		akhand		3072		2043		102				
	27	West	Bengal	19	9039		11393		764	6			
		CDA Pensioner	CDA Male	Pensioner	CDA F	emale I	Pensioner	TDA F	Pensioner	\			
	0	31		29			2		154	`			
	1	2494		2426			60		17513				
	2	1140		1098			41		6381				
	3	3941		3248			63		8486				
	4	595		592			3		2883				
	5	16595		14338			2088		7113				
	6	4142		3748			385		26548				
	7	345		331			12		2704				
	8	827		803			24		5545				
	9 10	440 889		435 870			5 17		2669 3586				
	11	5544		5047			480		24715				
	12	4880		4351			443		22094				
	13	3229		2981			221		14130				
	14	4150		3973			174		15975				
	15	8007		7443			557		34959				
	16	8158		6500			1064		1				
	17	5		5			0		477				
	18	372		354			17		2225				
	19	916		907			4		6192				
	20	1597		1522			61		10411				
	21	2918		2879			37		13689				
	22	10643		8908			945		41389				
	23 24	2375 5610		2152			203		14860 17436				
	24 25	5619 2817		5345 2758			266 57		17436 9534				
	26	440		425			10		2632				
	27	4627		4456			166		14412				
	•	IDA Male Pensi		Female Per									
	0	1	137		1420								
	1		.6074		1430								
	2		5904 7441		461 333								
	4		2667		216								
	5		6568		529								
	6		23823		2673								
	7		2536		167								
	8		5124		416								
	9		2442		227								
	10		3440		142								
	11		20141		4535								
	12	1	.8179		3826								

```
13
                   12269
                                             1785
14
                   14893
                                             1078
15
                   30129
                                             4820
16
                       1
                                               0
17
                     436
                                               41
18
                    1905
                                             307
19
                    5973
                                             199
                    9441
                                              960
20
21
                   13343
                                             329
22
                   32721
                                             7297
                   12793
23
                                             2021
                                             1158
24
                   16235
25
                    9109
                                             421
26
                    2462
                                             146
27
                   13362
                                             1021 >
```

```
In [13]: df.columns
```

```
In [15]: df["CDA Percentage"] = (df["CDA Pensioner"] / df["Total Pensioner"]) * 100
    df["IDA Percentage"] = (df["IDA Pensioner"] / df["Total Pensioner"]) * 100
    df["Male Percentage"] = ((df["CDA Male Pensioner"] + df["IDA Male Pensioner"]) / df["Total Pensioner"] + df["Female Percentage"] = ((df["CDA Female Pensioner"] + df["IDA Female Pensioner"]) / df["Total Pensioner"]
```

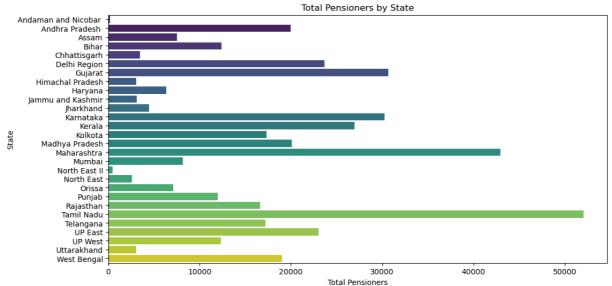
In [17]: df.columns

```
In [19]: plt.figure(figsize=(12, 6))
    sns.barplot(y=df["State Name"], x=df["Total Pensioner"], palette="viridis")
    plt.xlabel("Total Pensioners")
    plt.ylabel("State")
    plt.title("Total Pensioners by State")
    plt.show()
```

C:\Users\ASUS\AppData\Local\Temp\ipykernel_22076\437390100.py:2: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `y` variable to `hue` and set `legend=False` for the same effect.

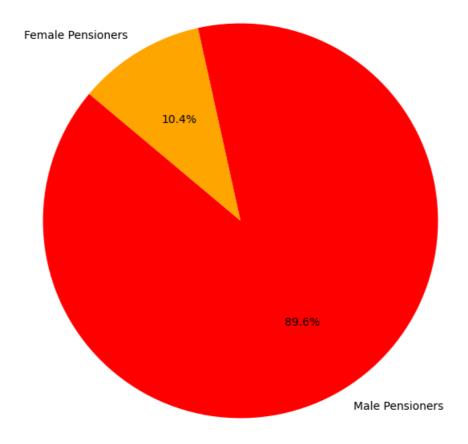
sns.barplot(y=df["State Name"], x=df["Total Pensioner"], palette="viridis")



```
In [23]: plt.figure(figsize=(8, 8))
   gender_counts = [df["CDA Male Pensioner"].sum() + df["IDA Male Pensioner"].sum(),
```

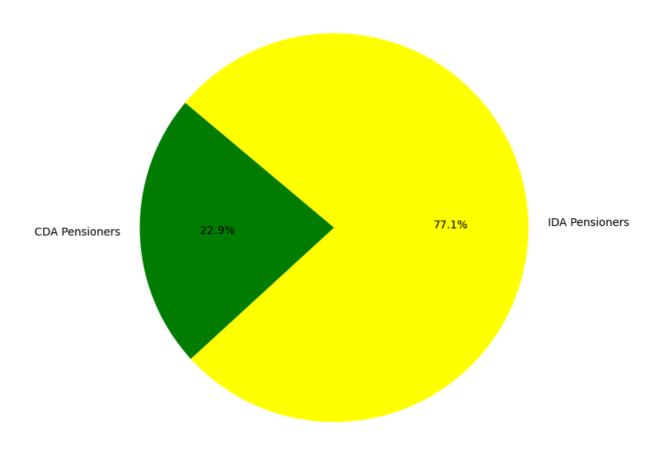
```
df["CDA Female Pensioner"].sum() + df["IDA Female Pensioner"].sum()]
labels = ["Male Pensioners", "Female Pensioners"]
colors = ["red", "orange"]
plt.pie(gender_counts, labels=labels, autopct="%1.1f%", colors=colors, startangle=140)
plt.title("Overall Gender Distribution of Pensioners")
plt.show()
```

Overall Gender Distribution of Pensioners

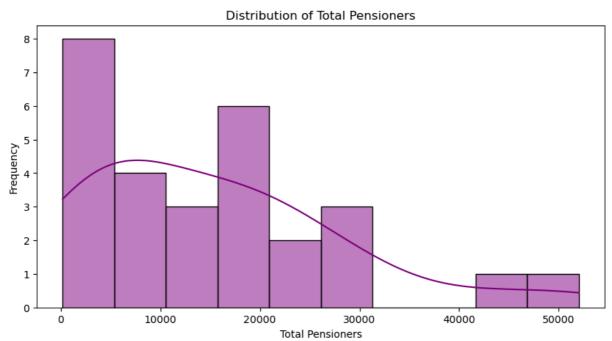


```
In [27]: plt.figure(figsize=(8, 8))
    pensioner_counts = [df["CDA Pensioner"].sum(), df["IDA Pensioner"].sum()]
    labels = ["CDA Pensioners", "IDA Pensioners"]
    colors = ["green", "yellow"]
    plt.pie(pensioner_counts, labels=labels, autopct="%1.1f%", colors=colors, startangle=140)
    plt.title("Distribution of Pensioner Types (CDA vs IDA)")
    plt.show()
```

Distribution of Pensioner Types (CDA vs IDA)



```
In [29]: plt.figure(figsize=(10, 5))
    sns.histplot(df["Total Pensioner"], bins=10, kde=True, color="purple")
    plt.xlabel("Total Pensioners")
    plt.ylabel("Frequency")
    plt.title("Distribution of Total Pensioners")
    plt.show()
```



```
In [33]: numeric_df = df.select_dtypes(include=["number"])
   plt.figure(figsize=(10, 6))
   corr_matrix = numeric_df.corr()
```

```
sns.heatmap(corr_matrix, annot=True, cmap="coolwarm", fmt=".2f", linewidths=0.5)
plt.title("Correlation Heatmap of Pensioner Data")
plt.show()
```

Correlation Heatmap of Pensioner Data														1.00		
Total Pensioner -	1.00	1.00	0.99	0.71	0.73	0.52	0.97	0.96	0.91	0.08	-0.08	-0.49	0.54			1.00
Normal Pensioner -	1.00	1.00	0.98	0.69	0.71	0.50	0.97	0.96	0.91	0.05	-0.05	-0.48	0.54		_	0.75
Family Pensioner -	0.99	0.98	1.00	0.73	0.75	0.54	0.95	0.94	0.88	0.14	-0.14	-0.49	0.53			
CDA Pensioner -	0.71	0.69	0.73	1.00	1.00	0.94	0.51	0.50	0.49	0.66	-0.66	-0.53	0.46			0.50
CDA Male Pensioner -	0.73	0.71	0.75	1.00	1.00	0.93	0.53	0.53	0.50	0.63	-0.63	-0.50	0.45			- 0.25
CDA Female Pensioner -	0.52	0.50		0.94	0.93	1.00	0.29	0.27	0.35	0.73	-0.73	-0.55	0.52			0.23
IDA Pensioner -	0.97	0.97	0.95	0.51	0.53	0.29	1.00	1.00	0.93	-0.14	0.14	-0.40	0.49		-	0.00
IDA Male Pensioner -	0.96	0.96	0.94	0.50	0.53	0.27	1.00	1.00	0.90	-0.15	0.15	-0.35	0.44			
IDA Female Pensioner -	0.91	0.91	0.88	0.49	0.50	0.35	0.93	0.90	1.00	-0.11	0.11	-0.60	0.70		-	-0.25
CDA Percentage -	0.08	0.05	0.14	0.66	0.63	0.73	-0.14	-0.15	-0.11	1.00	-1.00	-0.44	0.21			-0.50
IDA Percentage -	-0.08	-0.05	-0.14	-0.66	-0.63	-0.73	0.14	0.15	0.11	-1.00	1.00	0.44	-0.21			
Male Percentage -	-0.49	-0.48	-0.49	-0.53	-0.50	-0.55	-0.40	-0.35	-0.60	-0.44	0.44	1.00	-0.87		-	-0.75
Female Percentage -		0.54	0.53	0.46	0.45	0.52	0.49	0.44	0.70	0.21	-0.21	-0.87	1.00			
	Total Pensioner -	Normal Pensioner -	Family Pensioner -	CDA Pensioner -	CDA Male Pensioner -	CDA Female Pensioner -	IDA Pensioner -	IDA Male Pensioner -	IDA Female Pensioner -	CDA Percentage -	IDA Percentage -	Male Percentage -	Female Percentage -			1.00

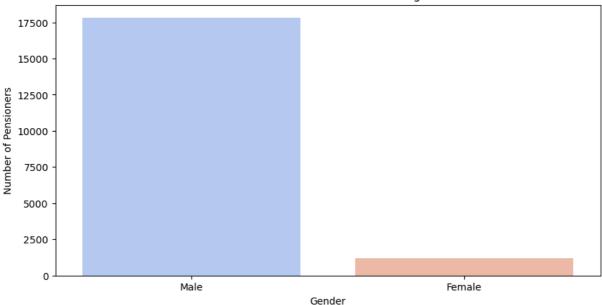
```
sns.barplot(x=["Male", "Female"], y=[male_pensioners, female_pensioners], palette="coolwarm")
plt.xlabel("Gender")
plt.ylabel("Number of Pensioners")
plt.title(f"Gender Distribution in {state}")
plt.show()
```

C:\Users\ASUS\AppData\Local\Temp\ipykernel_22076\1324546647.py:3: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

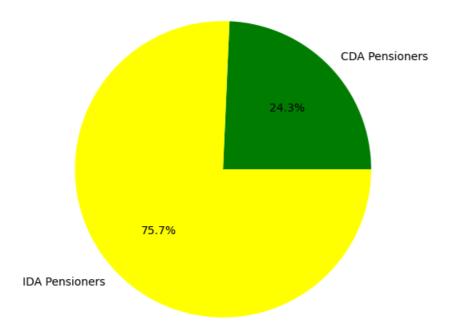
sns.barplot(x=["Male", "Female"], y=[male_pensioners, female_pensioners], palette="coolwarm")





```
In [51]: plt.figure(figsize=(6, 6))
   plt.pie([state_df["CDA Pensioner"].values[0], state_df["IDA Pensioner"].values[0]],labels=["CDA
   plt.title(f"Pensioner Type Distribution in {state}")
   plt.show()
```

Pensioner Type Distribution in West Bengal



```
In [55]: df.to_csv("SAMPANN_statewise_dataset_modified.csv", index=False)#save the modified dataset

In [61]: from sklearn.cluster import KMeans
    from sklearn.preprocessing import StandardScaler
    from sklearn.decomposition import PCA

In [65]: numeric_df = df.select_dtypes(include=["number"])
    scaler = StandardScaler()
    scaled_data = scaler.fit_transform(numeric_df.drop(columns=["Total Pensioner"]))

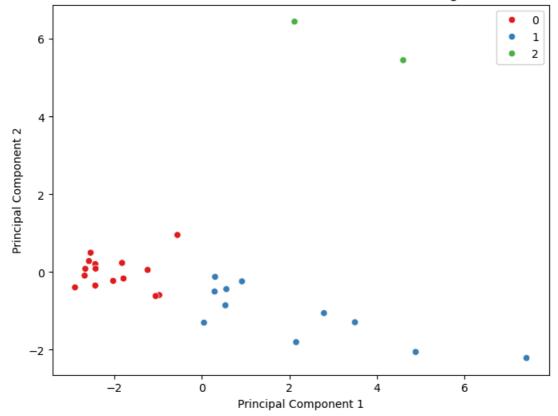
In [69]: kmeans = KMeans(n_clusters=3, random_state=42)
    df["Cluster"] = kmeans.fit_predict(scaled_data)
```

C:\Users\ASUS\anaconda3\Lib\site-packages\sklearn\cluster_kmeans.py:1429: UserWarning: KMeans i
s known to have a memory leak on Windows with MKL, when there are less chunks than available thr
eads. You can avoid it by setting the environment variable OMP_NUM_THREADS=1.
 warnings.warn(

```
In [71]: pca = PCA(n_components=2)
    pca_result = pca.fit_transform(scaled_data)
    df["PCA1"] = pca_result[:, 0]
    df["PCA2"] = pca_result[:, 1]

In [73]: plt.figure(figsize=(8, 6))
    sns.scatterplot(x=df["PCA1"], y=df["PCA2"], hue=df["Cluster"], palette="Set1")
    plt.xlabel("Principal Component 1")
    plt.ylabel("Principal Component 2")
    plt.title("PCA of Pensioner Data with K-Means Clustering")
    plt.legend()
    plt.show()
```

PCA of Pensioner Data with K-Means Clustering



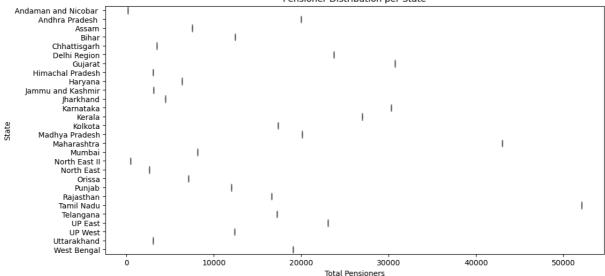
```
In [77]: plt.figure(figsize=(12, 6))
    sns.boxplot(x=df["Total Pensioner"], y=df["State Name"], palette="coolwarm")
    plt.xlabel("Total Pensioners")
    plt.ylabel("State")
    plt.title("Pensioner Distribution per State")
    plt.show()
```

C:\Users\ASUS\AppData\Local\Temp\ipykernel_22076\2207481037.py:2: FutureWarning:

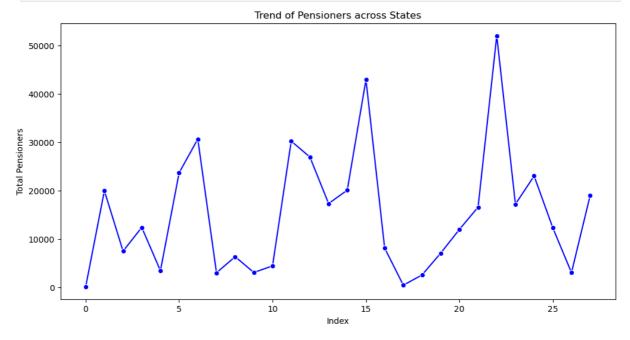
Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `y` variable to `hue` and set `legend=False` for the same effect.

sns.boxplot(x=df["Total Pensioner"], y=df["State Name"], palette="coolwarm")

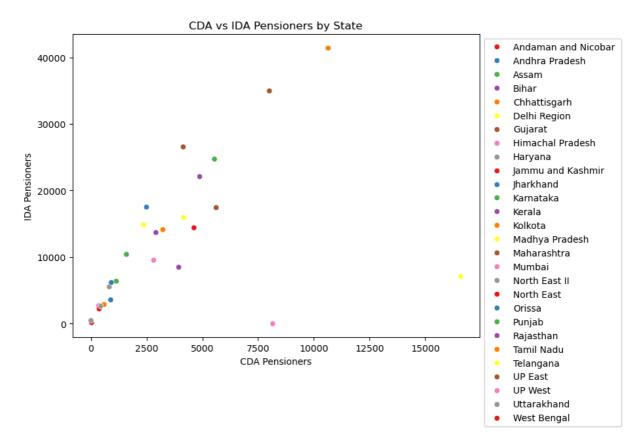
Pensioner Distribution per State



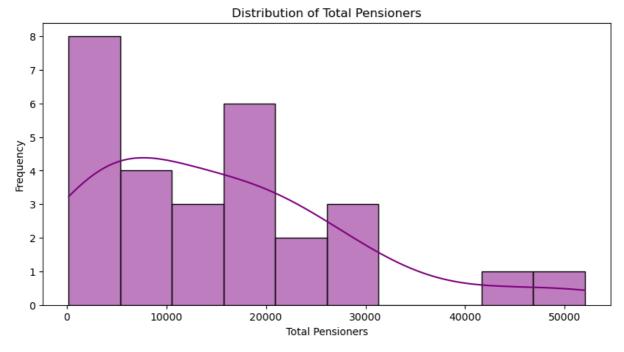
```
In [79]: plt.figure(figsize=(12, 6))
sns.lineplot(x=df.index, y=df["Total Pensioner"], marker="o", linestyle="-", color="blue")
plt.xlabel("Index")
plt.ylabel("Total Pensioners")
plt.title("Trend of Pensioners across States")
plt.show()
```



```
In [81]: plt.figure(figsize=(8, 6))
    sns.scatterplot(x=df["CDA Pensioner"], y=df["IDA Pensioner"], hue=df["State Name"], palette="Se
    plt.xlabel("CDA Pensioners")
    plt.ylabel("IDA Pensioners")
    plt.title("CDA vs IDA Pensioners by State")
    plt.legend(loc="best", bbox_to_anchor=(1, 1))
    plt.show()
```







In []: