In [1]:
```
import seaborn as sns
sns.set()
sns.set(style="darkgrid")
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import warnings
warnings.filterwarnings("ignore")
plt.rcParams["figure.figsize"]=(10,10)
```

In [2]:
```
df=pd.read_csv("college_majors.csv")
df
```

Out[2]:

	Rank	Major_code	Major	Total	Men	Women	Major_category	ShareWomen	Sa
0	1	2419	PETROLEUM ENGINEERING	2339.0	2057.0	282.0	Engineering	0.1206	
1	2	2416	MINING AND MINERAL ENGINEERING	756.0	679.0	77.0	Engineering	0.1019	
2	3	2415	METALLURGICAL ENGINEERING	856.0	725.0	131.0	Engineering	0.1530	
3	4	2417	NAVAL ARCHITECTURE AND MARINE ENGINEERING	1258.0	1123.0	135.0	Engineering	0.1073	
4	5	2405	CHEMICAL ENGINEERING	32260.0	21239.0	11021.0	Engineering	0.3416	
...	...	...	...	...	...	...	...	...	
168	169	3609	ZOOLOGY	8409.0	3050.0	5359.0	Biology & Life Science	0.6373	
169	170	5201	EDUCATIONAL PSYCHOLOGY	2854.0	522.0	2332.0	Psychology & Social Work	0.8171	
170	171	5202	CLINICAL PSYCHOLOGY	2838.0	568.0	2270.0	Psychology & Social Work	0.7999	
171	172	5203	COUNSELING PSYCHOLOGY	4626.0	931.0	3695.0	Psychology & Social Work	0.7987	
172	173	3501	LIBRARY SCIENCE	1098.0	134.0	964.0	Education	0.8780	

173 rows × 21 columns

In [3]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 173 entries, 0 to 172
Data columns (total 21 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   Rank                  173 non-null    int64
 1   Major_code            173 non-null    int64
 2   Major                 173 non-null    object
 3   Total                 172 non-null    float64
 4   Men                   172 non-null    float64
 5   Women                 172 non-null    float64
 6   Major_category        173 non-null    object
 7   ShareWomen            172 non-null    float64
 8   Sample_size           173 non-null    int64
 9   Employed              173 non-null    int64
 10  Full_time             173 non-null    int64
 11  Part_time             173 non-null    int64
 12  Full_time_year_round  173 non-null    int64
 13  Unemployed            173 non-null    int64
 14  Unemployment_rate     173 non-null    float64
 15  Median                173 non-null    int64
 16  P25th                 173 non-null    int64
 17  P75th                 173 non-null    int64
 18  College_jobs          173 non-null    int64
 19  Non_college_jobs      173 non-null    int64
 20  Low_wage_jobs         173 non-null    int64
dtypes: float64(5), int64(14), object(2)
memory usage: 28.5+ KB
```

In [4]:
```
df.describe()
```

Out[4]:

	Rank	Major_code	Total	Men	Women	ShareWomen	Sample_s
count	173.000000	173.000000	172.000000	172.000000	172.000000	172.000000	173.000(
mean	87.000000	3879.815029	39370.081395	16723.406977	22646.674419	0.522223	356.080(
std	50.084928	1687.753140	63483.491009	28122.433474	41057.330740	0.231203	618.361(
min	1.000000	1100.000000	124.000000	119.000000	0.000000	0.000000	2.000(
25%	44.000000	2403.000000	4549.750000	2177.500000	1778.250000	0.336050	39.000(
50%	87.000000	3608.000000	15104.000000	5434.000000	8386.500000	0.534000	130.000(
75%	130.000000	5503.000000	38909.750000	14631.000000	22553.750000	0.703275	338.000(
max	173.000000	6403.000000	393735.000000	173809.000000	307087.000000	0.969000	4212.000(

In [5]:
```
df.shape
```

Out[5]:
```
(173, 21)
```

In [6]:
```
df.head()
```

Out[6]:

	Rank	Major_code	Major	Total	Men	Women	Major_category	ShareWomen	Sam
0	1	2419	PETROLEUM ENGINEERING	2339.0	2057.0	282.0	Engineering	0.1206	
1	2	2416	MINING AND MINERAL ENGINEERING	756.0	679.0	77.0	Engineering	0.1019	
2	3	2415	METALLURGICAL ENGINEERING	856.0	725.0	131.0	Engineering	0.1530	
3	4	2417	NAVAL ARCHITECTURE AND MARINE ENGINEERING	1258.0	1123.0	135.0	Engineering	0.1073	
4	5	2405	CHEMICAL ENGINEERING	32260.0	21239.0	11021.0	Engineering	0.3416	

5 rows × 21 columns

In [7]:
```
df.ndim
```

Out[7]: 2

In [8]:
```
df["Total"].fillna(df["Total"].mean(),inplace=True)
```

In [9]:
```
df["Women"].fillna(df["Women"].mean(),inplace=True)
```

In [10]:
```
df["Men"].fillna(df["Men"].mean(),inplace=True)
```

In [11]:
```
df["ShareWomen"].fillna(df["ShareWomen"].mean(),inplace=True)
```

In [12]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 173 entries, 0 to 172
Data columns (total 21 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   Rank                  173 non-null    int64
 1   Major_code            173 non-null    int64
 2   Major                 173 non-null    object
 3   Total                 173 non-null    float64
 4   Men                   173 non-null    float64
 5   Women                 173 non-null    float64
 6   Major_category        173 non-null    object
 7   ShareWomen            173 non-null    float64
 8   Sample_size           173 non-null    int64
 9   Employed              173 non-null    int64
 10  Full_time             173 non-null    int64
 11  Part_time             173 non-null    int64
 12  Full_time_year_round  173 non-null    int64
 13  Unemployed            173 non-null    int64
 14  Unemployment_rate     173 non-null    float64
 15  Median                173 non-null    int64
 16  P25th                 173 non-null    int64
 17  P75th                 173 non-null    int64
 18  College_jobs          173 non-null    int64
 19  Non_college_jobs      173 non-null    int64
 20  Low_wage_jobs         173 non-null    int64
dtypes: float64(5), int64(14), object(2)
memory usage: 28.5+ KB
```

In [13]:
```
df.tail()
```

Out[13]:

	Rank	Major_code	Major	Total	Men	Women	Major_category	ShareWomen	Sampl
168	169	3609	ZOOLOGY	8409.0	3050.0	5359.0	Biology & Life Science	0.6373	
169	170	5201	EDUCATIONAL PSYCHOLOGY	2854.0	522.0	2332.0	Psychology & Social Work	0.8171	
170	171	5202	CLINICAL PSYCHOLOGY	2838.0	568.0	2270.0	Psychology & Social Work	0.7999	
171	172	5203	COUNSELING PSYCHOLOGY	4626.0	931.0	3695.0	Psychology & Social Work	0.7987	
172	173	3501	LIBRARY SCIENCE	1098.0	134.0	964.0	Education	0.8780	

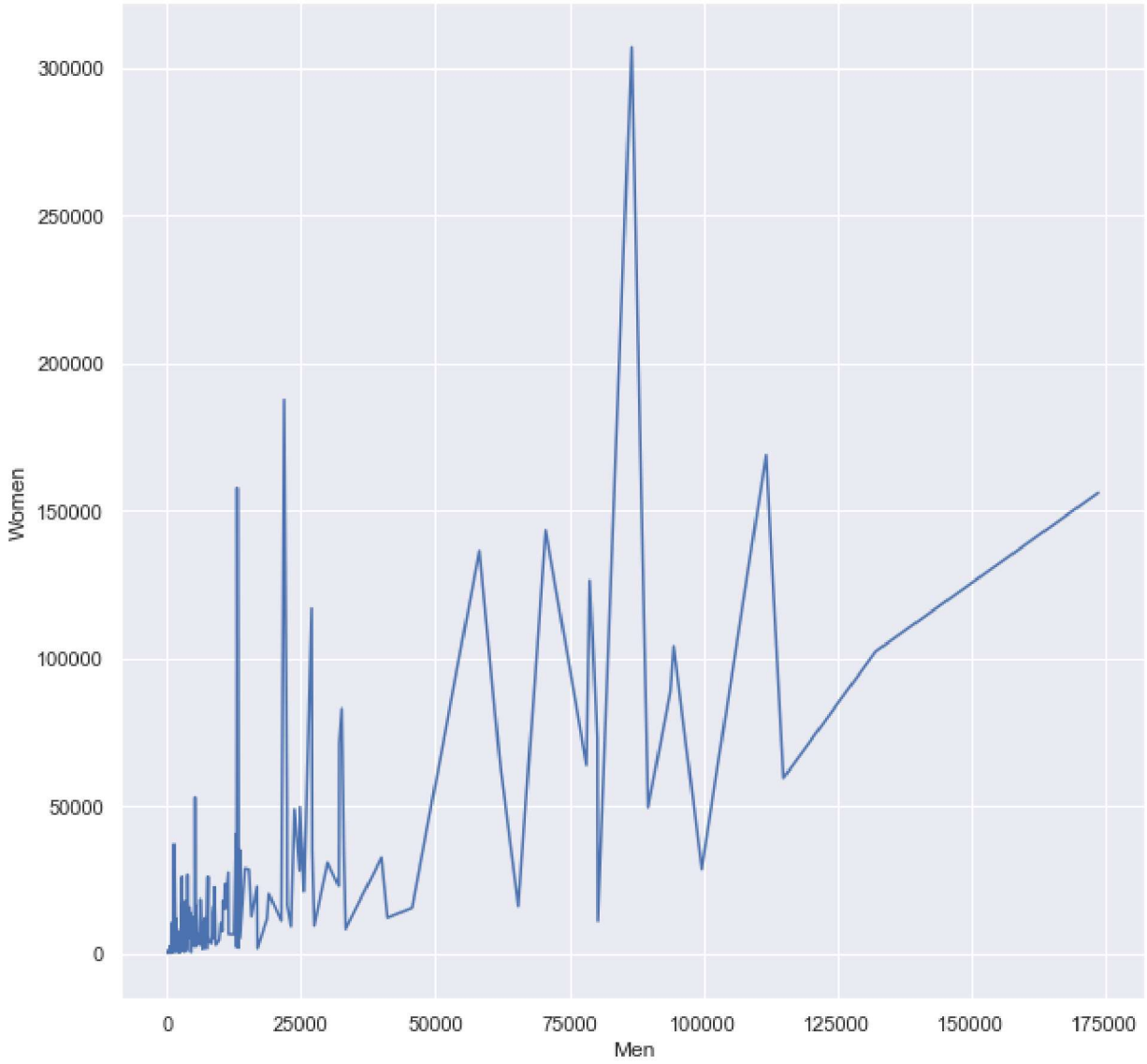
5 rows × 21 columns

In [14]:
```
df.describe()
```

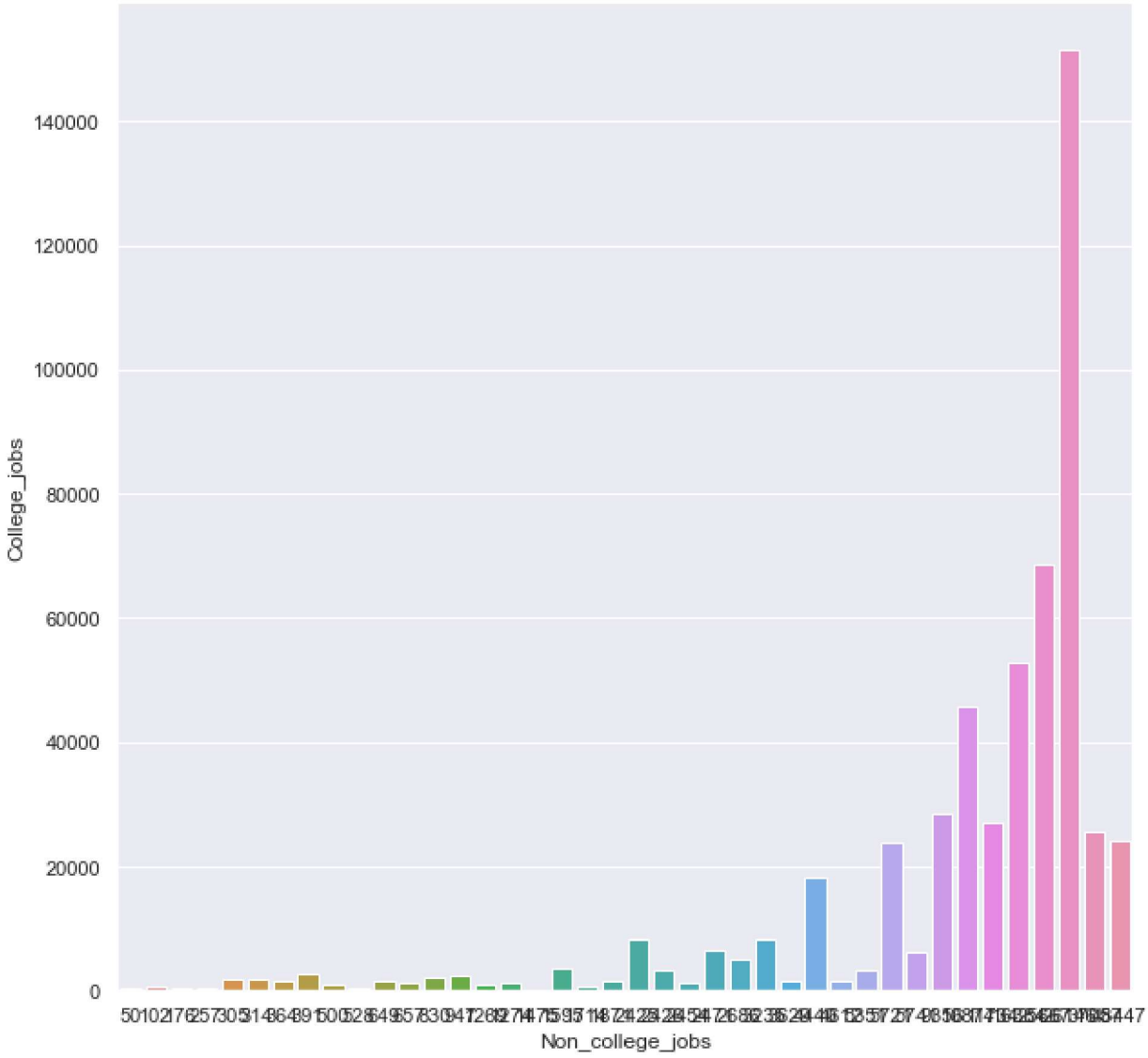
Out[14]:

	Rank	Major_code	Total	Men	Women	ShareWomen	Sample_s
count	173.000000	173.000000	173.000000	173.000000	173.000000	173.000000	173.000(
mean	87.000000	3879.815029	39370.081395	16723.406977	22646.674419	0.522223	356.080(
std	50.084928	1687.753140	63298.676961	28040.563043	40937.804050	0.230530	618.361(
min	1.000000	1100.000000	124.000000	119.000000	0.000000	0.000000	2.000(
25%	44.000000	2403.000000	4626.000000	2200.000000	1784.000000	0.339700	39.000(
50%	87.000000	3608.000000	15150.000000	5521.000000	8489.000000	0.532300	130.000(
75%	130.000000	5503.000000	39107.000000	15204.000000	22646.674419	0.702000	338.000(
max	173.000000	6403.000000	393735.000000	173809.000000	307087.000000	0.969000	4212.000(

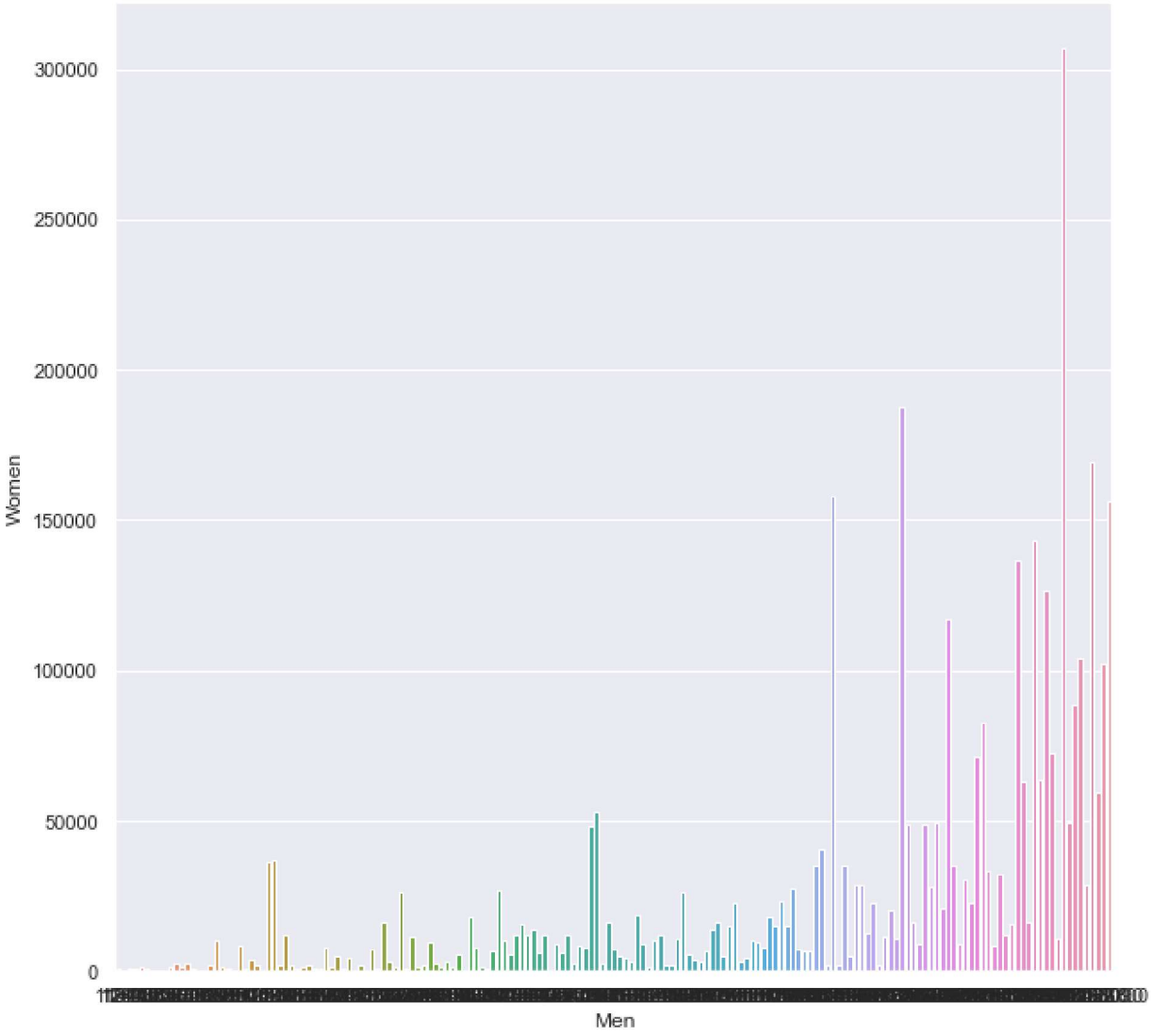
In [15]:
```
sns.lineplot(x="Men",y="Women",data=df[:500]);
```



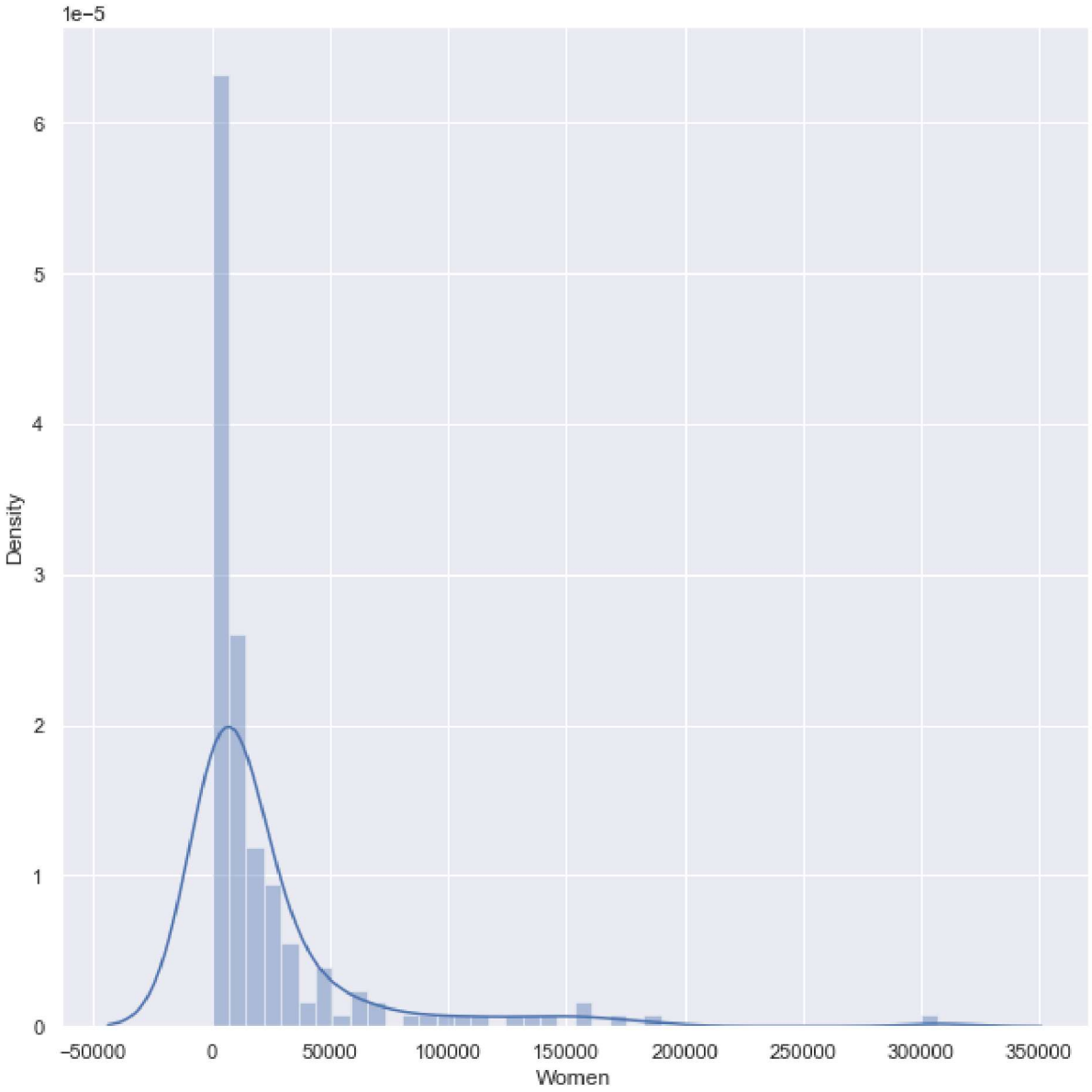
In [16]:
```
sns.barplot(x="Non_college_jobs",y="College_jobs",data=df[:40]);
```



In [17]:
```
sns.barplot(x="Men",y="Women",data=df[:500]);
```

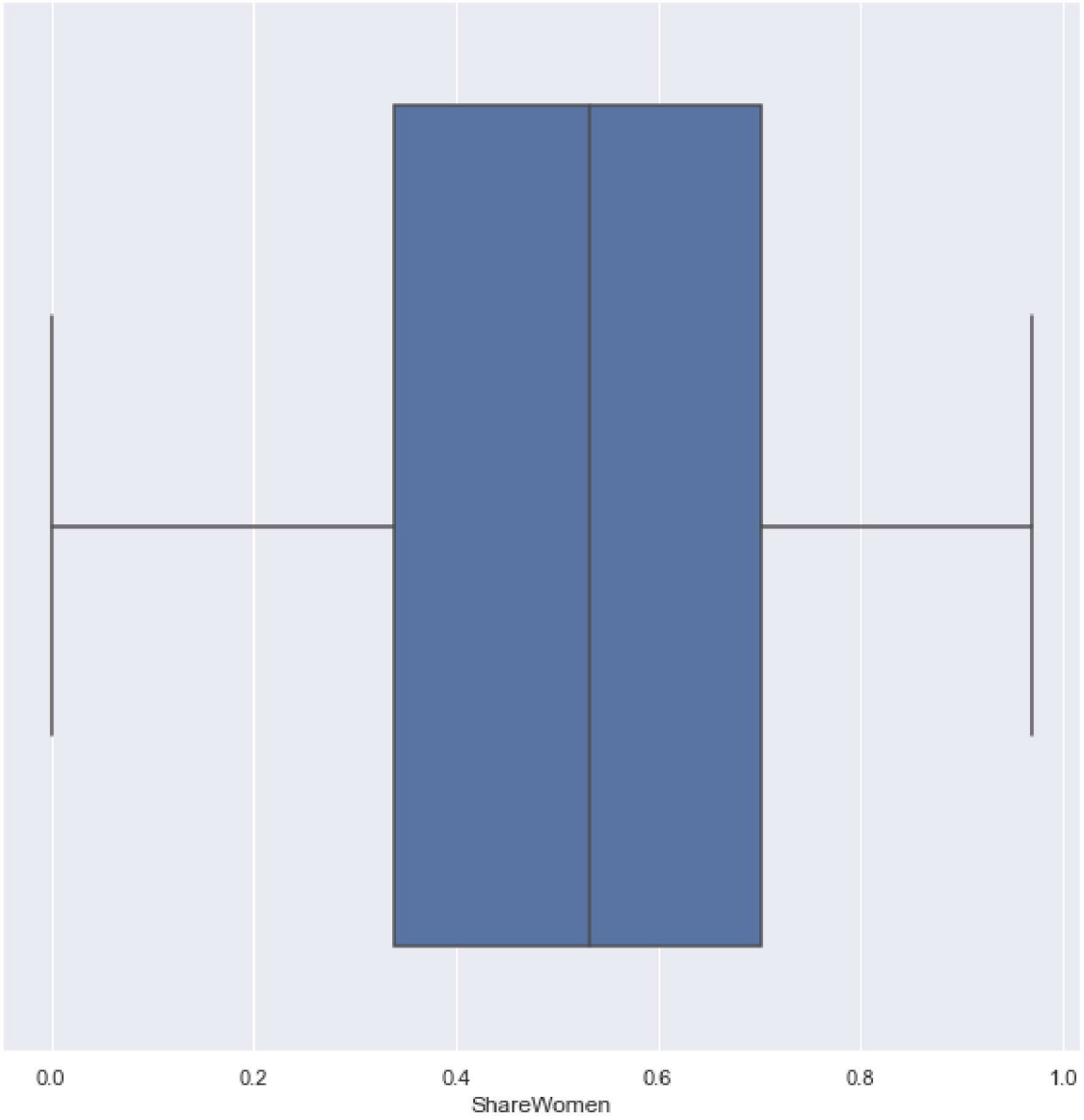


In [18]:
```
sns.distplot(df["Women"]);
```



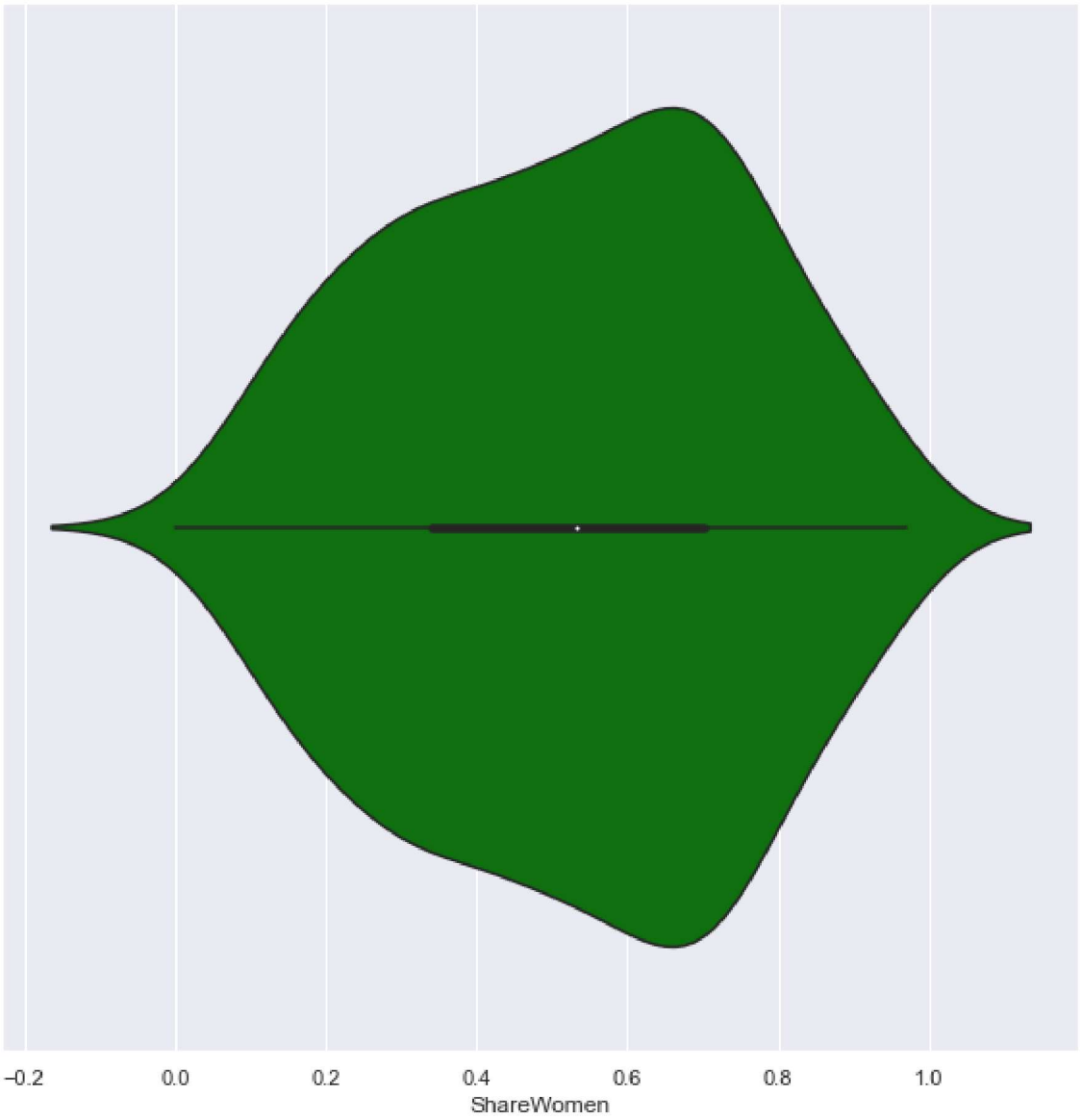
In [19]:
```
sns.boxplot(df["ShareWomen"],orient="verticle")
```

Out[19]:
```
<AxesSubplot:xlabel='ShareWomen'>
```

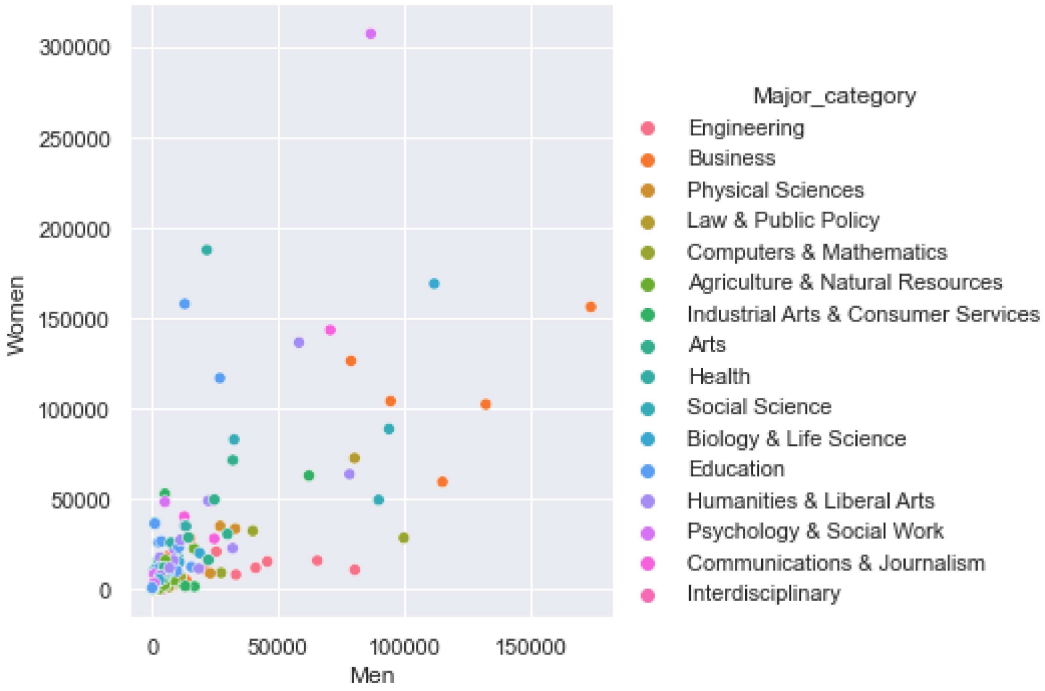


In [20]:
```
sns.violinplot(df["ShareWomen"],orient="verticle",color="green")
```

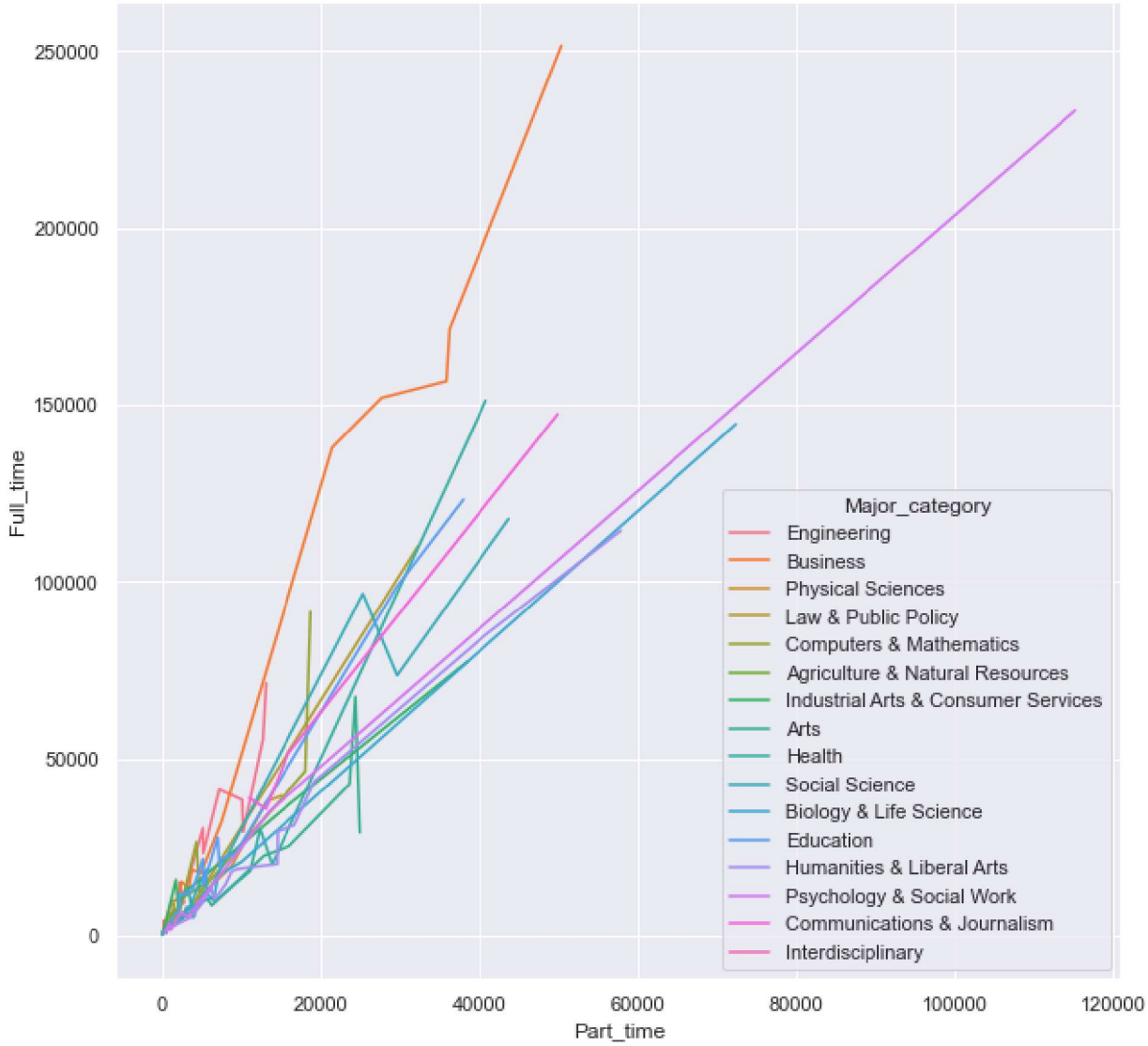
Out[20]:
```
<AxesSubplot:xlabel='ShareWomen'>
```



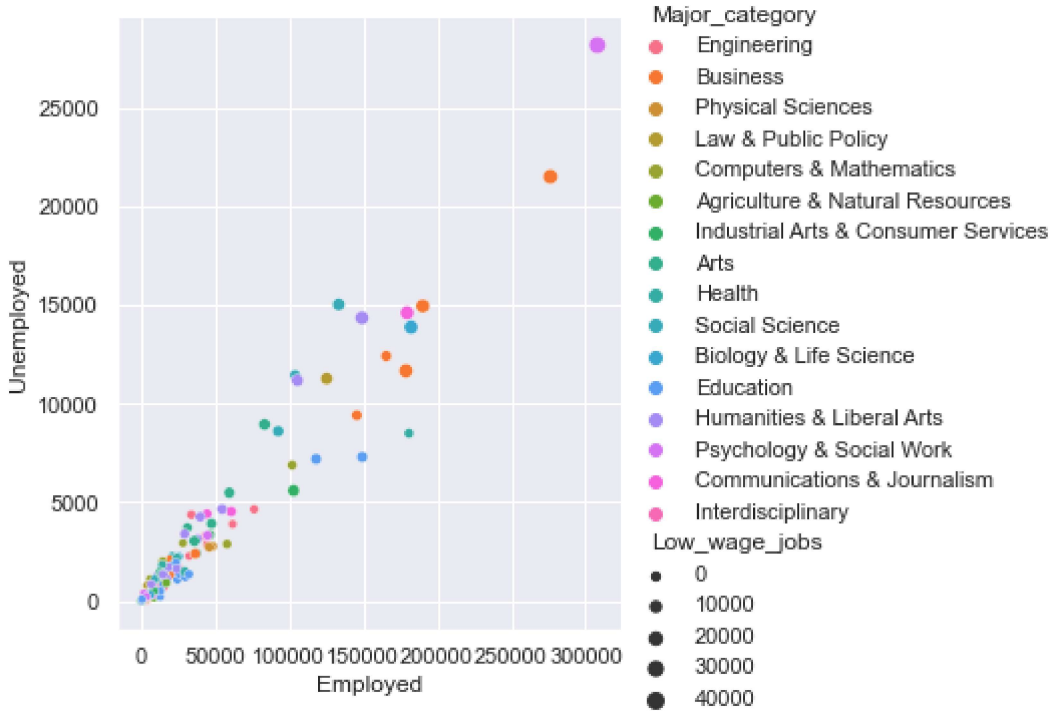
In [21]:
```
sns.relplot(x="Men",y="Women",hue="Major_category",data=df[:200],kind="scatter");
```



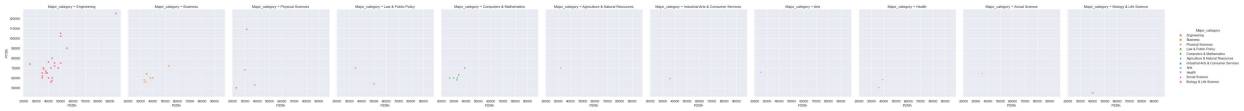
In [22]:
```
sns.lineplot(x="Part_time",y="Full_time",hue="Major_category",data=df[:500]);
```



In [23]:
```
sns.relplot(x="Employed",y="Unemployed",data=df[:200],kind="scatter",hue="Major_categ
```



In [33]:
```
sns.relplot(x="P25th",y="P75th",hue="Major_category",style="Major_category",col="Major
```



In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: