In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from matplotlib import pyplot as plt
import seaborn as sns
from scipy.stats import norm
import scipy
from scipy import stats
import statsmodels.formula.api as smf
from scipy.stats import linregress
from scipy.stats import uniform
from statsmodels.stats.proportion import proportions_ztest
from scipy.stats import binom
import pingouin
from scipy.stats import chisquare
df = pd.read_csv(r"C:\Users\olale\Dropbox\My PC (DESKTOP-04PLJ90)\Desktop\Code school\employees.csv")
```

In [ ]:
```
#importing the dataframe
```

In [2]:
```
dff = pd.DataFrame(df)
```

In [ ]:
```
#checking dataframe's
```

In [3]:
```
dff.head()
```

Out[3]:

	First Name	Gender	Start Date	Last Login Time	Salary	Bonus %	Senior Management	Team
0	Douglas	Male	8/6/1993	12:42 PM	97308	6.945	True	Marketing
1	Thomas	Male	3/31/1996	6:53 AM	61933	4.170	True	NaN
2	Maria	Female	4/23/1993	11:17 AM	130590	11.858	False	Finance
3	Jerry	Male	3/4/2005	1:00 PM	138705	9.340	True	Finance
4	Larry	Male	1/24/1998	4:47 PM	101004	1.389	True	Client Services
...	...	...	...	...	...	...	...	...
995	Henry	NaN	11/23/2014	6:09 AM	132483	16.655	False	Distribution
996	Phillip	Male	1/31/1984	6:30 AM	42392	19.675	False	Finance
997	Russell	Male	5/20/2013	12:39 PM	96914	1.421	False	Product
998	Larry	Male	4/20/2013	4:45 PM	60500	11.985	False	Business Development
999	Albert	Male	5/15/2012	6:24 PM	129949	10.169	True	Sales

1000 rows × 8 columns

In [4]:
```
dff.dropna(inplace=True)
```

In [5]:
```
dff["start_date"] =pd.to_datetime(dff["Start Date"])
```

In [6]:
```
dff.drop("Start Date",axis=1, inplace=True)
```

In [7]:
```
dff
```

Out[7]:

	First Name	Gender	Last Login Time	Salary	Bonus %	Senior Management	Team	start_date
0	Douglas	Male	12:42 PM	97308	6.945	True	Marketing	1993-08-06
2	Maria	Female	11:17 AM	130590	11.858	False	Finance	1993-04-23
3	Jerry	Male	1:00 PM	138705	9.340	True	Finance	2005-03-04
4	Larry	Male	4:47 PM	101004	1.389	True	Client Services	1998-01-24
5	Dennis	Male	1:35 AM	115163	10.125	False	Legal	1987-04-18
...	...	...	...	...	...	...	...	...
994	George	Male	5:47 PM	98874	4.479	True	Marketing	2013-06-21
996	Phillip	Male	6:30 AM	42392	19.675	False	Finance	1984-01-31
997	Russell	Male	12:39 PM	96914	1.421	False	Product	2013-05-20
998	Larry	Male	4:45 PM	60500	11.985	False	Business Development	2013-04-20

999	Albert	Male	6:24 PM	129949	10.169	True	Sales	2012-05-15

764 rows × 8 columns

In [8]:
```
#convert the start date to separate month and year columns
dff["Year"] = dff["start_date"].dt.year
dff["Month"] = dff["start_date"].dt.month
dff["Day"] = dff["start_date"].dt.day
```

In [9]:
```
dff
```

Out[9]:

	First Name	Gender	Last Login Time	Salary	Bonus %	Senior Management	Team	start_date	Year	Month	Day
0	Douglas	Male	12:42 PM	97308	6.945	True	Marketing	1993-08-06	1993	8	6
2	Maria	Female	11:17 AM	130590	11.858	False	Finance	1993-04-23	1993	4	23
3	Jerry	Male	1:00 PM	138705	9.340	True	Finance	2005-03-04	2005	3	4
4	Larry	Male	4:47 PM	101004	1.389	True	Client Services	1998-01-24	1998	1	24
5	Dennis	Male	1:35 AM	115163	10.125	False	Legal	1987-04-18	1987	4	18
...	...	...	...	...	...	...	...	...	...	...	...
994	George	Male	5:47 PM	98874	4.479	True	Marketing	2013-06-21	2013	6	21
996	Phillip	Male	6:30 AM	42392	19.675	False	Finance	1984-01-31	1984	1	31
997	Russell	Male	12:39 PM	96914	1.421	False	Product	2013-05-20	2013	5	20
998	Larry	Male	4:45 PM	60500	11.985	False	Business Development	2013-04-20	2013	4	20
999	Albert	Male	6:24 PM	129949	10.169	True	Sales	2012-05-15	2012	5	15

764 rows × 11 columns

In [10]:
```
dff.Year.min()
```

Out[10]:
```
1980
```

In [11]:
```
dff.Year.max()
```

Out[11]:
```
2016
```

In [12]:
```
dff.Year.mode()
```

Out[12]:
```
0    2009
dtype: int64
```

In [13]:
```
#All folks recruited in 2009
recruits_2009 = dff[dff["Year"] ==2009]
```

In [14]:
```
recruits_2009.sort_values(by=["start_date", "Team"], ascending= [False, False])
```

Out[14]:

	First Name	Gender	Last Login Time	Salary	Bonus %	Senior Management	Team	start_date	Year	Month	Day
908	Janice	Female	6:42 AM	102697	3.283	False	Engineering	2009-12-17	2009	12	17
887	David	Male	8:48 AM	92242	15.407	False	Legal	2009-12-05	2009	12	5
652	Willie	Male	5:39 AM	141932	1.017	True	Engineering	2009-12-05	2009	12	5
560	Shawn	Male	10:24 AM	96610	2.097	True	Client Services	2009-12-03	2009	12	3
46	Bruce	Male	10:47 PM	114796	6.796	False	Finance	2009-11-28	2009	11	28
65	Steve	Male	11:44 PM	61310	12.428	True	Distribution	2009-11-11	2009	11	11
362	Joshua	Male	6:32 AM	72893	9.555	False	Distribution	2009-11-09	2009	11	9
71	Johnny	Male	4:23 PM	118172	16.194	True	Sales	2009-11-06	2009	11	6
212	Lisa	Female	9:42 AM	115387	1.821	False	Client Services	2009-11-02	2009	11	2

675	Diane	Female	6:56 PM	130577	12.791	False	Marketing	2009-10-30	2009	10	30
369	Mary	Female	6:32 PM	87721	12.484	False	Product	2009-10-18	2009	10	18
308	Cheryl	Female	10:16 AM	81308	2.196	True	Legal	2009-09-08	2009	9	8
637	Wayne	Male	1:37 AM	126956	18.396	False	Human Resources	2009-09-02	2009	9	2
450	Willie	Male	1:03 PM	55038	19.691	False	Legal	2009-08-22	2009	8	22
772	Lillian	Female	5:41 AM	113554	18.018	True	Business Development	2009-08-14	2009	8	14
370	Linda	Female	10:12 PM	144001	2.194	False	Business Development	2009-07-15	2009	7	15
780	Steven	Male	4:55 AM	110306	16.843	True	Human Resources	2009-07-05	2009	7	5
113	Tina	Female	7:16 AM	114767	3.711	True	Engineering	2009-06-12	2009	6	12
215	Mary	Female	11:41 PM	92544	3.800	False	Client Services	2009-05-30	2009	5	30
879	Amy	Female	6:26 AM	75415	19.132	False	Client Services	2009-05-20	2009	5	20
447	Gregory	Male	3:52 PM	142208	11.204	True	Engineering	2009-05-15	2009	5	15
340	Steven	Male	2:14 PM	113060	2.846	True	Sales	2009-05-12	2009	5	12
344	Scott	Male	4:36 AM	58248	3.914	False	Business Development	2009-05-12	2009	5	12
730	Nicole	Female	12:40 AM	66047	18.674	True	Marketing	2009-04-26	2009	4	26
145	Jennifer	Female	10:47 PM	71715	13.079	True	Client Services	2009-04-04	2009	4	4
919	Sean	Male	11:38 AM	131423	8.957	False	Distribution	2009-03-21	2009	3	21
255	Denise	Female	7:57 AM	115118	5.108	False	Human Resources	2009-03-20	2009	3	20
283	Todd	Male	3:43 AM	107281	1.612	True	Engineering	2009-03-11	2009	3	11
700	Frank	Male	9:15 PM	78891	7.927	True	Distribution	2009-03-02	2009	3	2
822	Deborah	Female	10:17 AM	118043	7.266	True	Business Development	2009-02-26	2009	2	26
343	Ronald	Male	2:09 PM	96633	4.990	True	Engineering	2009-02-24	2009	2	24
519	Raymond	Male	10:38 PM	37812	3.178	False	Human Resources	2009-02-16	2009	2	16
36	Rachel	Female	8:47 PM	142032	12.599	False	Business Development	2009-02-16	2009	2	16

In [15]:
```
recruits_2009[recruits_2009["Team"]=="Marketing"]
```

Out[15]:

	First Name	Gender	Last Login Time	Salary	Bonus %	Senior Management	Team	start_date	Year	Month	Day
675	Diane	Female	6:56 PM	130577	12.791	False	Marketing	2009-10-30	2009	10	30
730	Nicole	Female	12:40 AM	66047	18.674	True	Marketing	2009-04-26	2009	4	26

In [16]:
```
recruits_2009.groupby('Team').Salary.agg(['mean', 'median'])
```

Out[16]:

Team	mean	median
Business Development	115175.600000	118043
Client Services	90334.200000	92544
Distribution	86129.250000	75892
Engineering	117586.333333	111024
Finance	114796.000000	114796
Human Resources	97548.000000	112712
Legal	76196.000000	81308
Marketing	98312.000000	98312
Product	87721.000000	87721
Sales	115616.000000	115616

In [17]:
```
#in what year did ditribution dpt. has its highest employees and how many are they
```

In [18]:
```
dist_recruitment = dff[dff["Team"] == "Distribution"]
```

In [19]:
```
dist_recruitment
```

Out[19]:

	First Name	Gender	Last Login Time	Salary	Bonus %	Senior Management	Team	start_date	Year	Month	Day
40	Michael	Male	11:25 AM	99283	2.665	True	Distribution	2008-10-10	2008	10	10
65	Steve	Male	11:44 PM	61310	12.428	True	Distribution	2009-11-11	2009	11	11
76	Margaret	Female	12:42 PM	131604	7.353	True	Distribution	1988-09-10	1988	9	10
137	Adam	Male	1:45 AM	95327	15.120	False	Distribution	2011-05-21	2011	5	21
177	Wayne	Male	8:00 AM	102652	14.085	True	Distribution	2012-04-07	2012	4	7
181	Randy	Male	12:12 PM	58129	1.952	True	Distribution	1999-11-14	1999	11	14
194	Irene	Female	8:25 PM	131038	8.996	False	Distribution	2004-01-30	2004	1	30
224	Sarah	Female	7:50 AM	87298	2.311	False	Distribution	1995-09-14	1995	9	14
229	Jeremy	Male	12:15 AM	49542	1.679	True	Distribution	2000-06-08	2000	6	8
248	Justin	Male	5:58 PM	82782	4.366	True	Distribution	1992-12-06	1992	12	6
260	Gloria	Female	1:44 AM	90730	2.491	False	Distribution	2007-03-27	2007	3	27
278	Betty	Female	6:03 PM	51613	12.984	False	Distribution	2005-06-28	2005	6	28
294	Virginia	Female	6:23 AM	46905	19.154	False	Distribution	1999-10-20	1999	10	20
307	Marilyn	Female	2:23 AM	86386	2.937	False	Distribution	1981-09-26	1981	9	26
356	Judy	Female	3:32 PM	38092	5.668	False	Distribution	1990-02-01	1990	2	1
362	Joshua	Male	6:32 AM	72893	9.555	False	Distribution	2009-11-09	2009	11	9
397	Clarence	Male	9:00 AM	116693	13.835	True	Distribution	2005-01-13	2005	1	13
430	Andrea	Female	11:54 AM	79123	19.422	False	Distribution	2010-10-01	2010	10	1
435	Billy	Male	3:32 PM	144709	10.069	True	Distribution	2006-12-01	2006	12	1
470	Ryan	Male	10:18 PM	139917	11.466	False	Distribution	1993-07-20	1993	7	20
486	Howard	Male	6:36 AM	37984	2.021	False	Distribution	2012-04-09	2012	4	9
501	Sean	Male	7:07 PM	42748	9.765	False	Distribution	2013-02-11	2013	2	11
522	Catherine	Female	7:24 PM	58047	14.858	True	Distribution	2013-08-31	2013	8	31
530	Kathleen	Female	9:16 AM	35575	14.595	False	Distribution	2014-06-13	2014	6	13
542	Amanda	Female	1:32 PM	80803	14.077	True	Distribution	2004-08-01	2004	8	1
557	Jane	Female	8:39 AM	42424	18.115	False	Distribution	1994-06-01	1994	6	1
566	Johnny	Male	1:35 PM	91124	12.986	True	Distribution	1995-01-08	1995	1	8
591	Rachel	Female	12:01 PM	110924	7.808	False	Distribution	1988-04-22	1988	4	22
597	Teresa	Female	7:37 PM	69740	8.294	False	Distribution	1987-06-24	1987	6	24
614	Eric	Male	9:16 PM	65168	11.513	False	Distribution	2004-11-12	2004	11	12
616	Kimberly	Female	2:23 PM	37916	12.929	True	Distribution	1986-12-06	1986	12	6
640	Kathleen	Female	10:49 AM	42553	3.756	True	Distribution	2004-08-28	2004	8	28
662	Katherine	Female	1:40 AM	41643	4.659	True	Distribution	2000-12-19	2000	12	19
663	Andrea	Female	11:10 PM	113760	12.866	True	Distribution	1989-08-31	1989	8	31
665	Anthony	Male	1:35 PM	146141	3.645	True	Distribution	2013-02-13	2013	2	13
671	Laura	Female	6:07 PM	84672	3.960	False	Distribution	2000-07-12	2000	7	12
700	Frank	Male	9:15 PM	78891	7.927	True	Distribution	2009-03-02	2009	3	2
704	Thomas	Male	9:51 AM	65251	11.211	False	Distribution	1991-09-07	1991	9	7
713	Ann	Female	5:44 AM	79796	9.851	False	Distribution	1994-09-28	1994	9	28
714	Jonathan	Male	6:02 PM	83809	12.922	False	Distribution	1984-07-30	1984	7	30
756	Stephen	Male	6:26 AM	121816	10.615	True	Distribution	1984-10-21	1984	10	21
759	Ruth	Female	6:52 AM	59678	10.895	False	Distribution	1980-09-02	1980	9	2
762	Terry	Male	4:33 AM	35633	3.947	True	Distribution	2004-11-10	2004	11	10
778	Antonio	Male	4:17 AM	137979	5.266	False	Distribution	2003-12-28	2003	12	28
793	Andrea	Female	9:25 AM	149105	13.707	True	Distribution	1999-07-22	1999	7	22
804	Shawn	Male	2:12 PM	39335	10.664	False	Distribution	2008-03-17	2008	3	17
814	Rachel	Female	12:52 AM	54941	3.221	True	Distribution	2011-06-23	2011	6	23
818	Ann	Female	1:08 AM	96941	10.048	True	Distribution	1980-10-03	1980	10	3
830	Michael	Male	1:20 AM	81206	19.908	True	Distribution	2002-08-31	2002	8	31
838	Billy	Male	3:14 PM	115280	9.153	False	Distribution	2000-04-06	2000	4	6
840	Lillian	Female	8:53 AM	103854	4.924	True	Distribution	2002-08-26	2002	8	26
878	Jacqueline	Female	7:13 PM	125418	8.064	False	Distribution	2003-05-25	2003	5	25
901	Patricia	Female	4:52 PM	119266	6.911	False	Distribution	1995-10-10	1995	10	10
914	Ann	Female	6:37 AM	71958	5.272	True	Distribution	2001-09-28	2001	9	28

919	Sean	Male	11:38 AM	131423	8.957	False	Distribution	2009-03-21	2009	3	21
926	Judith	Female	7:32 AM	109324	19.488	False	Distribution	2004-03-01	2004	3	1
931	Harold	Male	12:40 PM	140444	3.771	False	Distribution	2012-06-23	2012	6	23
940	Andrew	Male	9:38 AM	137386	8.611	True	Distribution	1990-09-28	1990	9	28
944	Kenneth	Male	8:24 AM	101914	1.905	True	Distribution	2006-05-10	2006	5	10
968	Louise	Female	10:27 PM	43050	11.671	False	Distribution	1995-03-27	1995	3	27

In [20]:
```
dist = dist_recruitment.groupby("Year").count()
```

In [21]:
```
dist.Team.max()
```

Out[21]:
6

In [22]:
```
dist
```

Out[22]:

Year	First Name	Gender	Last Login Time	Salary	Bonus %	Senior Management	Team	start_date	Month	Day
1980	2	2	2	2	2	2	2	2	2	2
1981	1	1	1	1	1	1	1	1	1	1
1984	2	2	2	2	2	2	2	2	2	2
1986	1	1	1	1	1	1	1	1	1	1
1987	1	1	1	1	1	1	1	1	1	1
1988	2	2	2	2	2	2	2	2	2	2
1989	1	1	1	1	1	1	1	1	1	1
1990	2	2	2	2	2	2	2	2	2	2
1991	1	1	1	1	1	1	1	1	1	1
1992	1	1	1	1	1	1	1	1	1	1
1993	1	1	1	1	1	1	1	1	1	1
1994	2	2	2	2	2	2	2	2	2	2
1995	4	4	4	4	4	4	4	4	4	4
1999	3	3	3	3	3	3	3	3	3	3
2000	4	4	4	4	4	4	4	4	4	4
2001	1	1	1	1	1	1	1	1	1	1
2002	2	2	2	2	2	2	2	2	2	2
2003	2	2	2	2	2	2	2	2	2	2
2004	6	6	6	6	6	6	6	6	6	6
2005	2	2	2	2	2	2	2	2	2	2
2006	2	2	2	2	2	2	2	2	2	2
2007	1	1	1	1	1	1	1	1	1	1
2008	2	2	2	2	2	2	2	2	2	2
2009	4	4	4	4	4	4	4	4	4	4
2010	1	1	1	1	1	1	1	1	1	1
2011	2	2	2	2	2	2	2	2	2	2
2012	3	3	3	3	3	3	3	3	3	3
2013	3	3	3	3	3	3	3	3	3	3
2014	1	1	1	1	1	1	1	1	1	1

In [23]:
```
#What month is Senior management most active
```

In [24]:
```
dff.head(3)
```

Out[24]:

	First Name	Gender	Last Login Time	Salary	Bonus %	Senior Management	Team	start_date	Year	Month	Day

0	Douglas	Male	12:42 PM	97308	6.945	True	Marketing	1993-08-06	1993	8	6
2	Maria	Female	11:17 AM	130590	11.858	False	Finance	1993-04-23	1993	4	23
3	Jerry	Male	1:00 PM	138705	9.340	True	Finance	2005-03-04	2005	3	4

In [25]:
```
#most_active_month =
```

In [26]:
```
year_max = dff["Month"].max()
```

In [27]:
```
year_max
```

Out[27]:
12

In [28]:
```
df_sr_mana = dff[dff["Senior Management"] == True]
```

In [29]:
```
df_sr_mana
```

Out[29]:

	First Name	Gender	Last Login Time	Salary	Bonus %	Senior Management	Team	start_date	Year	Month	Day
0	Douglas	Male	12:42 PM	97308	6.945	True	Marketing	1993-08-06	1993	8	6
3	Jerry	Male	1:00 PM	138705	9.340	True	Finance	2005-03-04	2005	3	4
4	Larry	Male	4:47 PM	101004	1.389	True	Client Services	1998-01-24	1998	1	24
6	Ruby	Female	4:20 PM	65476	10.012	True	Product	1987-08-17	1987	8	17
8	Angela	Female	6:29 AM	95570	18.523	True	Engineering	2005-11-22	2005	11	22
...	...	...	...	...	...	...	...	...	...	...	...
991	Rose	Female	5:12 AM	134505	11.051	True	Marketing	2002-08-25	2002	8	25
992	Anthony	Male	8:35 AM	112769	11.625	True	Finance	2011-10-16	2011	10	16
993	Tina	Female	3:53 PM	56450	19.040	True	Engineering	1997-05-15	1997	5	15
994	George	Male	5:47 PM	98874	4.479	True	Marketing	2013-06-21	2013	6	21
999	Albert	Male	6:24 PM	129949	10.169	True	Sales	2012-05-15	2012	5	15

381 rows × 11 columns

In [30]:
```
#df_sr_mana["Year"].max()
```

In [31]:
```
#s = df_sr_mana[df_sr_mana["Year"] == "2016"]
```

In [32]:
```
df_sr_mana
```

Out[32]:

	First Name	Gender	Last Login Time	Salary	Bonus %	Senior Management	Team	start_date	Year	Month	Day
0	Douglas	Male	12:42 PM	97308	6.945	True	Marketing	1993-08-06	1993	8	6
3	Jerry	Male	1:00 PM	138705	9.340	True	Finance	2005-03-04	2005	3	4
4	Larry	Male	4:47 PM	101004	1.389	True	Client Services	1998-01-24	1998	1	24
6	Ruby	Female	4:20 PM	65476	10.012	True	Product	1987-08-17	1987	8	17
8	Angela	Female	6:29 AM	95570	18.523	True	Engineering	2005-11-22	2005	11	22
...	...	...	...	...	...	...	...	...	...	...	...
991	Rose	Female	5:12 AM	134505	11.051	True	Marketing	2002-08-25	2002	8	25
992	Anthony	Male	8:35 AM	112769	11.625	True	Finance	2011-10-16	2011	10	16
993	Tina	Female	3:53 PM	56450	19.040	True	Engineering	1997-05-15	1997	5	15
994	George	Male	5:47 PM	98874	4.479	True	Marketing	2013-06-21	2013	6	21
999	Albert	Male	6:24 PM	129949	10.169	True	Sales	2012-05-15	2012	5	15

381 rows × 11 columns

In [33]:
```
df_sr_mana
```

Out[33]:

	First Name	Gender	Last Login Time	Salary	Bonus %	Senior Management	Team	start_date	Year	Month	Day
0	Douglas	Male	12:42 PM	97308	6.945	True	Marketing	1993-08-06	1993	8	6
3	Jerry	Male	1:00 PM	138705	9.340	True	Finance	2005-03-04	2005	3	4
4	Larry	Male	4:47 PM	101004	1.389	True	Client Services	1998-01-24	1998	1	24
6	Ruby	Female	4:20 PM	65476	10.012	True	Product	1987-08-17	1987	8	17
8	Angela	Female	6:29 AM	95570	18.523	True	Engineering	2005-11-22	2005	11	22
...	...	...	...	...	...	...	...	...	...	...	...
991	Rose	Female	5:12 AM	134505	11.051	True	Marketing	2002-08-25	2002	8	25
992	Anthony	Male	8:35 AM	112769	11.625	True	Finance	2011-10-16	2011	10	16
993	Tina	Female	3:53 PM	56450	19.040	True	Engineering	1997-05-15	1997	5	15
994	George	Male	5:47 PM	98874	4.479	True	Marketing	2013-06-21	2013	6	21
999	Albert	Male	6:24 PM	129949	10.169	True	Sales	2012-05-15	2012	5	15

381 rows × 11 columns

In [34]:
```
dff
```

Out[34]:

	First Name	Gender	Last Login Time	Salary	Bonus %	Senior Management	Team	start_date	Year	Month	Day
0	Douglas	Male	12:42 PM	97308	6.945	True	Marketing	1993-08-06	1993	8	6
2	Maria	Female	11:17 AM	130590	11.858	False	Finance	1993-04-23	1993	4	23
3	Jerry	Male	1:00 PM	138705	9.340	True	Finance	2005-03-04	2005	3	4
4	Larry	Male	4:47 PM	101004	1.389	True	Client Services	1998-01-24	1998	1	24
5	Dennis	Male	1:35 AM	115163	10.125	False	Legal	1987-04-18	1987	4	18
...	...	...	...	...	...	...	...	...	...	...	...
994	George	Male	5:47 PM	98874	4.479	True	Marketing	2013-06-21	2013	6	21
996	Phillip	Male	6:30 AM	42392	19.675	False	Finance	1984-01-31	1984	1	31
997	Russell	Male	12:39 PM	96914	1.421	False	Product	2013-05-20	2013	5	20
998	Larry	Male	4:45 PM	60500	11.985	False	Business Development	2013-04-20	2013	4	20
999	Albert	Male	6:24 PM	129949	10.169	True	Sales	2012-05-15	2012	5	15

764 rows × 11 columns

In [35]:
```
dff.duplicated().sum()
```

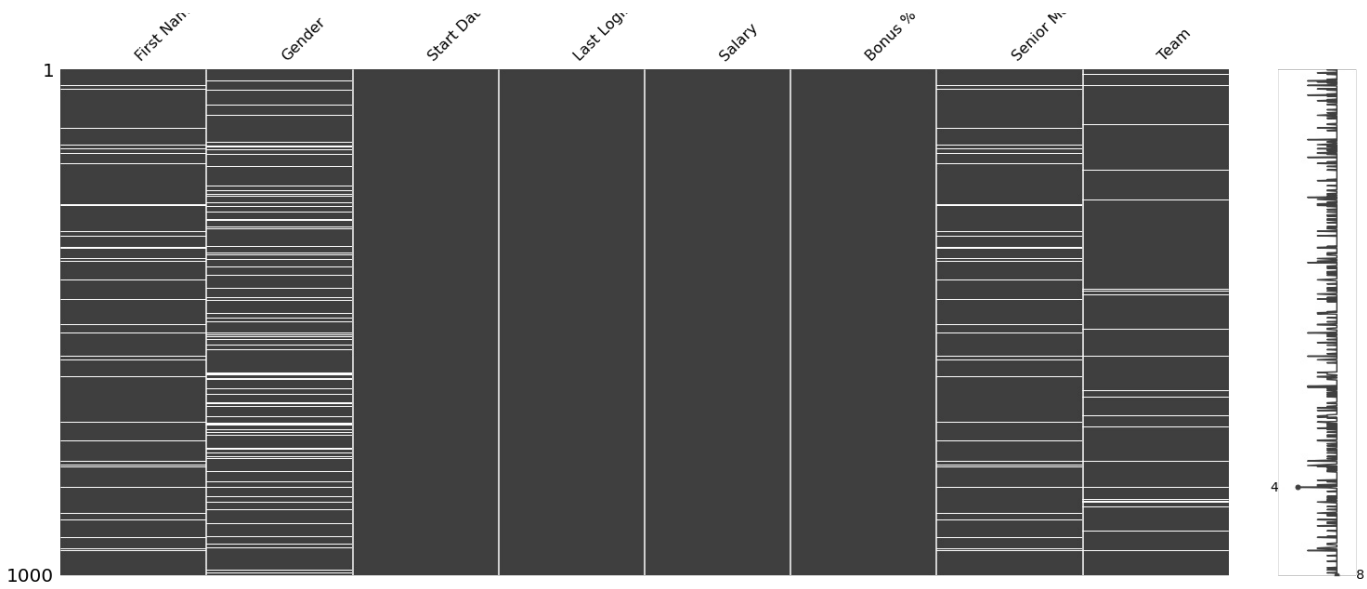
Out[35]:
```
0
```

In [36]:
```
df.isna().sum()
```

Out[36]:
```
First Name           67
Gender              145
Start Date            0
Last Login Time       0
Salary                0
Bonus %               0
Senior Management    67
Team                 43
dtype: int64
```

In [37]:
```
import matplotlib.pyplot as plt
import missingno as msno
```

In [38]:
```
msno.matrix(df)
plt.show()
```



In [39]:
```
missing= df[df["First Name"].isna()]
missing
```

Out[39]:

	First Name	Gender	Start Date	Last Login Time	Salary	Bonus %	Senior Management	Team
7	NaN	Female	7/20/2015	10:43 AM	45906	11.598	NaN	Finance
23	NaN	Male	6/14/2012	4:19 PM	125792	5.042	NaN	NaN
25	NaN	Male	10/8/2012	1:12 AM	37076	18.576	NaN	Client Services
32	NaN	Male	8/21/1998	2:27 PM	122340	6.417	NaN	NaN
39	NaN	Male	1/29/2016	2:33 AM	122173	7.797	NaN	Client Services
...	...	...	...	...	...	...	...	...
902	NaN	Male	5/23/2001	7:52 PM	103877	6.322	NaN	Distribution
925	NaN	Female	8/23/2000	4:19 PM	95866	19.388	NaN	Sales
946	NaN	Female	9/15/1985	1:50 AM	133472	16.941	NaN	Distribution
947	NaN	Male	7/30/2012	3:07 PM	107351	5.329	NaN	Marketing
951	NaN	Female	9/14/2010	5:19 AM	143638	9.662	NaN	NaN

67 rows × 8 columns

In [40]:
```
left_overs = df[~df["First Name"].isna()]
```

In [41]:
```
left_overs
```

Out[41]:

	First Name	Gender	Start Date	Last Login Time	Salary	Bonus %	Senior Management	Team
0	Douglas	Male	8/6/1993	12:42 PM	97308	6.945	True	Marketing
1	Thomas	Male	3/31/1996	6:53 AM	61933	4.170	True	NaN
2	Maria	Female	4/23/1993	11:17 AM	130590	11.858	False	Finance
3	Jerry	Male	3/4/2005	1:00 PM	138705	9.340	True	Finance
4	Larry	Male	1/24/1998	4:47 PM	101004	1.389	True	Client Services
...	...	...	...	...	...	...	...	...
995	Henry	NaN	11/23/2014	6:09 AM	132483	16.655	False	Distribution
996	Phillip	Male	1/31/1984	6:30 AM	42392	19.675	False	Finance
997	Russell	Male	5/20/2013	12:39 PM	96914	1.421	False	Product
998	Larry	Male	4/20/2013	4:45 PM	60500	11.985	False	Business Development
999	Albert	Male	5/15/2012	6:24 PM	129949	10.169	True	Sales

933 rows × 8 columns

In [42]:
```
df_cleaned= df.dropna( axis=0)
```

In [ ]:

In [43]:
```
df.head(10)
```

Out[43]:

	First Name	Gender	Start Date	Last Login Time	Salary	Bonus %	Senior Management	Team
0	Douglas	Male	8/6/1993	12:42 PM	97308	6.945	True	Marketing
1	Thomas	Male	3/31/1996	6:53 AM	61933	4.170	True	NaN
2	Maria	Female	4/23/1993	11:17 AM	130590	11.858	False	Finance
3	Jerry	Male	3/4/2005	1:00 PM	138705	9.340	True	Finance
4	Larry	Male	1/24/1998	4:47 PM	101004	1.389	True	Client Services
5	Dennis	Male	4/18/1987	1:35 AM	115163	10.125	False	Legal
6	Ruby	Female	8/17/1987	4:20 PM	65476	10.012	True	Product
7	NaN	Female	7/20/2015	10:43 AM	45906	11.598	NaN	Finance
8	Angela	Female	11/22/2005	6:29 AM	95570	18.523	True	Engineering
9	Frances	Female	8/8/2002	6:51 AM	139852	7.524	True	Business Development

In [ ]:

In [44]:
```
df.isnull().sum().dropna(inplace=True)
```

In [45]:
```
salary_dist = df.Salary
```

In [46]:
```
men = df.Gender=="Male"
```

In [47]:
```
salary_dist[men].sum()
```

Out[47]:
```
38660604
```

In [48]:
```
men_srn_mangt= df["Senior Management"]==True
```

In [49]:
```
salary_dist[men_srn_mangt&men]
```

Out[49]:
```
0       97308
1       61933
3      138705
4      101004
12     112807
        ...
974     67656
979    142935
992    112769
994     98874
999    129949
Name: Salary, Length: 197, dtype: int64
```

In [50]:
```
snr_manager_bonus= df["Bonus %"]
```

In [51]:
```
team= df.Team =="Marketing"
```

In [52]:
```
snr_manager_bonus[team]
```

Out[52]:
```
0       6.945
21     13.645
26      7.757
43      5.207
62     19.414
        ...
942     6.537
947     5.329
```

```
986    17.999
991    11.051
994     4.479
Name: Bonus %, Length: 98, dtype: float64
```

In [53]:
```
s= np.random.normal(100, 10,size=(4,8))
```

In [54]:
```
df.head(2)
```

Out[54]:

	First Name	Gender	Start Date	Last Login Time	Salary	Bonus %	Senior Management	Team
0	Douglas	Male	8/6/1993	12:42 PM	97308	6.945	True	Marketing
1	Thomas	Male	3/31/1996	6:53 AM	61933	4.170	True	NaN

In [55]:
```
df.dropna(subset= ["Team"],inplace=True)
```

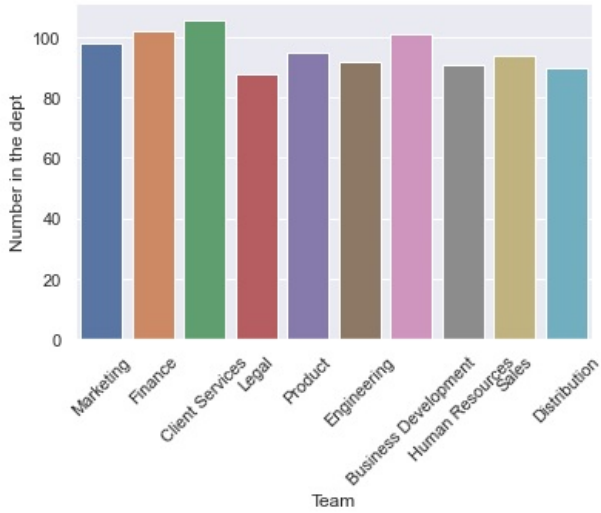
In [56]:
```
df
```

Out[56]:

	First Name	Gender	Start Date	Last Login Time	Salary	Bonus %	Senior Management	Team
0	Douglas	Male	8/6/1993	12:42 PM	97308	6.945	True	Marketing
2	Maria	Female	4/23/1993	11:17 AM	130590	11.858	False	Finance
3	Jerry	Male	3/4/2005	1:00 PM	138705	9.340	True	Finance
4	Larry	Male	1/24/1998	4:47 PM	101004	1.389	True	Client Services
5	Dennis	Male	4/18/1987	1:35 AM	115163	10.125	False	Legal
...	...	...	...	...	...	...	...	...
995	Henry	NaN	11/23/2014	6:09 AM	132483	16.655	False	Distribution
996	Phillip	Male	1/31/1984	6:30 AM	42392	19.675	False	Finance
997	Russell	Male	5/20/2013	12:39 PM	96914	1.421	False	Product
998	Larry	Male	4/20/2013	4:45 PM	60500	11.985	False	Business Development
999	Albert	Male	5/15/2012	6:24 PM	129949	10.169	True	Sales

957 rows × 8 columns

In [57]:
```
sns.set_theme(style="darkgrid")
sns.load_dataset("titanic")
sns.countplot(x="Team", data=df)
plt.ylabel("Number in the dept")
plt.xticks(rotation=45)
plt.show()
```



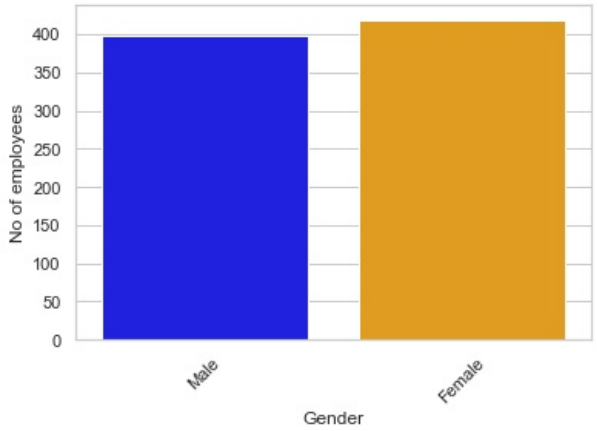
In [58]:
```
df.head(3)
```

	First Name	Gender	Start Date	Last Login Time	Salary	Bonus %	Senior Management	Team
0	Douglas	Male	8/6/1993	12:42 PM	97308	6.945	True	Marketing
2	Maria	Female	4/23/1993	11:17 AM	130590	11.858	False	Finance
3	Jerry	Male	3/4/2005	1:00 PM	138705	9.340	True	Finance

In [59]:
```
df.groupby("Gender")["Salary"].sum()
```

Out[59]:
```
Gender
Female    37555235
Male      36395003
Name: Salary, dtype: int64
```

In [60]:
```
sns.set_style("whitegrid")
sns.countplot( data=df, x="Gender",palette=["Blue", "Orange"])
plt.ylabel("No of employees")
plt.xticks(rotation=45)
plt.show()
```



In [61]:
```
df.Gender.value_counts()
```

Out[61]:
```
Female    418
Male      398
Name: Gender, dtype: int64
```

In [62]:
```
df_groupby_gender = df.groupby("Gender")
```

In [63]:
```
bonus_dist = df_groupby_gender["Salary"].sum()
```

In [64]:
```
plt.plot(bonus_dist, "o" ,alpha=0.5)
plt.show()
```

In [65]:
```
df.head(2)
```

Out[65]:

	First Name	Gender	Start Date	Last Login Time	Salary	Bonus %	Senior Management	Team
0	Douglas	Male	8/6/1993	12:42 PM	97308	6.945	True	Marketing
2	Maria	Female	4/23/1993	11:17 AM	130590	11.858	False	Finance

In [66]:
```
df.rename(columns={"Start Date":"start_date","Last Login Time": "last_login_time", "Bonus %":"percentage_bonus"},
```

In [67]:
```
df.head(1)
```

Out[67]:

	First Name	Gender	start_date	last_login_time	Salary	percentage_bonus	Senior Management	Team
0	Douglas	Male	8/6/1993	12:42 PM	97308	6.945	True	Marketing

In [68]:
```
df_reg= df[[ "Salary", "percentage_bonus"]]
```

In [69]:
```
df_reg
```

Out[69]:

	Salary	percentage_bonus
0	97308	6.945
2	130590	11.858
3	138705	9.340
4	101004	1.389
5	115163	10.125
...	...	...
995	132483	16.655
996	42392	19.675
997	96914	1.421
998	60500	11.985
999	129949	10.169

957 rows × 2 columns

In [70]:
```
df_reg.sum(axis=1)
```

Out[70]:
```
0       97314.945
2      130601.858
3      138714.340
4      101005.389
5      115173.125
          ...
995    132499.655
996     42411.675
997     96915.421
998     60511.985
999    129959.169
Length: 957, dtype: float64
```

In [71]:
```
xs= df_reg.Salary
ys= df_reg.percentage_bonus
```

In [72]:
```
linreg = linregress(xs,ys)
```

In [73]:
```
linreg
```

Out[73]:
```
LinregressResult(slope=-5.354011834280753e-06, intercept=10.68499129811776, rvalue=-0.031721917436836194, pvalue=
0.32694057787918585, stderr=5.458830251155699e-06, intercept_stderr=0.5263336243392354)
```

In [74]:
```
df.head(2)
```

Out[74]:

	First Name	Gender	start_date	last_login_time	Salary	percentage_bonus	Senior Management	Team
0	Douglas	Male	8/6/1993	12:42 PM	97308	6.945	True	Marketing
2	Maria	Female	4/23/1993	11:17 AM	130590	11.858	False	Finance

In [75]:
```
df["date_and_time"] = df.start_date.str.cat(df.last_login_time,sep=" ")
df["date_and_time"]  = pd.to_datetime(df["date_and_time"] )
```

In [76]:
```
df.head(2)
```

Out[76]:

	First Name	Gender	start_date	last_login_time	Salary	percentage_bonus	Senior Management	Team	date_and_time
0	Douglas	Male	8/6/1993	12:42 PM	97308	6.945	True	Marketing	1993-08-06 12:42:00
2	Maria	Female	4/23/1993	11:17 AM	130590	11.858	False	Finance	1993-04-23 11:17:00

In [77]:
```
df.dtypes
```

Out[77]:
```
First Name                    object
Gender                        object
start_date                    object
last_login_time               object
Salary                         int64
percentage_bonus             float64
Senior Management             object
Team                          object
date_and_time         datetime64[ns]
dtype: object
```

In [78]:
```
g = df.Gender
```

In [79]:
```
df.head(3)
```

Out[79]:

	First Name	Gender	start_date	last_login_time	Salary	percentage_bonus	Senior Management	Team	date_and_time
0	Douglas	Male	8/6/1993	12:42 PM	97308	6.945	True	Marketing	1993-08-06 12:42:00
2	Maria	Female	4/23/1993	11:17 AM	130590	11.858	False	Finance	1993-04-23 11:17:00
3	Jerry	Male	3/4/2005	1:00 PM	138705	9.340	True	Finance	2005-03-04 13:00:00

In [80]:
```
df.dtypes
```

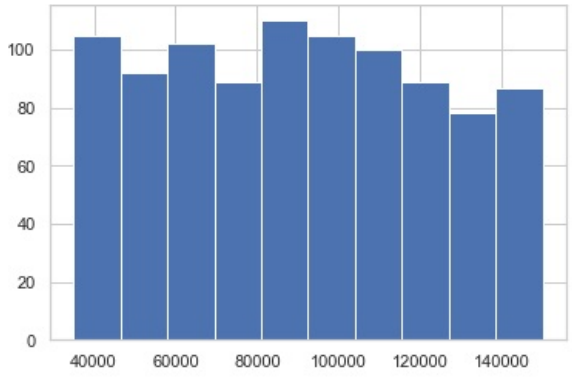
Out[80]:
```
First Name                    object
Gender                        object
start_date                    object
last_login_time               object
Salary                         int64
percentage_bonus             float64
Senior Management             object
Team                          object
date_and_time         datetime64[ns]
dtype: object
```

In [81]:
```
df.head(2)
```

Out[81]:

	First Name	Gender	start_date	last_login_time	Salary	percentage_bonus	Senior Management	Team	date_and_time
0	Douglas	Male	8/6/1993	12:42 PM	97308	6.945	True	Marketing	1993-08-06 12:42:00
2	Maria	Female	4/23/1993	11:17 AM	130590	11.858	False	Finance	1993-04-23 11:17:00

In [82]:
```
df.Salary.hist()
plt.show()
```



In [83]:
```
np.random.seed(25)
```

In [84]:
```
df.Salary.sample(50,replace=True).head(5)
```

Out[84]:
```
138    112238
329     87760
491     58478
150    135490
328     76076
Name: Salary, dtype: int64
```

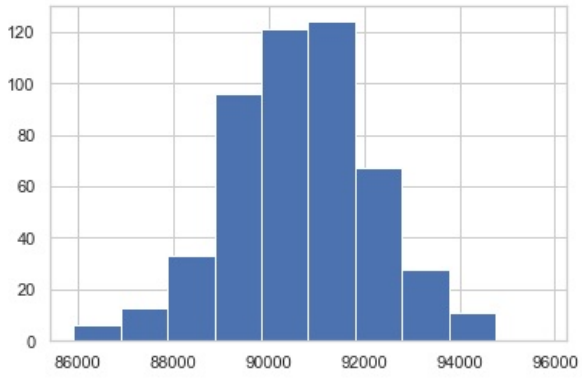
In [ ]:
```

```

In [85]:
```
ran_sample_500 = []
for i in range(500):
    sample_loop = df.Salary.sample(500,replace=True)
    ran_sample_500.append(np.mean(sample_loop))
```

In [86]:
```
SR_sample_series = pd.Series(ran_sample_500)
```

In [87]:
```
SR_sample_series.hist()
```

Out[87]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x229fc8bd790>
```



In [88]:
```
df.head(2)
```

Out[88]:

	First Name	Gender	start_date	last_login_time	Salary	percentage_bonus	Senior Management	Team	date_and_time
0	Douglas	Male	8/6/1993	12:42 PM	97308	6.945	True	Marketing	1993-08-06 12:42:00
2	Maria	Female	4/23/1993	11:17 AM	130590	11.858	False	Finance	1993-04-23 11:17:00

In [89]:
```
#z = (x-μ)/σ
```

In [90]:
```
df_population_mean = df[df.Team == "Marketing"].mean()
```

```
<ipython-input-90-3ed16a7c85c1>:1: FutureWarning: DataFrame.mean and DataFrame.median with numeric_only=None will
```

```
<ipython-input-90-3ed16a7c85c1>:1: FutureWarning: DataFrame.mean and DataFrame.median with numeric_only=None will
include datetime64 and datetime64tz columns in a future version.
  df_population_mean = df[df.Team == "Marketing"].mean()
```

In [91]:
```
df_population_std = df_population_mean.std()
```

In [92]:
```
df_sample_mean = df[df.Team == "Marketing"].sample(100, replace = True).mean()
```

```
<ipython-input-92-f027d5d0999a>:1: FutureWarning: DataFrame.mean and DataFrame.median with numeric_only=None will
include datetime64 and datetime64tz columns in a future version.
  df_sample_mean = df[df.Team == "Marketing"].sample(100, replace = True).mean()
```

In [93]:
```
num = df_population_mean - df_sample_mean
```

In [94]:
```
denom = df_population_std
```

In [95]:
```
df_z_score = num/denom
```

In [96]:
```
df_z_score
```

Out[96]:
```
Salary              -1.925303e-02
percentage_bonus    -1.747564e-05
Senior Management    4.342486e-07
dtype: float64
```

In [ ]:

In [ ]:

In [ ]:

In [97]:
```
df.Team.value_counts()
```

Out[97]:
```
Client Services         106
Finance                 102
Business Development    101
Marketing                98
Product                  95
Sales                    94
Engineering              92
Human Resources          91
Distribution             90
Legal                    88
Name: Team, dtype: int64
```

In [98]:
```
#calculating z_score using proportio
```

In [99]:
```
#Ho = the proportion of women is greater than 10%
```

In [100…
```
p_ho = 0.1
```

In [101…
```
p_hat = (df.Gender == "Female").mean()
```

In [102…
```
n = len(df.Gender)
```

In [103…
```
num = p_hat-p_ho
```

In [104…
```
deno = np.sqrt(p_ho*(1-p_ho)/n)
```

In [105…
```
Z_score_prop = num/deno
```

In [106…
```
Z_score_prop
```

Out[106…
```
34.72826460834943
```

In [107…
```
p_value_proportion = 1-norm.cdf(Z_score_prop)
```

In [108…
```
p_value_proportion
```

Out[108…
```
0.0
```

In [109…
```
#we reject the null hypothesis
```

In [110…
```
df.head(2)
```

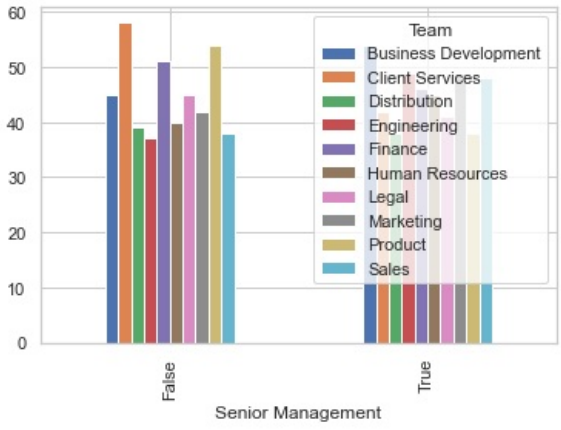
Out[110…

	First Name	Gender	start_date	last_login_time	Salary	percentage_bonus	Senior Management	Team	date_and_time
0	Douglas	Male	8/6/1993	12:42 PM	97308	6.945	True	Marketing	1993-08-06 12:42:00
2	Maria	Female	4/23/1993	11:17 AM	130590	11.858	False	Finance	1993-04-23 11:17:00

In [111…
```
df_stacked = df.groupby("Senior Management").Team.value_counts()
```

In [112…
```
df_stacked.unstack().plot(kind="bar")
```

Out[112…
```
<matplotlib.axes._subplots.AxesSubplot at 0x229fc7863a0>
```



In [113…
```
df_stacked
```

Out[113…
```
Senior Management  Team
False              Client Services         58
                   Product                 54
                   Finance                 51
                   Business Development    45
                   Legal                   45
                   Marketing               42
                   Human Resources         40
                   Distribution            39
                   Sales                   38
                   Engineering             37
True               Business Development    54
                   Engineering             49
                   Marketing               49
                   Sales                   48
                   Finance                 46
                   Human Resources         45
                   Client Services         42
                   Legal                   41
```

```
                  Distribution           38
                  Product                38
Name: Team, dtype: int64
```

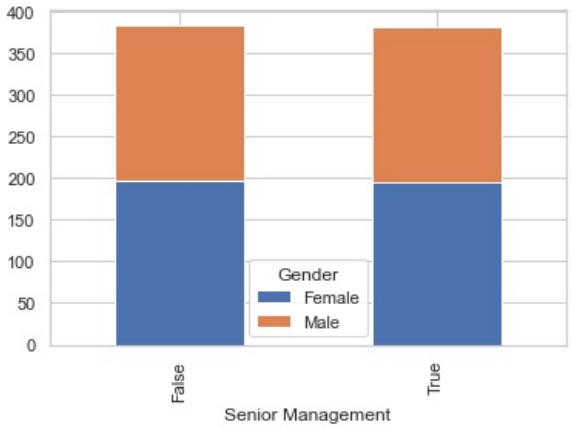
In [114…
```
df_stacked_2 = df.groupby("Senior Management").Gender.value_counts()
```

In [115…
```
df_stacked_2
```

Out[115…
```
Senior Management  Gender
False              Female    197
                   Male      186
True               Female    196
                   Male      185
Name: Gender, dtype: int64
```

In [116…
```
df_stacked_2.unstack().plot(kind="bar", stacked=True)
```

Out[116…
```
<matplotlib.axes._subplots.AxesSubplot at 0x229fc6fbf10>
```



In [ ]:

In [ ]:

In [117…
```
stats = pingouin.chi2_independence(data=df, x= "Senior Management",y="Gender", correction=False)
```

In [118…
```
stats
```

Out[118…
```
(Gender                Female        Male
 Senior Management
 False              197.014398  185.985602
 True               195.985602  185.014398,
 Gender             Female  Male
 Senior Management
 False                 197   186
 True                  196   185,
                  test    lambda      chi2  dof      pval    cramer  power
 0             pearson  1.000000  0.000004  1.0  0.998337  0.000067   0.05
 1       cressie-read  0.666667  0.000004  1.0  0.998337  0.000067   0.05
 2      log-likelihood  0.000000  0.000004  1.0  0.998337  0.000067   0.05
 3       freeman-tukey -0.500000  0.000004  1.0  0.998337  0.000067   0.05
 4  mod-log-likelihood -1.000000  0.000004  1.0  0.998337  0.000067   0.05
 5              neyman -2.000000  0.000004  1.0  0.998337  0.000067   0.05)
```

In [119…
```
df_stacked_team = df.groupby("Team")["Senior Management"].value_counts()
```

In [120…
```
df_stacked_team.unstack()
```

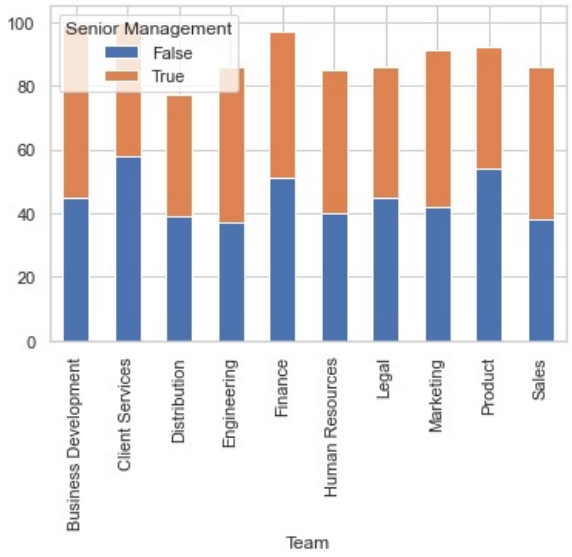
Out[120…

Senior Management	False	True
Team		

Business Development	45	54
Client Services	58	42
Distribution	39	38
Engineering	37	49
Finance	51	46
Human Resources	40	45
Legal	45	41
Marketing	42	49
Product	54	38
Sales	38	48

In [121…
```
df_stacked_team.unstack().plot(kind="bar",stacked=True)
```

Out[121…
```
<matplotlib.axes._subplots.AxesSubplot at 0x229fc8d8d60>
```



In [122…
```
stats_team = pingouin.chi2_independence(data=df, x= "Team",y="Senior Management", correction=False)
```

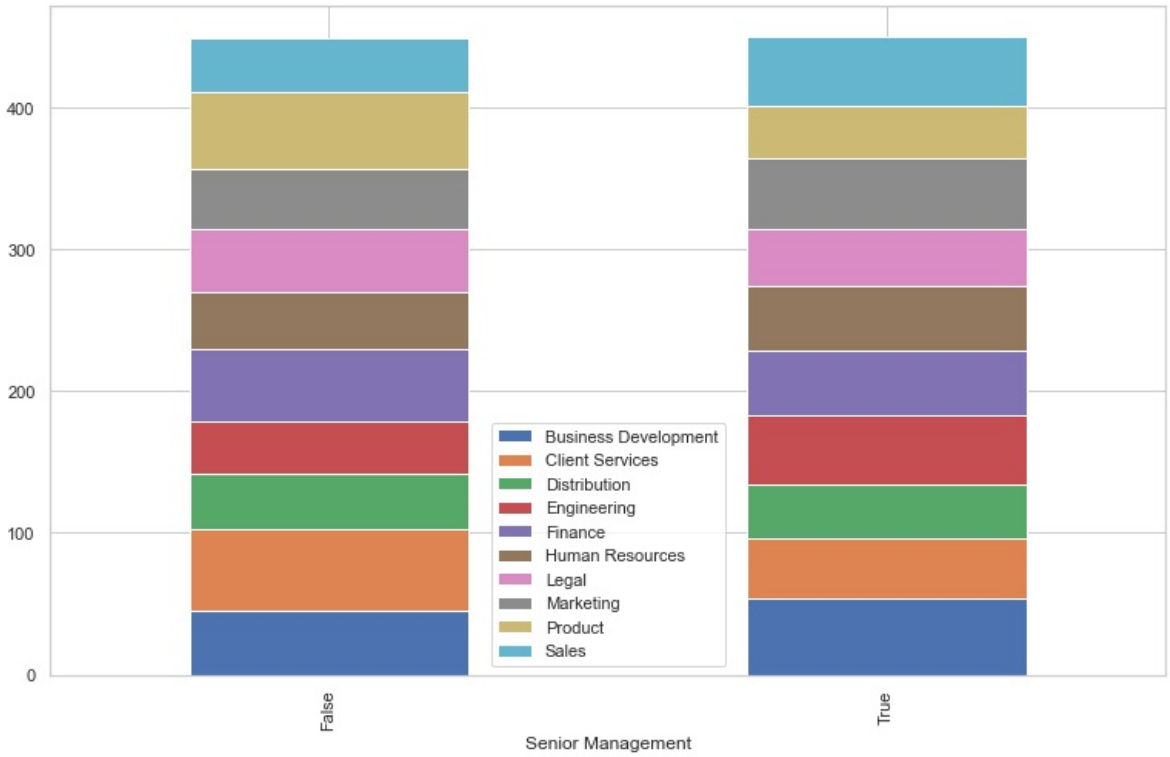
In [123…
```
stats_team
```

Out[123…
```
(Senior Management         False      True
 Team
 Business Development  49.444939  49.555061
 Client Services       49.944383  50.055617
 Distribution          38.457175  38.542825
 Engineering           42.952169  43.047831
 Finance               48.446051  48.553949
 Human Resources       42.452725  42.547275
 Legal                 42.952169  43.047831
 Marketing             45.449388  45.550612
 Product               45.948832  46.051168
 Sales                 42.952169  43.047831,
 Senior Management     False  True
 Team
 Business Development     45     54
 Client Services          58     42
 Distribution             39     38
 Engineering              37     49
 Finance                  51     46
 Human Resources          40     45
 Legal                    45     41
 Marketing                42     49
 Product                  54     38
 Sales                    38     48,
                  test    lambda       chi2  dof      pval    cramer     power
 0             pearson  1.000000  10.286245  9.0  0.327813  0.103675  0.578721
 1        cressie-read  0.666667  10.294036  9.0  0.327210  0.103714  0.579119
 2      log-likelihood  0.000000  10.321593  9.0  0.325083  0.103853  0.580527
 3       freeman-tukey -0.500000  10.352856  9.0  0.322682  0.104010  0.582121
 4  mod-log-likelihood -1.000000  10.393337  9.0  0.319590  0.104213  0.584180
 5              neyman -2.000000  10.502574  9.0  0.311350  0.104759  0.589710)
```

In [ ]:

In [124…
```
df_stacked_SR = df.groupby("Senior Management")["Team"].value_counts()
```

In [125…
```
plt.rcParams["figure.figsize"] = (13,8)
df_stacked_SR.unstack().plot(kind="bar",stacked=True)
plt.legend()
plt.show()
```



In [126…
```
stats_SR = pingouin.chi2_independence(data=df, x= "Senior Management",y= "Team", correction=False)
```

In [127…
```
stats_SR
```

Out[127…
```
(Team               Business Development  Client Services  Distribution  \
 Senior Management
 False                         49.444939        49.944383     38.457175
 True                          49.555061        50.055617     38.542825

 Team               Engineering    Finance  Human Resources      Legal  \
 Senior Management
 False                42.952169  48.446051        42.452725  42.952169
 True                 43.047831  48.553949        42.547275  43.047831

 Team               Marketing    Product      Sales
 Senior Management
 False              45.449388  45.948832  42.952169
 True               45.550612  46.051168  43.047831  ,
 Team               Business Development  Client Services  Distribution  \
 Senior Management
 False                                45               58            39
 True                                 54               42            38

 Team               Engineering  Finance  Human Resources  Legal  Marketing  \
 Senior Management
 False                       37       51               40     45         42
 True                        49       46               45     41         49

 Team               Product  Sales
 Senior Management
 False                   54     38
 True                    38     48  ,
                 test    lambda       chi2  dof      pval    cramer     power
 0            pearson  1.000000  10.286245  9.0  0.327813  0.103675  0.578721
 1        cressie-read  0.666667  10.294036  9.0  0.327210  0.103714  0.579119
 2      log-likelihood  0.000000  10.321593  9.0  0.325083  0.103853  0.580527
 3       freeman-tukey -0.500000  10.352856  9.0  0.322682  0.104010  0.582121
```

```
4  mod-log-likelihood -1.000000  10.393337  9.0  0.319590  0.104213  0.584180
5               neyman -2.000000  10.502574  9.0  0.311350  0.104759  0.589710)
```

In [ ]:

In [128…
```
df_team = df.Team.value_counts()
```

In [129…
```
df_team = df_team.rename_axis("team").reset_index(name="n")
```

In [130…
```
df_team
```

Out[130…

	team	n
0	Client Services	106
1	Finance	102
2	Business Development	101
3	Marketing	98
4	Product	95
5	Sales	94
6	Engineering	92
7	Human Resources	91
8	Distribution	90
9	Legal	88

In [131…
```
df_team["proportion"]= df_team["n"]/len(df)
```

In [132…
```
df_team
```

Out[132…

	team	n	proportion
0	Client Services	106	0.110763
1	Finance	102	0.106583
2	Business Development	101	0.105538
3	Marketing	98	0.102403
4	Product	95	0.099269
5	Sales	94	0.098224
6	Engineering	92	0.096134
7	Human Resources	91	0.095089
8	Distribution	90	0.094044
9	Legal	88	0.091954

In [133…
```
df_team_hypothesized = pd.DataFrame([{"Client Services":0.15,
"Finance":0.1,
"Business Development":0.1,
"Marketing":0.1,
"Product":0.1,
"Sales":0.1,
"Engineering":0.05,
"Human Resources":0.1,
"Distribution":0.1,
"Legal":0.1}]).unstack()
```

In [134…
```
df_team_hypothesized
```

Out[134…
```
Client Services       0    0.15
Finance               0    0.10
Business Development  0    0.10
Marketing             0    0.10
Product               0    0.10
Sales                 0    0.10
```

```
Engineering           0    0.05
Human Resources       0    0.10
Distribution          0    0.10
Legal                 0    0.10
dtype: float64
```

In [135…
```
df_team_hypothesized
```

Out[135…
```
Client Services       0    0.15
Finance               0    0.10
Business Development  0    0.10
Marketing             0    0.10
Product               0    0.10
Sales                 0    0.10
Engineering           0    0.05
Human Resources       0    0.10
Distribution          0    0.10
Legal                 0    0.10
dtype: float64
```

In [136…
```
df_team_hypothesized = pd.DataFrame(df_team_hypothesized).reset_index()
```

In [ ]:

In [137…
```
df_team_hypothesized.rename(columns={"level_0":"team",0: "proportion"},inplace=True)
```

In [138…
```
df_team_hypothesized.drop(columns="level_1", inplace=True)
```

In [139…
```
df_team_hypothesized["n"]= df_team_hypothesized.proportion*len(df)
```

In [140…
```
df_team_hypothesized
```

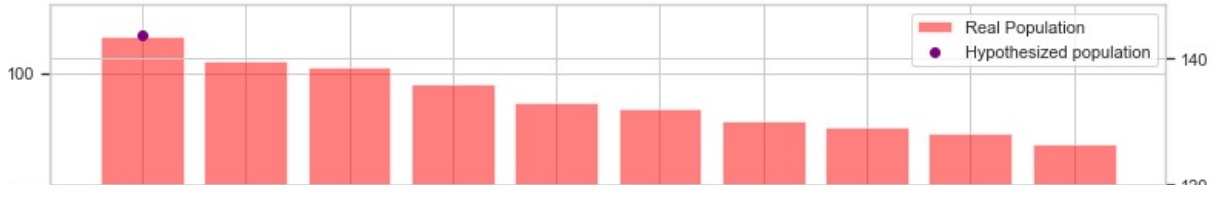
Out[140…

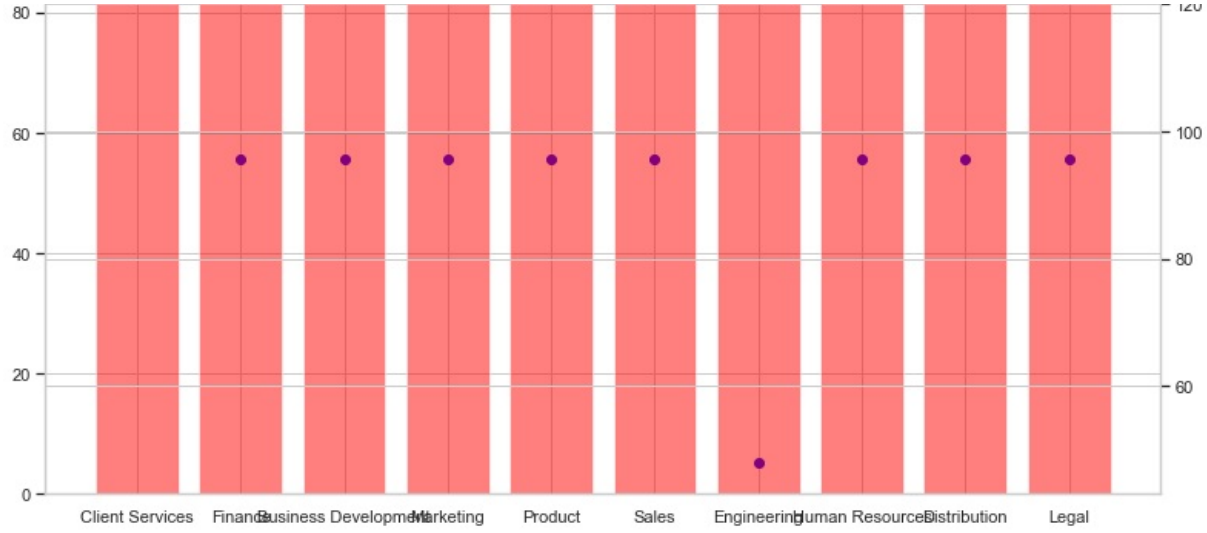
	team	proportion	n
0	Client Services	0.15	143.55
1	Finance	0.10	95.70
2	Business Development	0.10	95.70
3	Marketing	0.10	95.70
4	Product	0.10	95.70
5	Sales	0.10	95.70
6	Engineering	0.05	47.85
7	Human Resources	0.10	95.70
8	Distribution	0.10	95.70
9	Legal	0.10	95.70

In [ ]:

In [ ]:

In [141…
```
ax1 = plt.subplot()
l1 = ax1.bar(df_team["team"], df_team["n"], alpha=0.5,color="red")
ax2 = ax1.twinx()
l2 = ax2.scatter(df_team_hypothesized["team"], df_team_hypothesized["n"],color="purple")
plt.legend([l1, l2], ["Real Population", "Hypothesized population"])
plt.show()
```





In [142…
```
chisquare(df_team["n"], df_team_hypothesized["n"])
```

Out[142…
```
Power_divergenceResult(statistic=52.547196098920224, pvalue=3.56137294841226e-08)
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: