

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
import pandas as pd
sal_data = pd.read_csv(r'/Dataset09-Employee-salary-prediction.csv')
sal_data.head()
```

```
↗
```

	Age	Gender	Education Level	Job Title	Years of Experience	Salary
0	32.0	Male	Bachelor's	Software Engineer	5.0	90000.0
1	28.0	Female	Master's	Data Analyst	3.0	65000.0
2	45.0	Male	PhD	Senior Manager	15.0	150000.0
3	36.0	Female	Bachelor's	Sales Associate	7.0	60000.0
4	52.0	Male	Master's	Director	20.0	200000.0

```
sal_data.shape
```

```
↗ (375, 6)
```

```
sal_data.columns
```

```
↗ Index(['Age', 'Gender', 'Education Level', 'Job Title', 'Years of Experience',
        'Salary'],
        dtype='object')
```

```
sal_data.columns = ['Age', 'Gender', 'Degree', 'Job Title', 'Experience_years', 'Salary']
```

```
sal_data.head()
```

```
↗
```

	Age	Gender	Degree	Job Title	Experience_years	Salary
0	32.0	Male	Bachelor's	Software Engineer	5.0	90000.0
1	28.0	Female	Master's	Data Analyst	3.0	65000.0
2	45.0	Male	PhD	Senior Manager	15.0	150000.0
3	36.0	Female	Bachelor's	Sales Associate	7.0	60000.0
4	52.0	Male	Master's	Director	20.0	200000.0

```
sal_data.dtypes
```

```
↗
```

Age	float64
Gender	object
Degree	object
Job Title	object
Experience_years	float64
Salary	float64

```
dtype: object
```

```
sal_data.info()
```

```
↗ <class 'pandas.core.frame.DataFrame'>
RangeIndex: 375 entries, 0 to 374
Data columns (total 6 columns):
```

#	Column	Non-Null Count	Dtype
0	Age	373 non-null	float64
1	Gender	373 non-null	object
2	Degree	373 non-null	object
3	Job_Title	373 non-null	object
4	Experience_years	373 non-null	float64
5	Salary	373 non-null	float64

dtypes: float64(3), object(3)
memory usage: 17.7+ KB

```
sal_data[sal_data.duplicated()]
```



	Age	Gender	Degree	Job_Title	Experience_years	Salary
195	28.0	Male	Bachelor's	Junior Business Analyst	2.0	40000.0
250	30.0	Female	Bachelor's	Junior Marketing Coordinator	2.0	40000.0
251	38.0	Male	Master's	Senior IT Consultant	9.0	110000.0
252	45.0	Female	PhD	Senior Product Designer	15.0	150000.0
253	28.0	Male	Bachelor's	Junior Business Development Associate	2.0	40000.0
254	35.0	Female	Bachelor's	Senior Marketing Analyst	8.0	85000.0
255	44.0	Male	Bachelor's	Senior Software Engineer	14.0	130000.0
256	34.0	Female	Master's	Senior Financial Advisor	6.0	100000.0
257	35.0	Male	Bachelor's	Senior Project Coordinator	9.0	95000.0
258	50.0	Female	PhD	Director of Operations	22.0	180000.0
260	NaN	NaN	NaN	NaN	NaN	NaN
262	46.0	Male	PhD	Senior Data Scientist	18.0	160000.0
281	41.0	Female	Bachelor's	Senior Project Coordinator	11.0	95000.0
287	35.0	Female	Bachelor's	Senior Marketing Analyst	8.0	85000.0
303	45.0	Male	PhD	Senior Data Engineer	16.0	150000.0
306	49.0	Female	Master's	Director of Marketing	21.0	180000.0
307	31.0	Male	Bachelor's	Junior Operations Analyst	3.0	50000.0
309	47.0	Male	Master's	Director of Marketing	19.0	170000.0
310	29.0	Female	Bachelor's	Junior Business Development Associate	1.5	35000.0
311	35.0	Male	Bachelor's	Senior Financial Manager	9.0	100000.0
312	44.0	Female	PhD	Senior Product Designer	15.0	150000.0
313	33.0	Male	Bachelor's	Junior Business Analyst	4.0	60000.0
314	35.0	Female	Bachelor's	Senior Marketing Analyst	8.0	85000.0
315	44.0	Male	Bachelor's	Senior Software Engineer	13.0	130000.0
317	36.0	Male	Bachelor's	Senior Marketing Specialist	8.0	95000.0
328	38.0	Female	Bachelor's	Senior Business Analyst	10.0	110000.0
345	33.0	Male	Bachelor's	Junior Business Analyst	4.0	60000.0
346	35.0	Female	Bachelor's	Senior Marketing Analyst	8.0	85000.0
352	38.0	Female	Bachelor's	Senior Business Analyst	10.0	110000.0
353	48.0	Male	Master's	Director of Marketing	21.0	180000.0
354	31.0	Female	Bachelor's	Junior Business Development Associate	3.0	50000.0
355	40.0	Male	Bachelor's	Senior Financial Analyst	12.0	130000.0
356	45.0	Female	PhD	Senior UX Designer	16.0	160000.0
357	33.0	Male	Bachelor's	Junior Product Manager	4.0	60000.0
358	36.0	Female	Bachelor's	Senior Marketing Manager	8.0	95000.0
359	47.0	Male	Master's	Director of Operations	19.0	170000.0
360	29.0	Female	Bachelor's	Junior Project Manager	2.0	40000.0
361	34.0	Male	Bachelor's	Senior Operations Coordinator	7.0	90000.0
362	44.0	Female	PhD	Senior Business Analyst	15.0	150000.0
363	33.0	Male	Bachelor's	Junior Marketing Specialist	5.0	70000.0
364	35.0	Female	Bachelor's	Senior Financial Manager	8.0	90000.0
365	43.0	Male	Master's	Director of Marketing	18.0	170000.0
366	31.0	Female	Bachelor's	Junior Financial Analyst	3.0	50000.0

367	41.0	Male	Bachelor's	Senior Product Manager	14.0	150000.0
368	44.0	Female	PhD	Senior Data Engineer	16.0	160000.0
369	33.0	Male	Bachelor's	Junior Business Analyst	4.0	60000.0
370	35.0	Female	Bachelor's	Senior Marketing Analyst	8.0	85000.0
372	29.0	Female	Bachelor's	Junior Project Manager	2.0	40000.0
373	34.0	Male	Bachelor's	Senior Operations Coordinator	7.0	90000.0
374	44.0	Female	PhD	Senior Business Analyst	15.0	150000.0

```
sal_data[sal_data.duplicated()].shape
```

```
(50, 6)
```

```
sal_data1 = sal_data.drop_duplicates(keep = 'first')
sal_data1.shape
```

```
(325, 6)
```

```
sal_data1.isnull().sum()
```

```
0
Age      1
Gender    1
Degree    1
Job_Title 1
Experience_years 1
Salary    1
```

```
dtype: int64
```

```
sal_data1.shape
```

```
(325, 6)
```

```
sal_data1.head()
```

```

Age  Gender  Degree  Job_Title  Experience_years  Salary
0   32.0    Male  Bachelor's  Software Engineer           5.0   90000.0
1   28.0   Female   Master's    Data Analyst           3.0   65000.0
2   45.0    Male     PhD      Senior Manager          15.0  150000.0
3   36.0   Female  Bachelor's   Sales Associate           7.0   60000.0
4   52.0    Male   Master's      Director          20.0  200000.0
```

```
sal_data1.describe()
```



	Age	Experience_years	Salary
count	324.000000	324.000000	324.000000
mean	37.382716	10.058642	99985.648148
std	7.185844	6.650470	48652.271440
min	23.000000	0.000000	350.000000
25%	31.000000	4.000000	55000.000000
50%	36.500000	9.000000	95000.000000
75%	44.000000	16.000000	140000.000000
max	53.000000	25.000000	250000.000000

```
corr = sal_data1[['Age', 'Experience_years', 'Salary']].corr()
corr
```

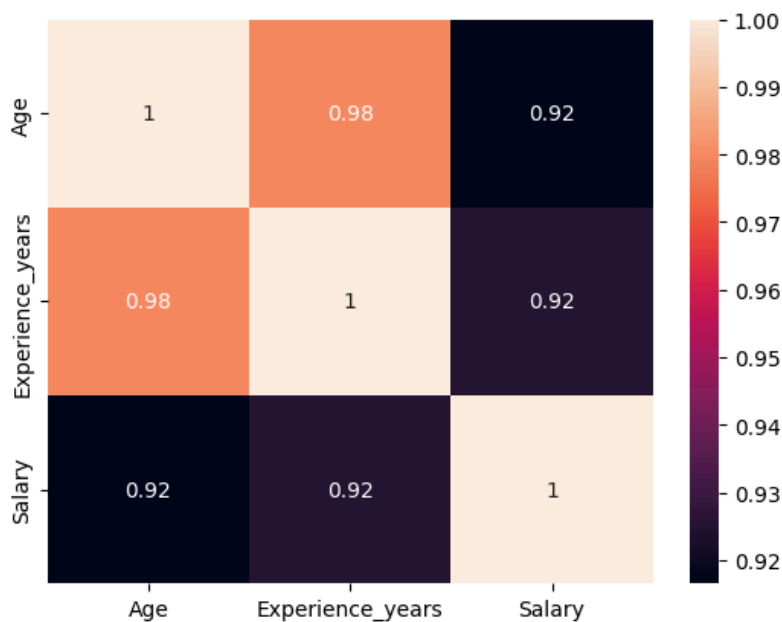


	Age	Experience_years	Salary
Age	1.000000	0.979192	0.916543
Experience_years	0.979192	1.000000	0.924455
Salary	0.916543	0.924455	1.000000

```
import seaborn as sns
sns.heatmap(corr, annot = True)
```



<Axes: >




```
sal_data1['Degree'].value_counts()
```

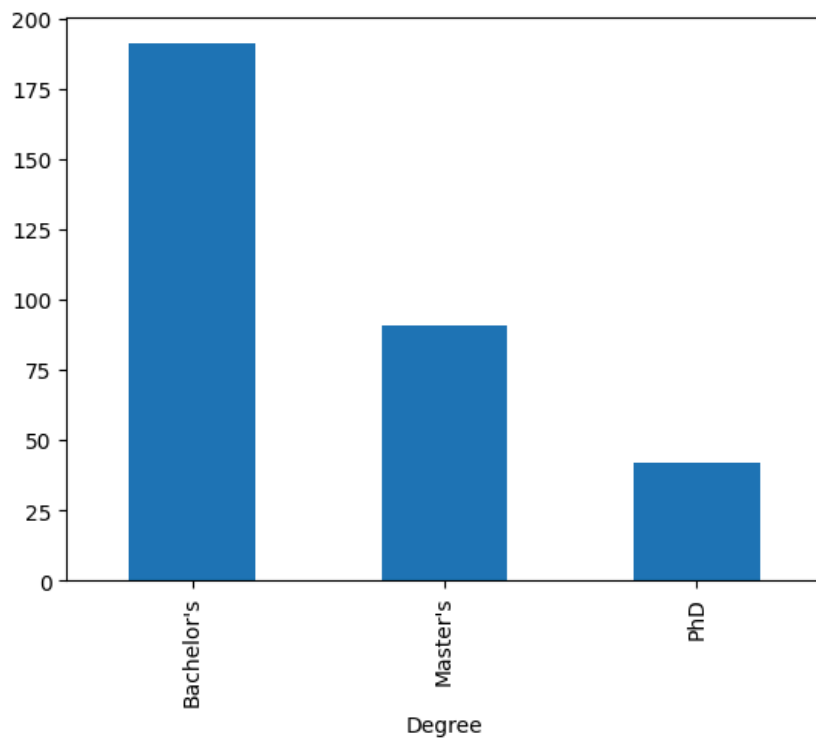


	count
Degree	
Bachelor's	191
Master's	91
PhD	42

dtype: int64

```
sal_data1['Degree'].value_counts().plot(kind = 'bar')
```

 <Axes: xlabel='Degree'>



`sal_data1['Job_Title'].value_counts()`



	count
Job_Title	
Director of Operations	9
Director of Marketing	8
Senior Marketing Manager	8
Senior Project Manager	7
Senior Data Scientist	6
...	...
Junior Social Media Specialist	1
Junior Operations Coordinator	1
Senior HR Specialist	1
Director of HR	1
Junior Financial Advisor	1

174 rows × 1 columns

dtype: int64

`sal_data1['Job_Title'].unique()`




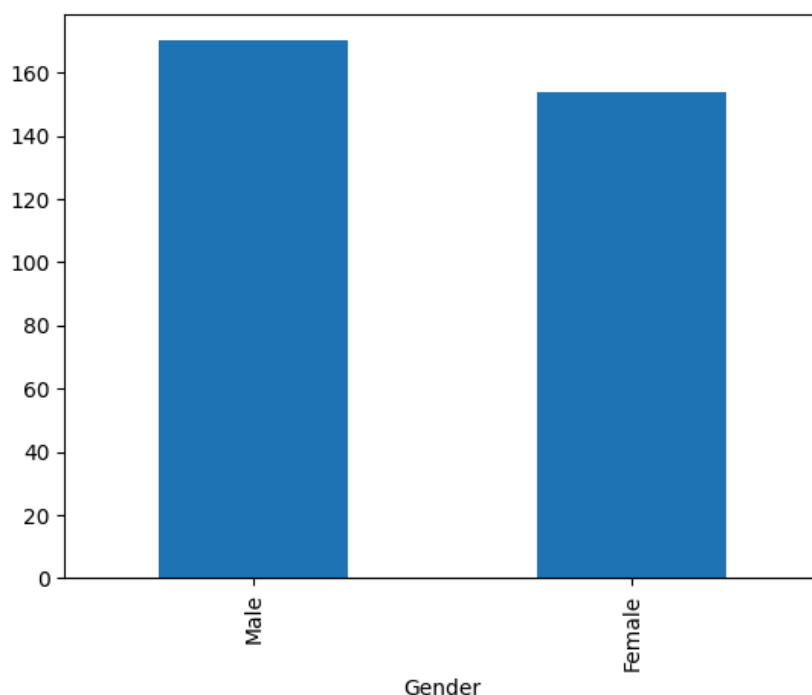
```

Senior Project Coordinator', 'Chief Data Officer',
'Digital Content Producer', 'IT Support Specialist',
'Senior Marketing Analyst', 'Customer Success Manager',
'Senior Graphic Designer', 'Software Project Manager',
'Supply Chain Analyst', 'Senior Business Analyst',
'Junior Marketing Analyst', 'Office Manager', 'Principal Engineer',
'Junior HR Generalist', 'Senior Product Manager',
'Junior Operations Analyst', 'Senior HR Generalist',
'Sales Operations Manager', 'Senior Software Developer',
'Junior Web Designer', 'Senior Training Specialist',
'Senior Research Scientist', 'Junior Sales Representative',
'Junior Marketing Manager', 'Junior Data Analyst',
'Senior Product Marketing Manager', 'Junior Business Analyst',
'Senior Sales Manager', 'Junior Marketing Specialist',
'Junior Project Manager', 'Senior Accountant', 'Director of Sales',
'Junior Recruiter', 'Senior Business Development Manager',
'Senior Product Designer', 'Junior Customer Support Specialist',
'Senior IT Support Specialist', 'Junior Financial Analyst',
'Senior Operations Manager', 'Director of Human Resources',
'Junior Software Engineer', 'Senior Sales Representative',
'Director of Product Management', 'Junior Copywriter',
'Senior Marketing Coordinator', 'Senior Human Resources Manager',
'Junior Business Development Associate', 'Senior Account Manager',
'Senior Researcher', 'Junior HR Coordinator',
'Director of Finance', 'Junior Marketing Coordinator', nan,
'Junior Data Scientist', 'Senior Operations Analyst',
'Senior Human Resources Coordinator', 'Senior UX Designer',
'Junior Product Manager', 'Senior Marketing Specialist',
'Senior IT Project Manager', 'Senior Quality Assurance Analyst',
'Director of Sales and Marketing', 'Senior Account Executive',
'Director of Business Development', 'Junior Social Media Manager',
'Senior Human Resources Specialist', 'Senior Data Analyst',
'Director of Human Capital', 'Junior Advertising Coordinator',
'Junior UX Designer', 'Senior Marketing Director',
'Senior IT Consultant', 'Senior Financial Advisor',
'Junior Business Operations Analyst',
'Junior Social Media Specialist',
'Senior Product Development Manager', 'Junior Operations Manager',
'Senior Software Architect', 'Junior Research Scientist',
'Senior Financial Manager', 'Senior HR Specialist',
'Senior Data Engineer', 'Junior Operations Coordinator',
'Director of HR', 'Senior Operations Coordinator',
'Junior Financial Advisor', 'Director of Engineering'],
dtype=object)

```


```
sal_data1['Gender'].value_counts().plot(kind = 'bar')
```

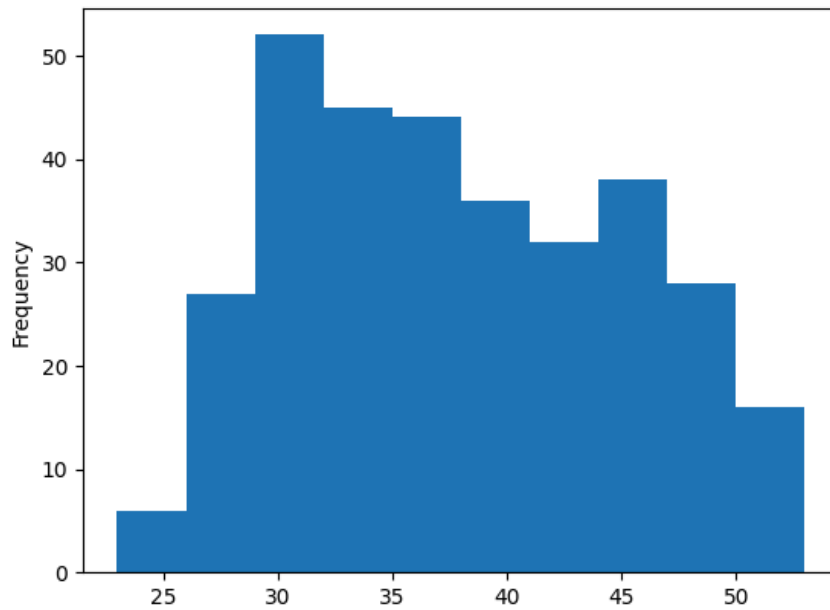
 <Axes: xlabel='Gender'>




####Numerical Variable-Plot Histogram/box plot:

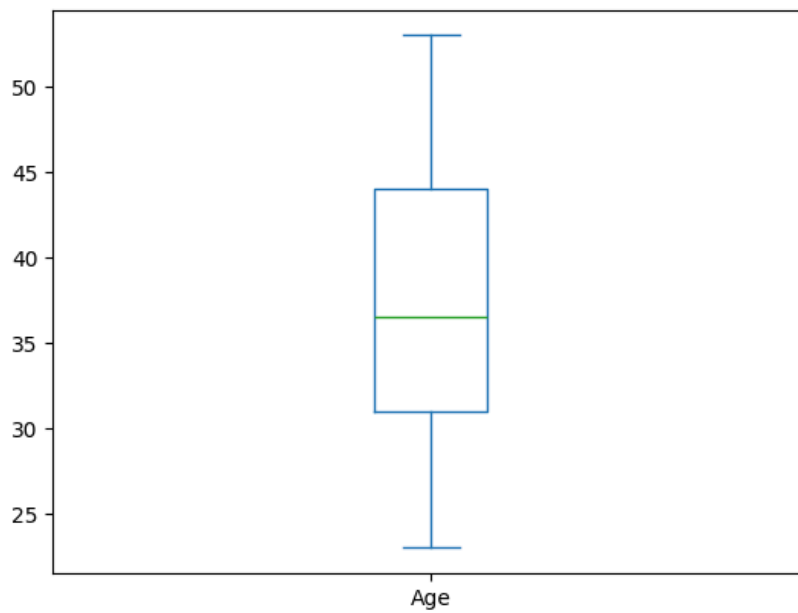
```
sal_data1['Age'].plot(kind = 'hist')
```

 <Axes: ylabel='Frequency'>


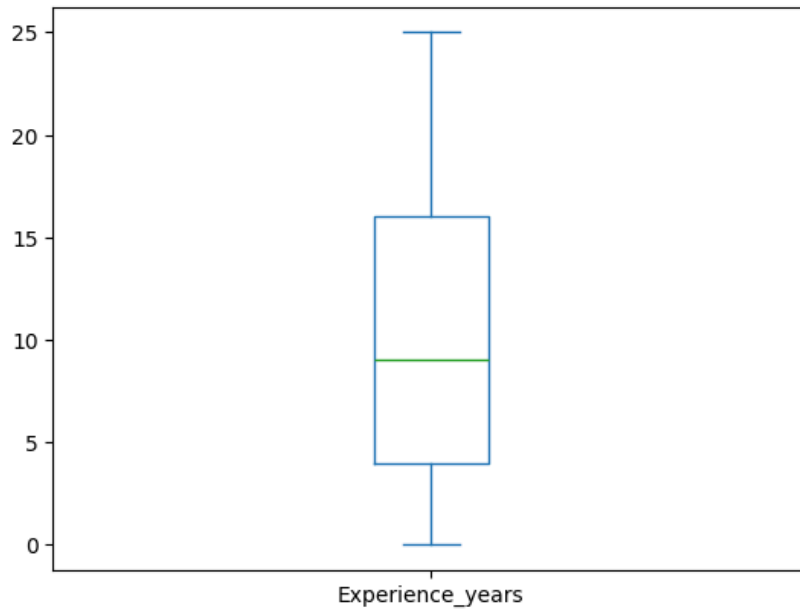


```
sal_data1.Age.plot(kind = 'box')
```

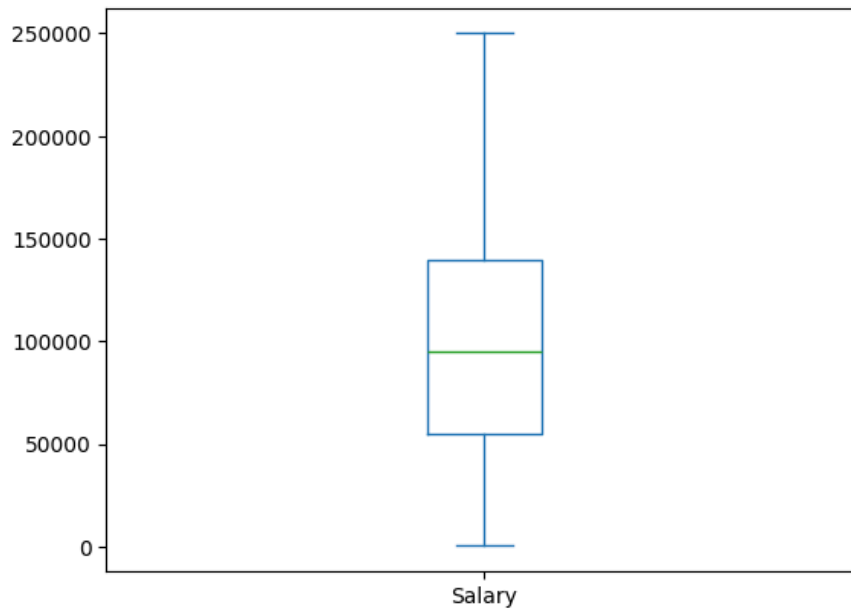
 <Axes: >



```
sal_data1.Experience_years.plot(kind = 'box')
```

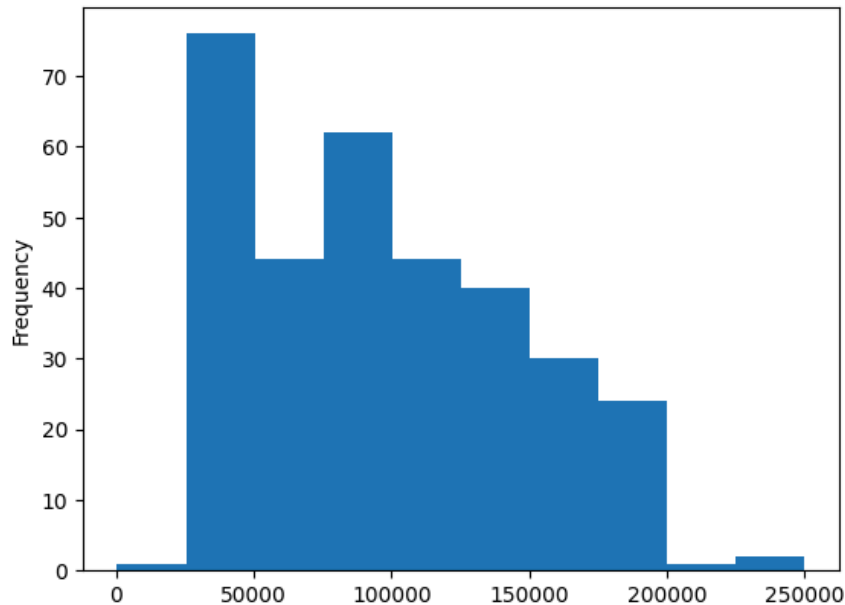

 <Axes: >

```
sal_data1.Salary.plot(kind = 'box')
```

 <Axes: >

```
sal_data1.Salary.plot(kind = 'hist')
```

<Axes: ylabel='Frequency'>



```
sal_data1.head()
```

	Age	Gender	Degree	Job_Title	Experience_years	Salary
0	32.0	Male	Bachelor's	Software Engineer	5.0	90000.0
1	28.0	Female	Master's	Data Analyst	3.0	65000.0
2	45.0	Male	PhD	Senior Manager	15.0	150000.0
3	36.0	Female	Bachelor's	Sales Associate	7.0	60000.0
4	52.0	Male	Master's	Director	20.0	200000.0

```
from sklearn.preprocessing import LabelEncoder
LabelEncoder = LabelEncoder()
```

```
sal_data1['Gender_Encode'] = LabelEncoder.fit_transform(sal_data1['Gender'])
```

```
sal_data1['Degree_Encode'] = LabelEncoder.fit_transform(sal_data1['Degree'])
```

```
sal_data1['Job_Title_Encode'] = LabelEncoder.fit_transform(sal_data1['Job_Title'])
```

```
sal_data1.head()
```

	Age	Gender	Degree	Job_Title	Experience_years	Salary	Gender_Encode	Degree_Encode	Job_Title_Encode
0	32.0	Male	Bachelor's	Software Engineer	5.0	90000.0	1	0	159
1	28.0	Female	Master's	Data Analyst	3.0	65000.0	0	1	17
2	45.0	Male	PhD	Senior Manager	15.0	150000.0	1	2	130

```
from sklearn.preprocessing import StandardScaler
std_scaler = StandardScaler()
```

```
sal_data1['Age_scaled'] = std_scaler.fit_transform(sal_data1[['Age']])
sal_data1['Experience_years_scaled'] = std_scaler.fit_transform(sal_data1[['Experience_years']])
```

```
sal_data1.head()
```

	Age	Gender	Degree	Job_Title	Experience_years	Salary	Gender_Encode	Degree_Encode	Job_Title_Encode	A
0	32.0	Male	Bachelor's	Software Engineer	5.0	90000.0	1	0		159
1	28.0	Female	Master's	Data Analyst	3.0	65000.0	0	1		17
2	45.0	Male	PhD	Senior Manager	15.0	150000.0	1	2		130
3	36.0	Female	Bachelor's	Sales Associate	7.0	60000.0	0	0		101
4	52.0	Male	Master's	Director	20.0	200000.0	1	1		22

```
x = sal_data1[['Age_scaled', 'Gender_Encode', 'Degree_Encode', 'Job_Title_Encode', 'Experience_years_scaled']]
y = sal_data1['Salary']
```

```
x.head()
```

↕

	Age_scaled	Gender_Encode	Degree_Encode	Job_Title_Encode	Experience_years_scaled
0	-0.750231	1	0	159	-0.761821
1	-1.307742	0	1	17	-1.063017
2	1.061680	1	2	130	0.744158
3	-0.192720	0	0	101	-0.460625
4	2.037324	1	1	22	1.497148

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.2, random_state = 42)
```

```
x_train.head()
```

→

	Age_scaled	Gender_Encode	Degree_Encode	Job_Title_Encode	Experience_years_scaled
172	NaN	2	3	174	NaN
183	-1.447120	1	0	69	-1.213615
17	0.225413	1	2	116	0.292364
24	0.504169	1	1	37	0.442962
132	0.364791	0	1	154	0.292364

```
x_train.shape, y_train.shape # 80%
```

```
((260, 5), (260,))
```

```
x_test.shape, y_test.shape # 20%
```

```
((65, 5), (65,))
```

```
from sklearn.linear_model import LinearRegression
Linear_regeression_model = LinearRegression()
```

```
x_train = x_train.fillna(x_train.mean())
# Remove rows where y_train is NaN
nan_mask = y_train.isna()
x_train = x_train[~nan_mask]
y_train = y_train[~nan_mask]
Linear_regeression_model.fit(x_train, y_train)
```



LinearRegression ⓘ ?
LinearRegression()

```
y_pred_lr = Linear_regeression_model.predict(x_test)
y_pred_lr
```



```
array([ 60525.62609168, 130399.42635063, 164969.04724596, 130499.12130869,
        187052.40952656, 165807.46945543,  57213.70811216, 102350.31324583,
        83838.87456416,  98718.56152467, 107944.58835943, 136958.91475683,
        117476.99382497, 131731.17979855,  34329.0002152 ,  38223.52081309,
        148741.17558062,  41777.08701725,  41013.37568196,  48566.80925296,
        31337.35908741, 161721.47681928,  63735.23370508, 137797.3369663 ,
        71822.54975045, 162581.34821304,  16768.60345967,  42201.82441776,
        185241.22306115,  85581.19404613, 177175.35620006,  43104.59418007,
        170582.86752128,  59108.07590664, 149601.04697437,  81170.43211304,
        88507.02670861,  91884.75315344,  81084.63537592, 175516.92943266,
        46686.58583954,  54287.87544968, 104502.07125168, 128811.88029701,
        35472.64655988,  48461.02424403,  39706.87946344,  40007.16804336,
        162721.59440702,  81276.08042879,  85465.95405131, 125126.17761574,
        60330.6794106 , 156367.66216337, 160150.30770147,  60031.85174316,
        92333.72511084, 153924.10814779,  75343.5809074 ,  47142.41949008,
        155236.31643397,  83558.13114597, 176798.72586892, 155987.03788377,
        74761.08799681])
```

```
df = pd.DataFrame({'y_Actual': y_test, 'y_Predicted': y_pred_lr})
df['Error'] = df['y_Actual'] - df['y_Predicted']
df['abs_error'] = abs(df['Error'])
df
```



	y_Actual	y_Predicted	Error	abs_error
235	45000.0	60525.626092	-15525.626092	15525.626092
110	110000.0	130399.426351	-20399.426351	20399.426351
249	170000.0	164969.047246	5030.952754	5030.952754
9	110000.0	130499.121309	-20499.121309	20499.121309
93	170000.0	187052.409527	-17052.409527	17052.409527
...
350	160000.0	155236.316434	4763.683566	4763.683566
233	85000.0	83558.131146	1441.868854	1441.868854
60	170000.0	176798.725869	-6798.725869	6798.725869
124	140000.0	155987.037884	-15987.037884	15987.037884
222	100000.0	74761.087997	25238.912003	25238.912003

65 rows × 4 columns

```
Mean_absolute_error = df['abs_error'].mean()
Mean_absolute_error
```



```
np.float64(8313.822546657084)
```