```python
import pandas as pd

# Load dataset
df = pd.read_csv("D:/Academics/TY SEM 1/FDSL/Assignment 
2/employee_dataset.csv")

print(df.head())
```

```
   EmpID        Name  Age  Gender Department   Salary JoiningDate  \
0      1  Employee_1   50  Female      Sales  90000.0  2015-01-01   
1      2  Employee_2   36    Male    Finance  62500.0  2015-01-02   
2      3  Employee_3   29    Male    Finance  39500.0  2015-01-03   
3      4  Employee_4   42    Male      Sales  35000.0  2015-01-04   
4      5  Employee_5   40    Male    Finance  41500.0  2015-01-05   

   PerformanceScore  WorkHours  
0               3.0       43.0  
1               2.0       54.0  
2               1.0       54.0  
3               4.0       37.0  
4               4.0       37.0  
```

```python
df.isnull().sum()              # check missing values
```

```
EmpID                 0
Name                  0
Age                   0
Gender                0
Department            0
Salary                5
JoiningDate           0
PerformanceScore    177
WorkHours            37
dtype: int64
```

```python
df.fillna(df.mean(numeric_only=True), inplace=True)    # fill NaN with 
mean

df.dropna()                    # drop rows with NaN
```

```
     EmpID               Name  Age  Gender Department   Salary JoiningDate
\
0        1      Employee_1   50  Female      Sales  90000.0  2015-01-01

1        2      Employee_2   36    Male    Finance  62500.0  2015-01-02

2        3      Employee_3   29    Male    Finance  39500.0  2015-01-03

3        4      Employee_4   42    Male      Sales  35000.0  2015-01-04

4        5      Employee_5   40    Male    Finance  41500.0  2015-01-05
```

```
..    ...              ...  ...     ...        ...      ...         ...
995    996    Employee_996   34  Female         HR  31000.0  2017-09-22
996    997    Employee_997   51  Female         IT  56500.0  2017-09-23
997    998    Employee_998   44    Male    Finance  98000.0  2017-09-24
998    999    Employee_999   40  Female      Sales  64500.0  2017-09-25
999   1000  Employee_1000   53  Female      Sales  86000.0  2017-09-26

     PerformanceScore  WorkHours
0                 3.0       43.0
1                 2.0       54.0
2                 1.0       54.0
3                 4.0       37.0
4                 4.0       37.0
..                ...        ...
995               2.0       36.0
996               1.0       44.0
997               4.0       51.0
998               1.0       53.0
999               5.0       40.0

[1000 rows x 9 columns]
```

```
df.isnull().sum()              # check missing values
```

```
EmpID               0
Name                0
Age                 0
Gender              0
Department          0
Salary              0
JoiningDate         0
PerformanceScore    0
WorkHours           0
dtype: int64
```

```
df.fillna(0, inplace=True)     # fill NaN with 0

df.fillna(df.mean(numeric_only=True), inplace=True)    # fill NaN with mean

df.duplicated().sum()          # check duplicates
```

```
np.int64(0)
```

```
df.drop_duplicates(inplace=True)  # remove duplicates
```

```
data = {
    'Name': ['A', 'B', 'C', 'D'],
    'Score': [90, None, 75, None]
}

df2 = pd.DataFrame(data)
print("Before:\n", df2)

df2['Score'] = df2['Score'].fillna(df2['Score'].mean())
print("\nAfter Filling NaN:\n", df2)
```

```
Before:
   Name  Score
0    A   90.0
1    B    NaN
2    C   75.0
3    D    NaN

After Filling NaN:
   Name  Score
0    A   90.0
1    B   82.5
2    C   75.0
3    D   82.5
```

```
data = {
    'Name': ['A', 'B', 'C', 'D', 'E'],
    'Class': ['X', 'X', 'Y', 'Y', 'X'],
    'Marks': [85, 90, 78, 88, 95]
}

df3 = pd.DataFrame(data)
print(df3.groupby('Class')['Marks'].mean())
```

```
Class
X    90.0
Y    83.0
Name: Marks, dtype: float64
```

```
df.groupby('Department')['Salary'].mean()
```

```
Department
Finance      62483.173077
HR           64678.213055
IT           59035.502959
Marketing    63694.335313
Sales        62329.769926
Name: Salary, dtype: float64
```

```
df.groupby('Department')['Age'].agg(['mean','max','min','count'])
```

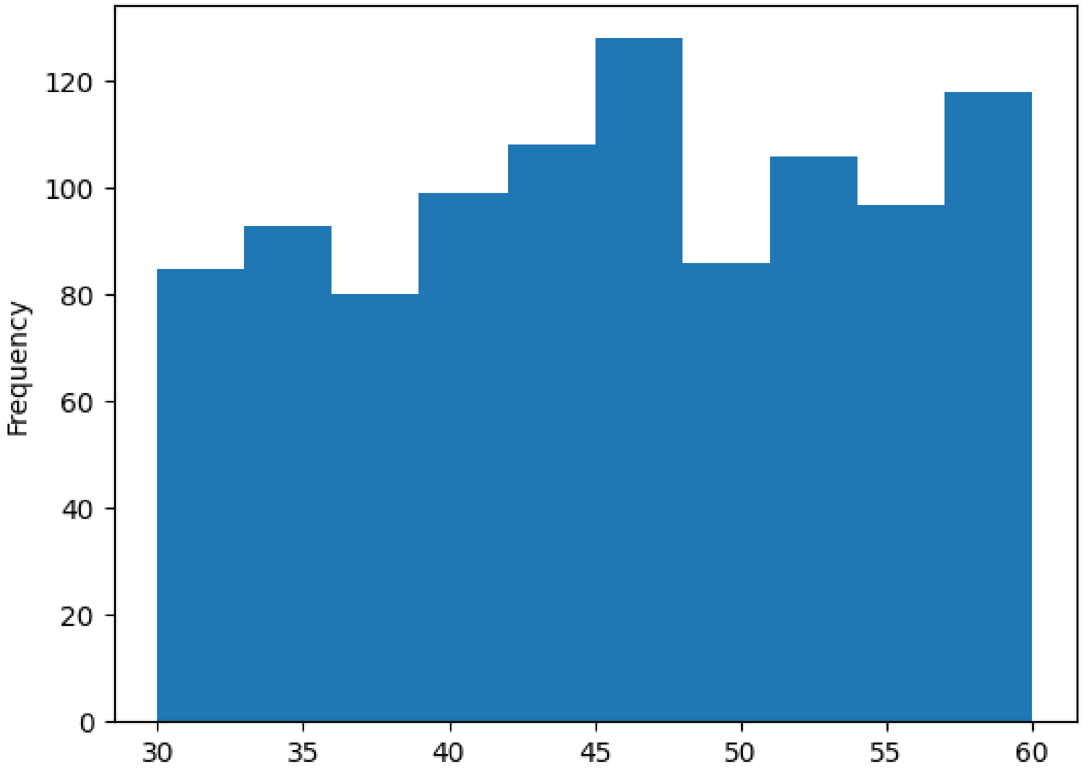
```
                 mean  max  min  count
Department
Finance     41.149038   59   22    208
HR          41.246073   59   22    191
IT          41.142012   59   22    169
Marketing   40.071770   59   22    209
Sales       41.174888   59   22    223
```

```
#group by multiple columns
df.groupby(['Department','Gender'])['WorkHours'].mean()
```

```
Department  Gender
Finance     Female    45.702766
            Male      44.183232
HR          Female    44.039099
            Male      46.803944
IT          Female    46.646512
            Male      46.403882
Marketing   Female    44.595900
            Male      44.122349
Sales       Female    45.436545
            Male      44.550195
Name: WorkHours, dtype: float64
```

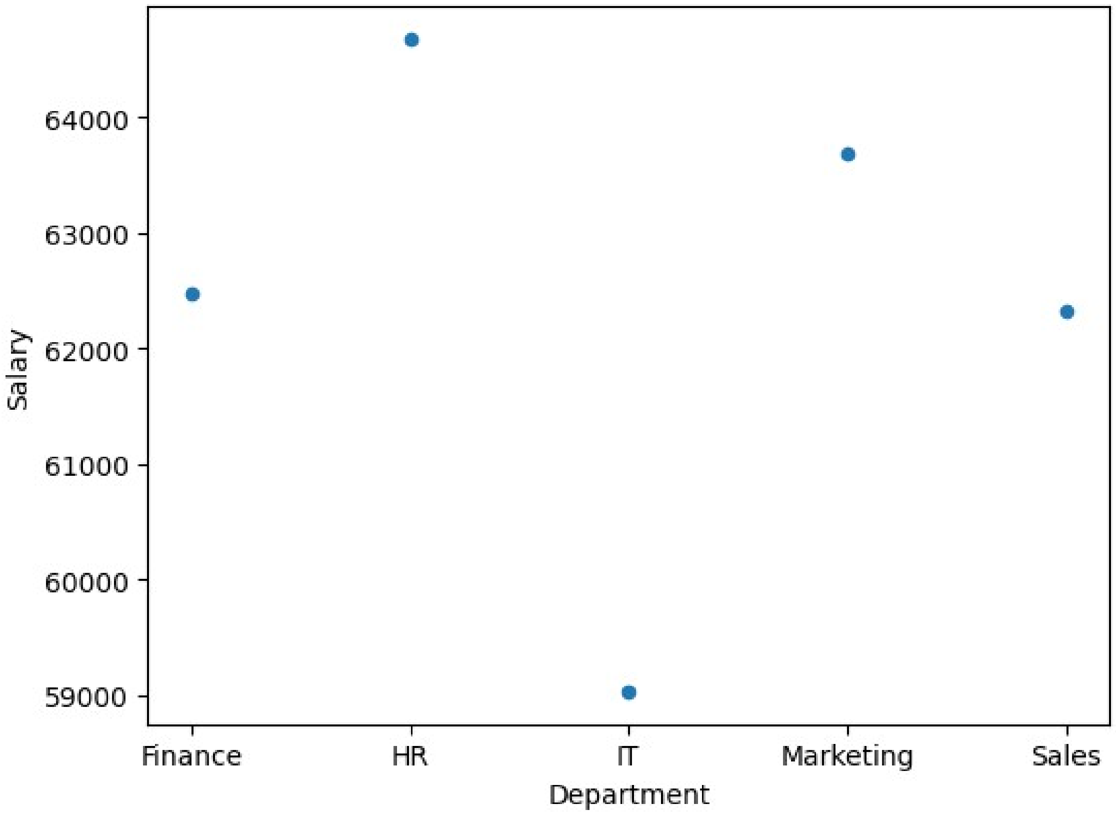
```
df['WorkHours'].plot(kind='hist')     # histogram
```

```
<Axes: ylabel='Frequency'>
```



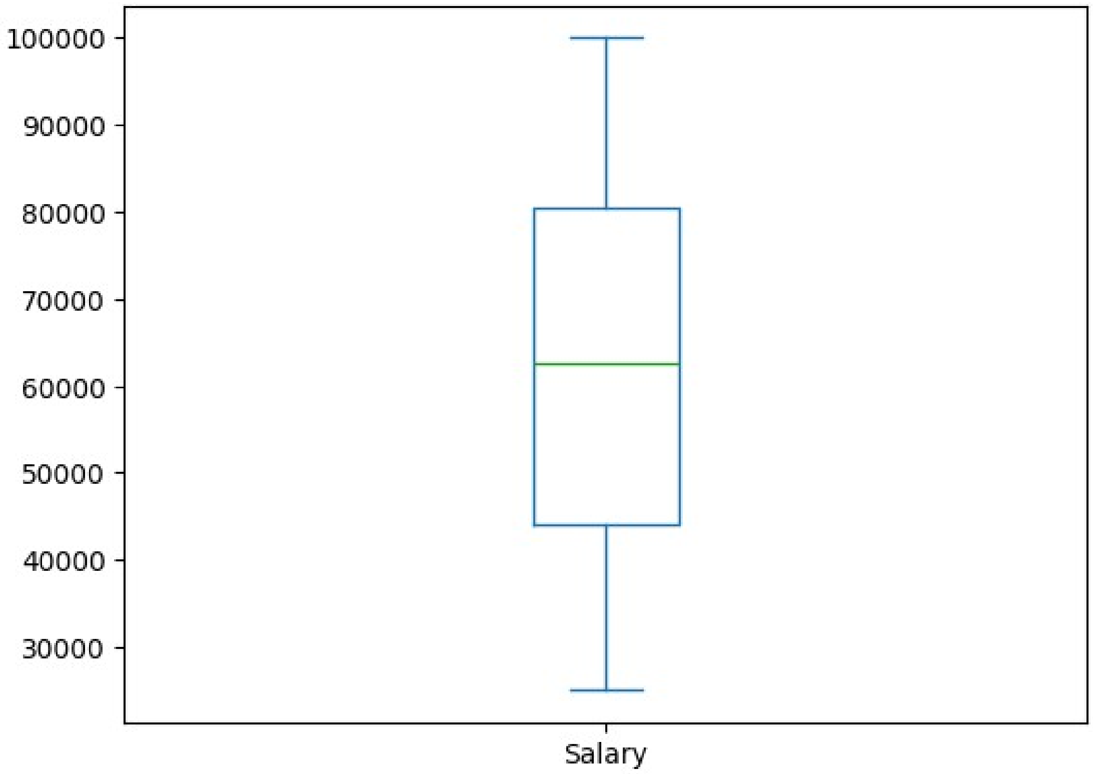
```
# Step 1: Group by Department and calculate mean salary
mean_salary = df.groupby('Department')['Salary'].mean().reset_index()

# Step 2: Plot
mean_salary.plot(x='Department', y='Salary', kind='scatter')
```

```
<Axes: xlabel='Department', ylabel='Salary'>
```



```
df['Salary'].plot(kind='box')      # boxplot
```

```
<Axes: >
```


100000
90000
80000
70000
60000
50000
40000
30000
Salary