**Symbiosis Institute of Technology, Nagpur**

**Sub: GenAI Sem: VII**

**CA – II (Assignment)**

**Name: Ayush Banode**

**PRN: 21070521021**

**Q3] Generate a model for an Insurance company to hold information on the insurer's vehicle,**

**and create a chart of monthly, yearly, and qtrly premiums based on no. of years of insurance**

**where in each year, the value of the vehicle depreciates by 7%.**

**Solution:**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

purchase\_price = 50000

depreciation\_rate = 0.07

years\_of\_insurance = 5

base\_premium\_rate = 0.05

data = {

    'Year': [],

    'Depreciated Value': [],

    'Yearly Premium': [],

    'Quarterly Premium': [],

    'Monthly Premium': []

}

for year in range(1, years\_of\_insurance + 1):

    depreciated\_value = round(purchase\_price \* (1 - depreciation\_rate) \*\* year,2)

    yearly\_premium = round(depreciated\_value \* base\_premium\_rate,2)

    quarterly\_premium = round(yearly\_premium / 4,2)

    monthly\_premium = round(yearly\_premium / 12,2)

    data['Year'].append(year)

    data['Depreciated Value'].append(depreciated\_value)

    data['Yearly Premium'].append(yearly\_premium)

    data['Quarterly Premium'].append(quarterly\_premium)

    data['Monthly Premium'].append(monthly\_premium)

df = pd.DataFrame(data)

plt.figure(figsize=(10, 6))

plt.plot(df['Year'], df['Yearly Premium'], label='Yearly Premium', marker='o')

plt.plot(df['Year'], df['Quarterly Premium'], label='Quarterly Premium', marker='o')

plt.plot(df['Year'], df['Monthly Premium'], label='Monthly Premium', marker='o')

plt.title('Premium Trend Over Time')

plt.xlabel('Year of Insurance')

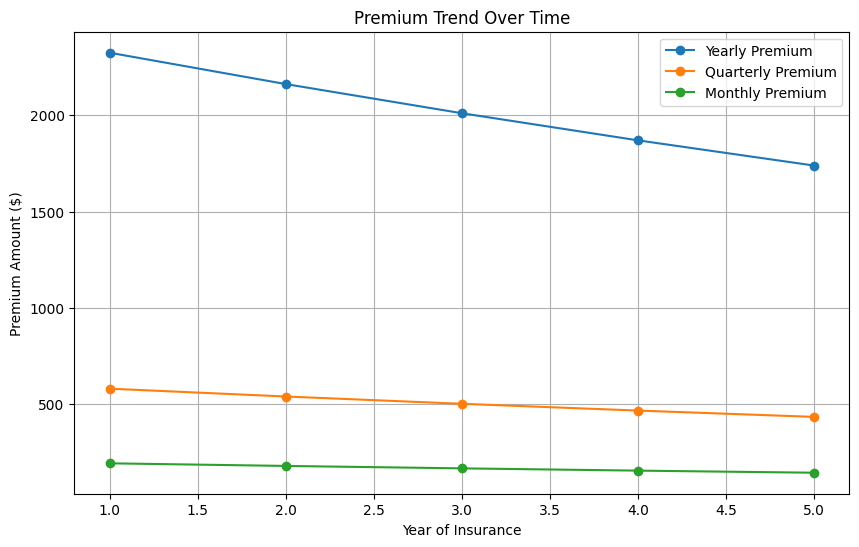
plt.ylabel('Premium Amount ($)')

plt.legend()

plt.grid(True)

plt.show()

df

**Output:**

|  | **Year** | **Depreciated Value** | **Yearly Premium** | **Quarterly Premium** | **Monthly Premium** |
| --- | --- | --- | --- | --- | --- |
| **0** | 1 | 46500.00 | 2325.00 | 581.25 | 193.75 |
| **1** | 2 | 43245.00 | 2162.25 | 540.56 | 180.19 |
| **2** | 3 | 40217.85 | 2010.89 | 502.72 | 167.57 |
| **3** | 4 | 37402.60 | 1870.13 | 467.53 | 155.84 |
| **4** | 5 | 34784.42 | 1739.22 | 434.81 | 144.94 |

**Q5] Generate a model for Covid 19 with symptoms of parameters like fever, cold, shivering,**

**weight loss, generate 100 model data with random values for each parameter and order by**

**parameter lowest to highest in display based on the input parameter.**

**Solution:**

import pandas as pd

import numpy as np

np.random.seed(42)

data = {

    'Fever (°F)': np.round(np.random.uniform(99, 104, 100), 2),

    'Cold': np.random.randint(0, 2, 100),

    'Shivering': np.random.randint(0, 2, 100),

    'Weight Loss (%)': np.round(np.random.uniform(0, 10, 100), 2),

}

df = pd.DataFrame(data)

def sort\_by\_parameter(df, parameter):

    if parameter in df.columns:

        sorted\_df = df.sort\_values(by=[parameter], ascending=True).reset\_index(drop=True)

        return sorted\_df

    else:

        return "Invalid parameter. Choose from 'Fever (°F)', 'Cold', 'Shivering', 'Weight Loss (%)'."

sorted\_df = sort\_by\_parameter(df, 'Fever (°F)')

sorted\_df.head(20)

**Output:**

|  | **Fever (°F)** | **Cold** | **Shivering** | **Weight Loss (%)** |
| --- | --- | --- | --- | --- |
| **0** | 99.03 | 1 | 1 | 8.67 |
| **1** | 99.10 | 0 | 1 | 5.49 |
| **2** | 99.13 | 0 | 1 | 6.23 |
| **3** | 99.17 | 1 | 0 | 9.15 |
| **4** | 99.23 | 0 | 1 | 6.31 |
| **5** | 99.23 | 1 | 0 | 5.70 |
| **6** | 99.29 | 1 | 1 | 1.01 |
| **7** | 99.32 | 0 | 0 | 0.94 |
| **8** | 99.33 | 1 | 1 | 5.77 |
| **9** | 99.37 | 1 | 0 | 6.50 |
| **10** | 99.37 | 0 | 0 | 3.59 |
| **11** | 99.44 | 0 | 1 | 9.36 |
| **12** | 99.49 | 1 | 0 | 0.24 |
| **13** | 99.54 | 1 | 0 | 0.85 |
| **14** | 99.58 | 1 | 0 | 7.96 |
| **15** | 99.60 | 0 | 0 | 0.31 |
| **16** | 99.61 | 1 | 0 | 9.40 |
| **17** | 99.70 | 0 | 0 | 5.68 |
| **18** | 99.70 | 1 | 1 | 6.97 |
| **19** | 99.78 | 0 | 0 | 0.09 |