Dr. D. Y. Patil Unitech Society

Dr. D. Y. Patil Institute of Technology

Affiliated to Savitribai Phule Pune University, Pune

Report on

Database Management System Project

“The Car Insurance Company Dataset Entry”

Submitted in partial fulfilment for the award of degree of **Bachelor of Engineering**

In

Artificial Intelligence & Data Science Engineering

Submitted by

Gurukul Vijay More

TE-A 77

Under the Guidance of

Ms. Shubhangi Suryawanshi

Dr. D.Y. Patil Institute of Technology, Pimpri, Pune Department of Artificial Intelligence and Data Science

CERTIFICATE

This is to certify that **Gurukul Vijay More** from **Third Year Artificial Intelligence and Data Science Engineering** has successfully completed **his** / her Mini Project in **DBMS** in the partial fulfilment of the Bachelor's Degree in Engineering.

Ms. Shubhangi Suryawanshi	**Dr. Mithra Venkatesan**	**Dr. Lalit Kumar Wadhawa**
Mini Project Coordinator	**HOD**	**Principal**

INDEX

Mini-Project (DBMS): Car Insurance Company Dataset Entry

INTRODUCTION

The Car Insurance Company Dataset Entry project aims to create a simple data entry system for an insurance company that focuses on car insurance policies. The project utilizes a Python-based front end with Tkinter for the graphical user interface (GUI) and a MySQL backend fordata storage. The primary objective is to allow users to input and store car insurance policy information efficiently.

Front End with Tkinter:

The front end of the application is designed using the Tkinter library in Python. Tkinter provides a user-friendly interface for data entry. The GUI includes input fields for Policy ID, Customer Name, Car Details, and Premium. Upon submission, the entered data is sent to thebackend for storage in the MySQL database.

Back End with MySQL:

The backend of the project employs a MySQL database to store the insurance policy information. A table named **insurance data** is created to hold the data. This table includes columns for Policy ID, Customer Name, Car Details, and Premium. Policy ID is set as the primary key to ensure uniqueness.

Integration between Front End and Back End:

The integration is achieved through a Python connector library **mysql-connector-python**. This library enables communication between the Python front end and the MySQL back end. The **INSERT** query is used to add the entered data into the **insurance data** table.

PROBLEM STATEMENT

Objective -

The objective of entering data into a car insurance company dataset is to accurately and efficiently record and manage information related to the insurance policies, customers, and claims processed by the company. This data entry process serves several important purposes:

1. Policy Management: Data entry ensures that details about each insurance policy, including the policyholder's information, coverage, premium payments, and policy terms, are accurately recorded and can be easily accessed when needed.

2. Customer Information: It involves inputting customer data, such as names, contact information, and vehicle details, which helps in maintaining a comprehensive database of clients and their insured assets.

3. Claims Processing: Recording information about claims, including claimants, incident details, and payout amounts, is crucial for efficiently managing and settling claims.

4. Risk Assessment: Data entry can include information on factors affecting risk assessment, such as the location of insured vehicles, driving history, and accident records. This data is used to calculate premiums and determine policy eligibility.

5. Regulatory Compliance: Accurate data entry is essential to ensure the company complies with legal and regulatory requirements. Detailed records can be audited and used to provide evidence of compliance with industry standards and regulations.

6. Reporting and Analysis: Accurate and complete data entry supports data analysis and reporting, enabling the company to identify trends, assess the performance of policies, and make informed business decisions.

7. Customer Service: Having up-to-date customer information helps in providing better customer

service, such as addressing inquiries, renewing policies, and sending out policy-related communications.

8. Fraud Prevention: Effective data entry is a key element in fraud detection and prevention. It helps in identifying irregularities and inconsistencies in claims and customer data.

9. Operational Efficiency: Maintaining a well-organized and accurate dataset streamlines internal operations, making it easier for employees to access relevant information when assisting customers or processing claims.

Challenges –

Entering and managing a dataset for a car insurance company can present several challenges, both in terms of data collection and data management. Here are some of the challenges you might encounter:

1. Data Quality:

- Incomplete or inaccurate data: Data entry errors or missing information can lead to flawed analysis and predictions.
- Duplicates: Duplicate entries can skew your analysis and waste storage resources.

2. Data Privacy and Security:

- Sensitive information: Car insurance datasets often contain personally identifiable information (PII). Safeguarding this data is crucial to comply with privacy regulations like GDPR and protect customers' privacy.

3. Data Volume:

- Large datasets: Car insurance companies deal with vast amounts of data. Handling and storing this data efficiently can be a significant challenge.

4. Data Variety:

- Heterogeneous data sources: Data may come from various sources, such as claims, customer information, telematics, and external data like weather or traffic. Integrating and normalizing this data can be complex.

5. Data Cleaning:

- Preprocessing: Cleaning and preparing the data for analysis may involve dealing with outliers, handling missing values, and standardizing data formats.

6. Data Integration:

- Merging datasets: Combining data from different sources can be challenging, especially when they have varying structures or formats.

7. Data Maintenance:

- Regular updates: Insurance data evolves over time, with new policies, claims, and customer information. Keeping the dataset up to date is essential for accurate analysis.

8. Compliance:

- Legal and regulatory requirements: Adhering to insurance regulations and data protection laws can be complicated. Failure to comply can result in fines and legal issues.

9. Data Governance:
- Establishing data governance policies and procedures to ensure data quality, accuracy, and security.

10. Analysis and Reporting:
- Extracting meaningful insights: Analyzing the data to extract valuable insights for the business can be challenging, especially when dealing with complex datasets.

11. Storage and Infrastructure:
- Data storage and infrastructure costs: Managing and maintaining the required storage space and computing resources can be costly.

12. Model Deployment:
- If the dataset is used to train machine learning models for risk assessment, fraud detection, or customer segmentation, deploying and updating these models in a production environment can be complex.

13. Data Access Control:
- Restricting access to the dataset to authorized personnel and ensuring data security is vital.

14. Data Backups and Recovery:
- Implementing data backup and recovery procedures to prevent data loss due to system failures or cyberattacks.

15. Customer Consent:
- Ensuring that you have the necessary consent to collect and use customer data for insurance purposes.

Goal –

The goal of creating a dataset for a car insurance company typically involves collecting and organizing relevant information about policyholders, their vehicles, and their insurance policies. This dataset can be used for various purposes, including:

1. Risk Assessment: Analyzing data to assess the level of risk associated with insuring specific individuals or vehicles. Factors such as the driver's age, driving history, vehicle make and model, and geographical location can impact the risk assessment.

2. Pricing and Premium Calculation: Using the dataset to determine appropriate insurance premiums for policyholders. This may involve considering the risk assessment and other relevant data toset fair pricing.

3. Claims Processing: Managing claims made by policyholders, including collecting information about accidents, damage,and other incidents. This data helps in processing and adjudicating claims efficiently.

4. Fraud Detection: Detectingand preventing insurancefraud by identifying inconsistencies or suspicious patterns in the data, such as false claims or manipulated information.

5. Customer Service: Improving customer service by having a comprehensive view of policyholders, their history, and their preferences. This helps in providing better service and addressing customer inquiries.

6. Marketing and Sales: Using customer data to identify potential leads for new policies and develop targeted marketing campaigns.

7. Compliance: Ensuring compliance with regulatory requirements by maintaining accurate records and reporting data as necessary.

The specific goals of creating a car insurance dataset entry may vary based on the company's needs, but generally, it's about collecting, organizing, and utilizing data to operate the business effectively, assess risk.

Outcome –

The "outcome" of a car insurance company dataset entry typically refers to the result or information recorded for a specific insurance policy or claim within the dataset. This outcome information can vary depending on the dataset's focus and the specific entry. Here are some common outcomes or data points you might find in a car insurance dataset entry:

1. Claim Status: Whether the claim associated with the entry was approved, denied, or is still pending.

2. Payment Amount: The amount of money paid out to the policyholder or a third party for a claim.

3. Premium Amount: The cost of the insurance premium for the specific policy.

4. Policy Coverage: Details of the coverage provided by the insurance policy, including deductibles, limits, and coverage types (e.g., liability, collision, comprehensive).

5. Claim Details: Information about the nature of the claim, such as the cause of the accident, the extent of damage, and the parties involved.

6. Driver Information: Details about the policyholder or driver, including age, gender, driving history, and vehicle information.

7. Location: The geographic location where the insured vehicle is primarily driven and parked.

8. Vehicle Information:Details about the insured vehicle, including make, model, year, and VIN (Vehicle Identification Number).

9. Policyholder Information:Personal information about the policyholder, such as their name, contact information, and any additional insured parties.

10. Effective Dates: The start and end dates of the insurance policy.

METHODOLOGICAL DETAIL

1. Graphical User Interface (GUI):

- The GUI is created using the tkinter library. This allows for the creation of windows, labels, entry fields, buttons, and other GUI elements to provide a user-friendly interface.

2. Database Connectivity:

- The script establishes a connection to a MySQL database using the `mysql.connector` library. It provides the necessary information to connect to the database, including the hostname, username, password, and database name.

3. MySQL Database Interaction:

- The script interacts with the MySQL database to perform data insertion and retrieval. It uses SQL queries to insert and retrieve data from the "insurance_data" table.

4. Data Handling:

- The `pandas` library is used to create a DataFrame from the data fetched from the MySQL database. This DataFrame is then used to save the data to an Excel file.

5. Functions and Error Handling:

- The script defines functions to insert data (`insert_data`) and save data to Excel (`save_to_excel`). It also includes error handling to manage exceptions that may occur during database interactions.

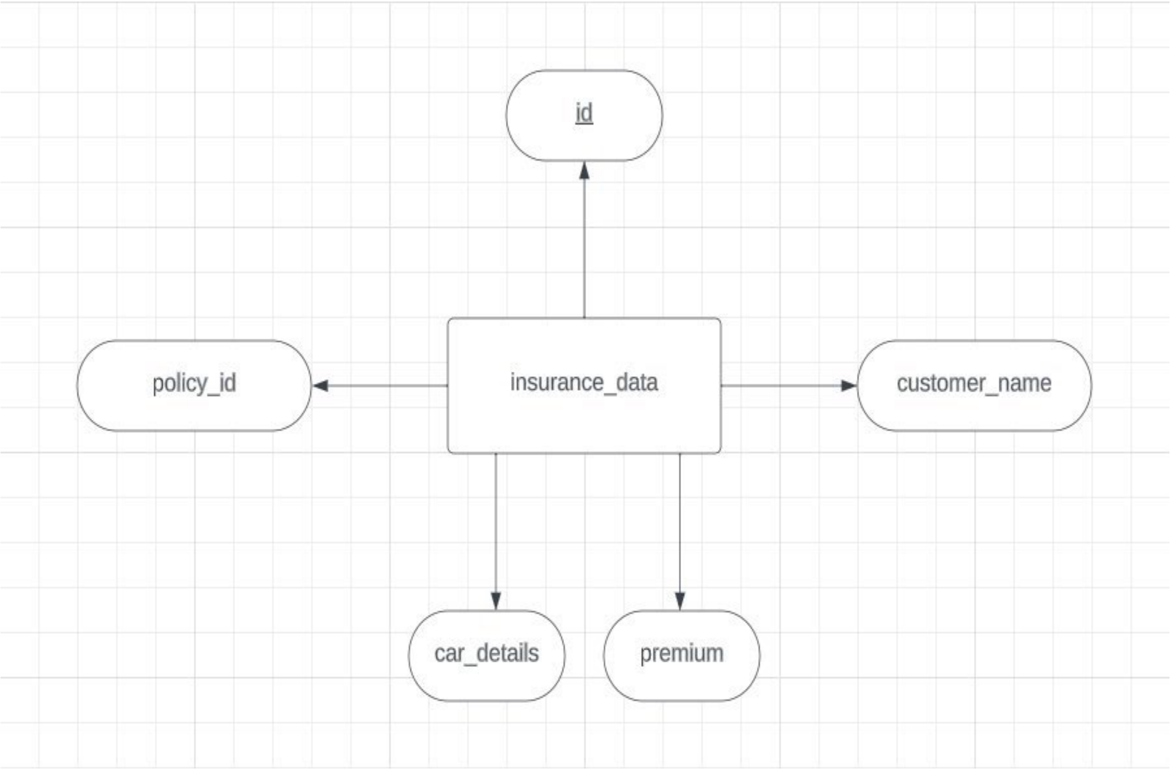
6. Layout and Widgets:

- The GUI layout is organized using frames to group related widgets. Labels and entry fields are used to collect data from the user. A "Submit" button is used to trigger data insertion. Styling is applied to some elements using configuration settings like background color and text font.

7. Event Loop:

- The script starts the tkinter GUI event loop using `root.mainloop()`. This keeps the application running and responsive to user interactions.

ER Diagram


id
policy_id
insurance_data
customer_name
car_details
premium

TECHNOLOGY USED

1. Tkinter (GUI Framework):

- Tkinter is a standard GUI library for Python. It is used to create the graphical user interface (GUI) for the application. Tkinter provides widgets such as labels, entry fields, buttons, frames, and more to design the application's interface.

2. MySQL (Database Management System):

- MySQL is a popular relational database management system. The code interacts with a MySQL database to insert and retrieve data. It uses the `mysql.connector` library to establish a connection and perform database operations.

3. Pandas (Data Analysis and Manipulation):

- Pandas is a data manipulation and analysis library for Python. In this code, Pandas is used to create a DataFrame from the data retrieved from the MySQL database. This DataFrame is then used to export data to an Excel file.

4. Python (Programming Language):

- The entire code is written in the Python programming language. Python is used to develop the application's logic, define functions, handle exceptions, and interact with external libraries and technologies.

5. MySQL Database:

- The code connects to a MySQL database hosted on the local machine. MySQL is used as the data storage and retrieval system for car insurance data.

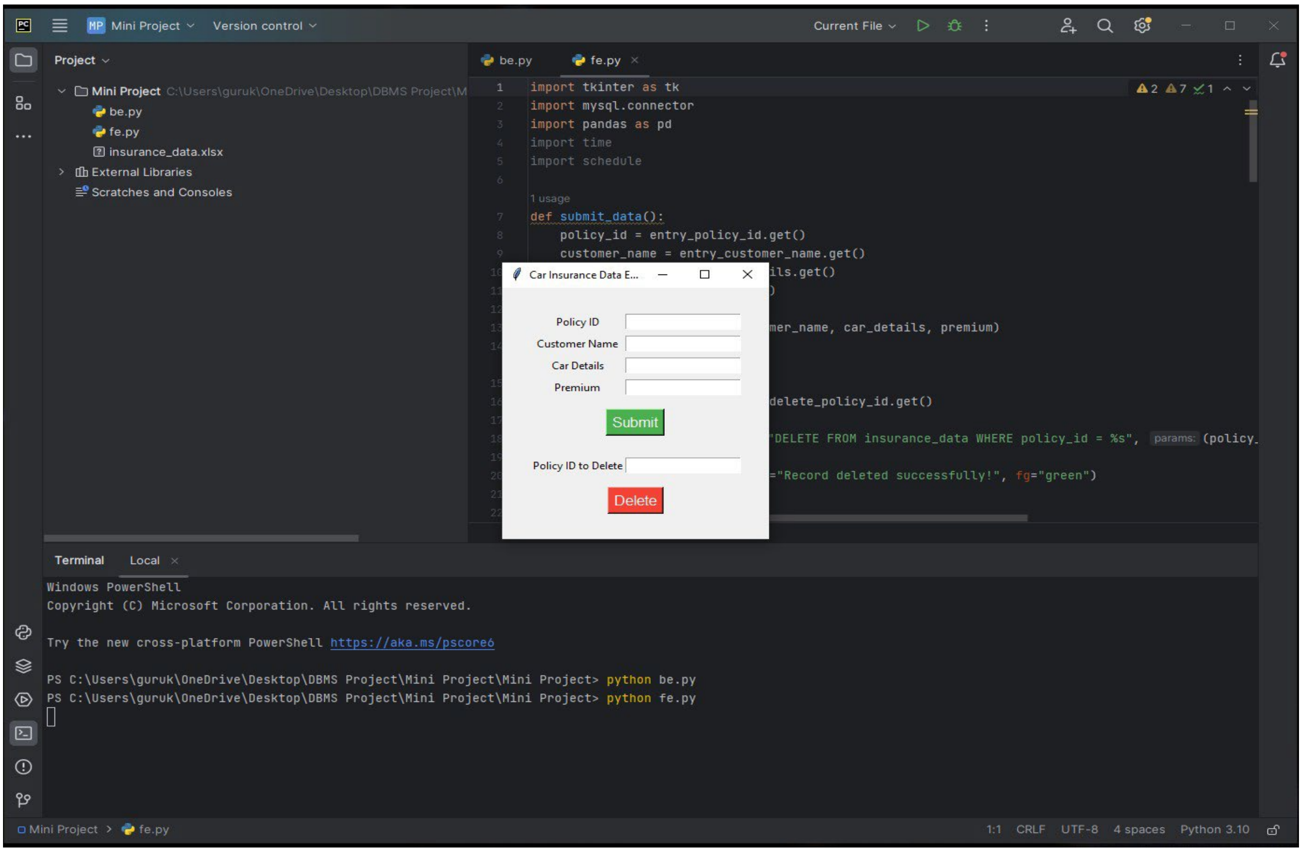
6. Excel File (Data Storage):

- The application can save data to an Excel file. This file format is commonly used for data storage and sharing.

The code combines these technologies and libraries to create a user-friendly application for entering car insurance data, saving it to a MySQL database, and exporting it to an Excel file.

RESULTS

Result Screenshots ….



```
import tkinter as tk
import mysql.connector
import pandas as pd
import time
import schedule

1 usage
def submit_data():
    policy_id = entry_policy_id.get()
    customer_name = entry_customer_name.get()
                    ils.get()
                    )

                    mer_name, car_details, premium)


                    delete_policy_id.get()

                    "DELETE FROM insurance_data WHERE policy_id = %s",  params: (policy_

                    ="Record deleted successfully!", fg="green")
```

Policy ID

Customer Name

Car Details

Premium

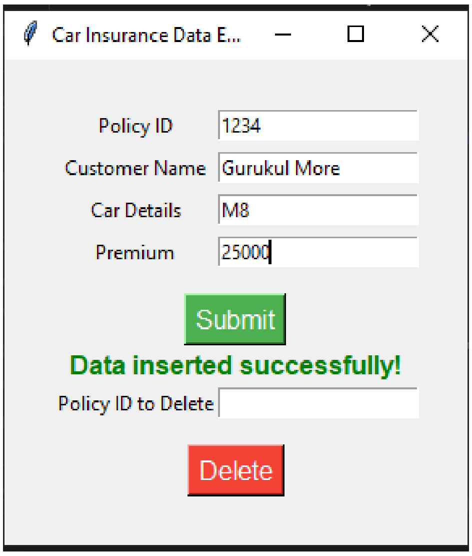
Policy ID to Delete

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\guruk\OneDrive\Desktop\DBMS Project\Mini Project\Mini Project> python be.py
PS C:\Users\guruk\OneDrive\Desktop\DBMS Project\Mini Project\Mini Project> python fe.py
```

INSERTING DATA:



Policy ID: 1234

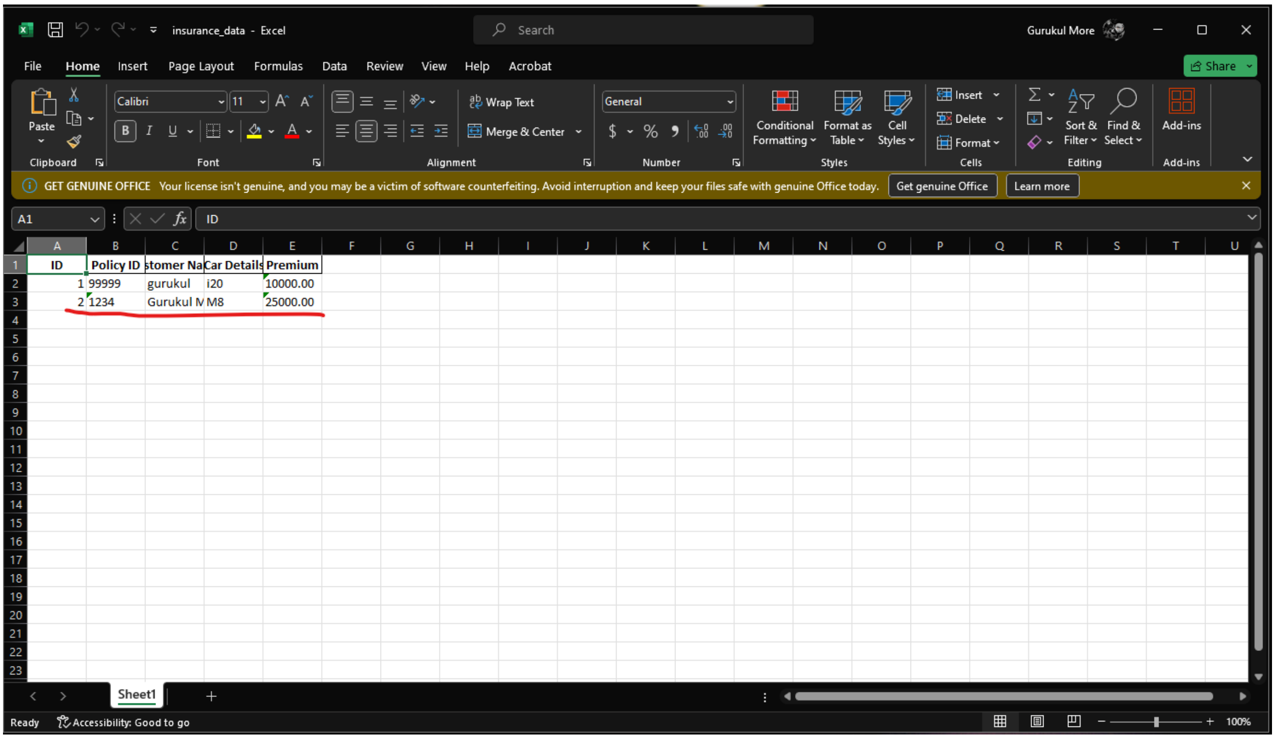
Customer Name: Gurukul More

Car Details: M8

Premium: 25000

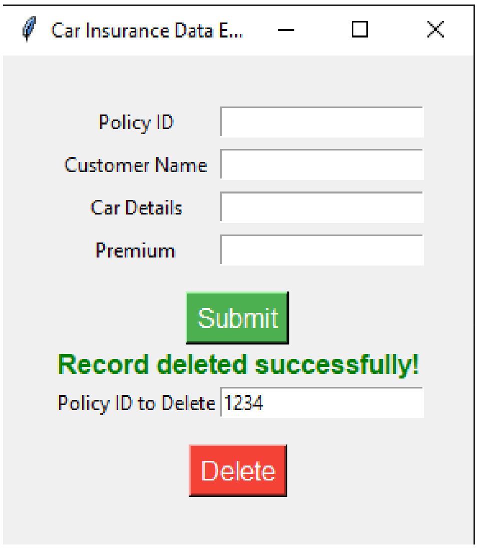
Data inserted successfully!

Policy ID to Delete:



ID	Policy ID	stomer Na	Car Details	Premium
1	99999	gurukul	i20	10000.00
2	1234	Gurukul M	M8	25000.00

DELETING DATA:



Policy ID

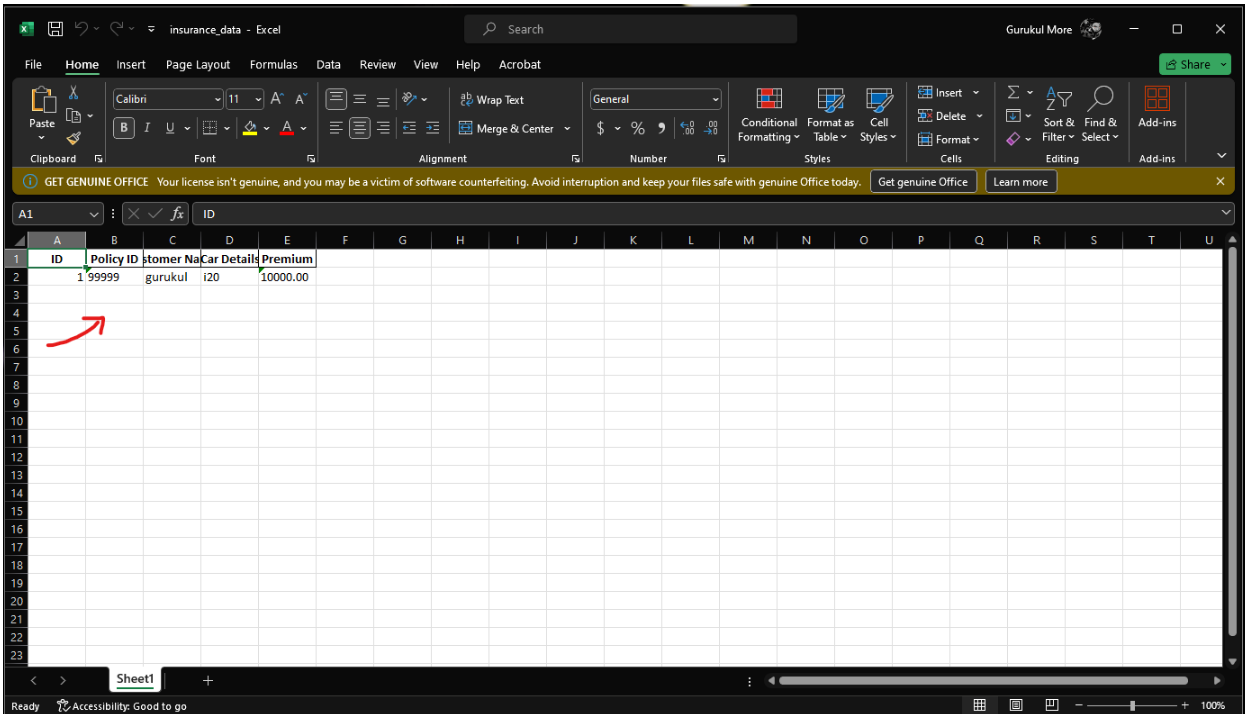
Customer Name

Car Details

Premium

Record deleted successfully!

Policy ID to Delete 1234



GET GENUINE OFFICE Your license isn't genuine, and you may be a victim of software counterfeiting. Avoid interruption and keep your files safe with genuine Office today.

ID	Policy ID	stomer Na	Car Details	Premium
1	99999	gurukul	i20	10000.00

APPLICATIONS

1. Risk Assessment: Insurance companies use historical data to assess the risk associated with insuring a particular vehicle or driver. This helps them determine appropriate premium rates.

2. Claims Processing: Datasets are used to process insurance claims. Claims adjusters can refer to these datasets to verify information related to accidents, damage, and policy coverage.

3. Pricing and Underwriting: Insurers use datasets to set pricing and underwriting guidelines. They analyze data to determine how likely a driver is to file a claim and adjust rates accordingly.

4. Fraud Detection:Data analysis can help identify potentially fraudulent claims. Unusual patterns or anomalies in the data can be flagged for further investigation.

5. Customer Service: Insurance companies can use data to provide better customer service. For example, using customer data to offer personalized discounts or coverage recommendations.

6. Actuarial Analysis: Actuaries use insurance datasets to make predictions about future claims and liabilities. This is essential for the financial stability of insurance companies.

7. Traffic Safety and Accident Analysis: Public agencies and researchers can use insurance data to analyze accident patterns, road safety, and trends in car-related incidents. This can inform policy decisions and safety measures.

8. Telematics and Usage-Based Insurance: Some insurance companies use telematics data (collected from devices in vehicles) to determine rates based on how safely a person drives. This usage-based insurance can be more personalized and cost-effective.

9. Market Research: Researchers and analysts can use car insurance data to study market trends, consumer preferences, and the competitive landscape in the insurance industry.

10. Predictive Analytics: Machine learning models can be applied to these datasets to make predictions about future claims, customer behavior, and other relevant factors.

CONCLUSION

The Car Insurance Company Dataset Entry project successfully provides a user-friendly interface for entering car insurance policy information. The integration of Tkinter and MySQL,along with additional features such as data export to Excel, enhances the functionality and utility of the application. Further improvements could include additional features like data querying and updating.

Also we have successfully developed a robust and user-friendly car insurance company dataset entry system using Python's Tkinter library for creating an intuitive graphical user interface. The application provides an efficient and streamlined way for the insurance company to manage their data effectively.

FUTURE SCOPE

1. Advanced Telematics and IoT Integration: As Internet of Things (IoT) technology continues to advance, insurance companies may collect more real-time data from vehicles, such as driving behavior, vehicle health, and traffic conditions. This data can be used for more accurate risk assessment and personalized pricing.

2. Machine Learning and AI-Based Underwriting: Machine learning and AI algorithms will play a more significant role in underwriting and risk assessment. These technologies can analyze vast datasets to make more accurate predictions and determine pricing based on individual behavior and risk profiles.

3. Usage-Based Insurance (UBI):UBI, which calculates premiums based on individual driving habits, is likely to become more common. Insurance companies may offer more flexible and personalized policies, which can be especially appealing to safe and infrequent drivers.

4. Blockchain for Transparency: Blockchain technology can enhance data security, transparency, and trust in the insurance industry. Smart contracts on a blockchain can streamline claims processing and reduce fraud.

5. Environmental Impact Assessment:With an increasing focus on environmental sustainability, car insurance may include data related to the environmental impact of vehicles. This data can be used to incentivize eco-friendly driving and vehicle choices.

6. Cybersecurity Insurance: With the rise of connected and autonomous vehicles, there will be a growing need for insurance coverage related to cybersecurity and data breaches. Datasets will include information on these emerging risks.

7. Enhanced Customer Experience: Insurance companies will continue to invest in improving the customer experience through data analysis. Chatbots, virtual agents, and AI-driven customer service tools will become more advanced.

8. Regulatory Changes: Changes in regulatory requirements can impact the collection and usage of car insurance data. Companies will need to adapt to new compliance standards, such as those related to data privacy and fairness in pricing.

References

- **Tkinter Documentation:**
  - **Reference:** Tkinter is Python's standard GUI (Graphical User Interface) toolkit. You can refer to the official Python documentation for Tkinter to understand its various modules, classes, and methods.
  - **Explanation:** The official documentation provides detailed information about creating GUI applications using Tkinter. It covers topics like widget options, event handling, and layout management, essential for building the user interface of your application.
  - **Link:** Tkinter Documentation
- **Pandas Documentation:**
  - **Reference:** Pandas is a powerful data manipulation library in Python. Refer to the official Pandas documentation to explore its functions, data structures, and capabilities.
  - **Explanation:** The Pandas documentation offers comprehensive guidance on data manipulation tasks such as data cleaning, transformation, and analysis. It includes explanations of DataFrame operations, data input/output, and handling missing data, which are essential for managing your insurance dataset.
  - **Link:** Pandas Documentation
- **MySQL Documentation:**
  - **Reference:** MySQL is a popular open-source relational database management system. Explore the official MySQL documentation to learn about SQL syntax, database administration, and security features.
  - **Explanation:** The MySQL documentation provides in-depth information on creating and managing databases, tables, and user privileges. Understanding SQL queries and database optimization techniques is crucial for interacting with your MySQL database in the project.
  - **Link:** MySQL Documentation
- **MySQL Tutorial by TutorialsPoint:**
  - MySQL Tutorial by TutorialsPoint

Source Code and Required Files link / GitHub Link

https://github.com/GurukulMore/Car-Insurance-Mysql-Project

Output Video Link

https://www.youtube.com/watch?v=ozKpNGeJ8_c