

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN TOÁN ỨNG DỤNG VÀ TIN HỌC



BÁO CÁO GIỮA KÌ
TRÌNH BÀY BÁO CÁO NHÓM
HỌC PHẦN: CƠ SỞ DỮ LIỆU

Giảng viên hướng dẫn: **ThS. Nguyễn Danh Tú**

Nhóm thực hiện: 20

Thành viên: Nguyễn Văn Triển - 20195934

 Nguyễn Công Hiếu - 20195016

 Nguyễn Hoàng Lương - 20195899

 Hoàng Phạm Thông - 20195922

 Nguyễn Việt Khánh - 20195891

Hà Nội, 08/2021

Mục lục

Đánh giá nhóm.....	1
Lời nói đầu.....	2
PHẦN 1: ĐỀ TÀI NHÓM	3
1.1 Đề tài nhóm	3
1.2 Thông tin csdl.....	3
PHẦN 2: TRUY VẤN DỮ LIỆU	7
2.1 Các câu lệnh truy vấn.....	7
2.2 Thực hiện các công việc để tối ưu thời gian truy vấn và đánh giá	19
□ Index.....	19
□ Partition.....	23
PHẦN 3: CẬP NHẬT DỮ LIỆU	27
3.1 Thủ tục (Procedure) cập nhật dữ liệu	27
3.2 Thực hiện giao dịch (transaction)	42
3.3 Sinh CSDL	43
KẾT LUẬN	50
Tài liệu tham khảo	50

Đánh giá nhóm

1. Đánh giá của nhóm trưởng:

BẢNG ĐÁNH GIÁ THÀNH VIÊN MÔN HỌC : CSDL						
	HỌ VÀ TÊN:	NGUYỄN VĂN TRIỀN				
	LỚP:	K64 – TOÁN TIN 02				
	NHÓM:	20				
STT	Tên thành viên	Làm tốt phần việc được giao	Liên hệ được khi cần	Khả năng đóng góp sáng kiến, ý kiến cho hoạt động nhóm	Sẵn sàng giúp đỡ	Đóng góp chung vào kết quả của nhóm
1	Nguyễn Văn Triển	5	5	5	4	5
2	Nguyễn Công Hiếu	5	5	5	5	5
3	Nguyễn Hoàng Lương	4	4	4	4	4
4	Hoàng Phạm Thông	2	3	2	3	3
5	Nguyễn Việt Khánh	3	2	3	2	2

2. Đánh giá của giảng viên:

Nhóm	Bổ cục											Giao lưu cùng lớp	Điểm cộng, điểm trừ		
	Truy vấn dữ liệu						Cập nhật dữ liệu								
	Thông tin bัน ghi	Đánh giá mức độ chuẩn hóa	Số câu lệnh?	Mức độ khó của truy vấn?	Có tối ưu truy vấn?	Đánh giá hiệu năng?	Số lượng procedure insert/update /delete?	Số nghiệp vụ thực tế (sử dụng transaction)	Tạo CSDL từ câu lệnh truy vấn?	Có cấu trúc? Kết luận?					
	▼	▼	▼	▼	▼	▼	▼	▼	▼	▼	▼	▼	▼		
20	4	3	15	5	Có	Có	30	3	Có	Có	Có	0	2		

Lời nói đầu

Trong cuộc sống hiện đại ngày nay, công nghệ đóng một vai trò hết sức quan trọng. Cùng với sự phát triển của công nghệ, cơ sở dữ liệu (CSDL) được coi là nguồn tài nguyên quý giá “không thể thay thế”. Bởi hệ thống CSDL đảm bảo được khả năng truy xuất của nhiều người dùng trên cùng một tài liệu. Nhiều người có thể sử dụng dữ liệu đó cùng lúc mà không cần phải trải qua các khâu rườm rà phức tạp nhờ việc truy xuất từ các cách khác nhau. Nhờ đó người dùng cũng gặp nhiều thuận lợi hơn trong việc sử dụng, quản lý, truy cập dữ liệu. CSDL còn giúp người dùng tạo lập, cập nhật và khai thác thông tin dễ dàng hơn. Dữ liệu có thể được cập nhật thường xuyên và hoàn toàn không trùng lặp. Với việc sử dụng CSDL, thông tin được lưu trữ có hệ thống hơn, giúp doanh nghiệp dễ dàng hơn trong công tác quản lý và truy xuất thông tin.

Có thể nhận thấy, CSDL mang lại rất nhiều lợi ích. Hiểu được điều đó chúng em đã cố gắng nghiên cứu về nó một cách trực quan, nghiêm túc. Tuy nhiên trong quá trình nghiên cứu tìm hiểu không thể tránh khỏi những thiếu sót. Chúng em xin chân thành cảm ơn thầy Nguyễn Danh Tú đã tận tình chỉ bảo, cũng như có thể có những hướng dẫn cụ thể chi tiết để chúng em có thể hoàn thành bài tập nhóm một cách tốt nhất.

PHẦN 1: ĐỀ TÀI NHÓM

1.1 Đề tài nhóm

Nhóm 20 chúng em nhận được đề tài về Insurance(bảo hiểm), và chúng em nghiên cứu tìm hiểu về lĩnh vực Insurance Car (bảo hiểm xe).

- Bảo hiểm là một hoạt động mà qua đó một cá nhân có quyền được hưởng trợ cấp nhờ vào một khoản đóng góp cho mình hoặc cho người thứ 3 trong trường hợp xảy ra rủi ro.
- Bảo hiểm xe là loại bảo hiểm kết hợp nhiều loại hình bảo hiểm bao gồm bảo hiểm về người, tài sản, hoàng hóa vận chuyển có liên quan.

1.2 Thông tin csdl

1.2.1. Tổng quan

- Nguồn: github
- Dữ liệu được sinh thêm trên: filldb.info

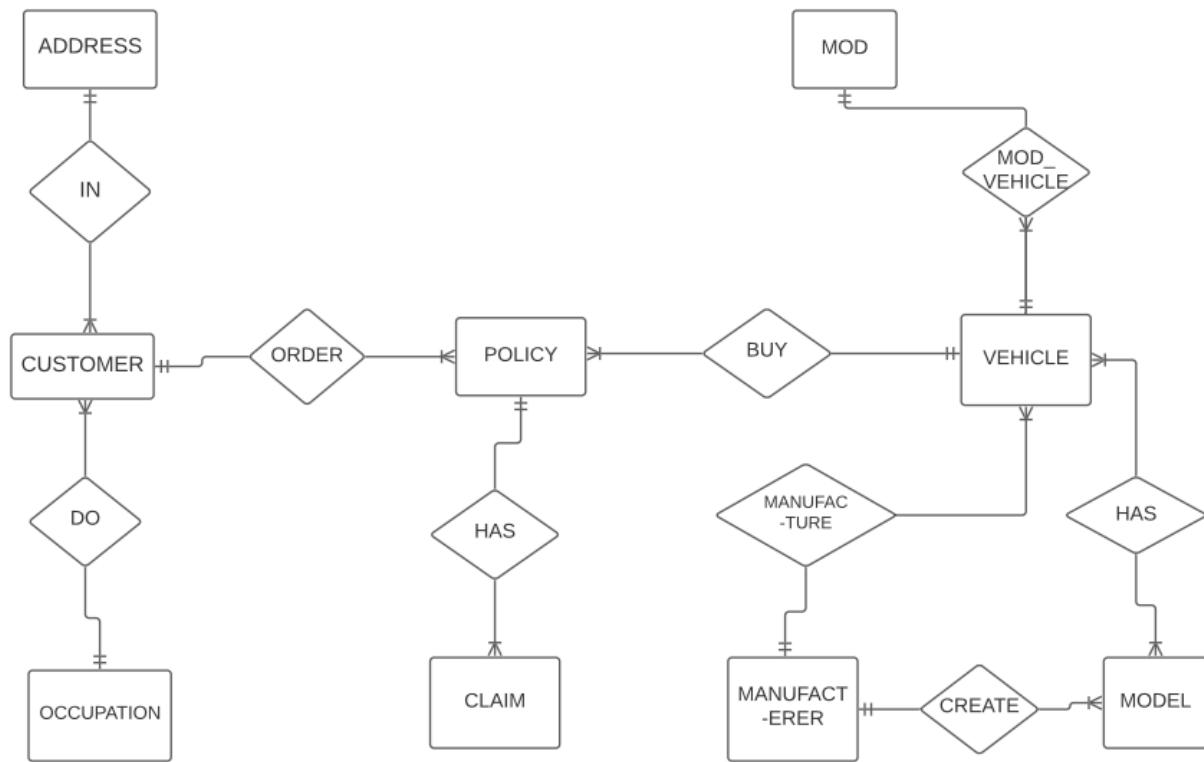
Gồm 10 bảng với độ lớn khoảng 126MB

Schema Details							
Default collation:	utf8mb4_0900_ai_ci						
Default characterset:	utf8mb4						
Table count:	10						
Database size (rough estimate):	125.4 MiB						

Name	Engine	Version	Row Format	Rows	Avg Row Length	Data Length
address	InnoDB	10	Dynamic	97800	48	4.5 MiB
claim	InnoDB	10	Dynamic	99181	153	14.5 MiB
customer	InnoDB	10	Dynamic	178580	161	27.6 MiB
manufacturer	InnoDB	10	Dynamic	69013	53	3.5 MiB
mod	InnoDB	10	Dynamic	10	1638	16.0 KiB
mod_vehicle	InnoDB	10	Dynamic	100320	47	4.5 MiB
model	InnoDB	10	Dynamic	72132	65	4.5 MiB
occupation	InnoDB	10	Dynamic	50	327	16.0 KiB
policy	InnoDB	10	Dynamic	185320	121	21.5 MiB
vehicle	InnoDB	10	Dynamic	82995	107	8.5 MiB

1.2.2. Lược đồ diagram

- Lược đồ E-R



- Lược đồ R-E

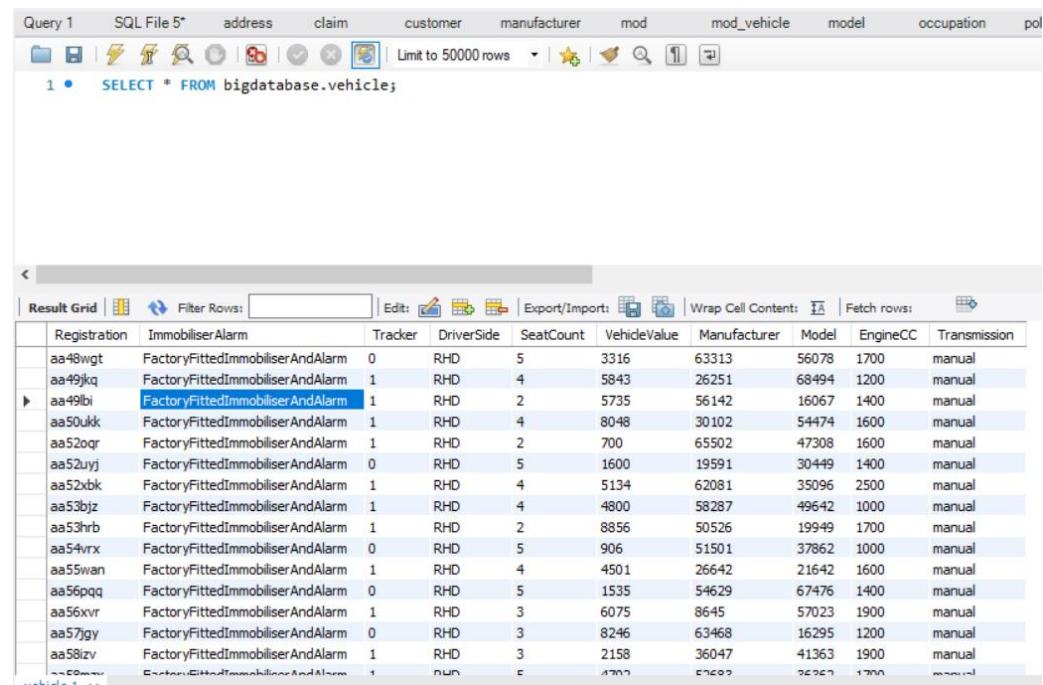


1.2.3. Đánh giá mức độ chuẩn hóa

- Chuẩn 1

Là quan hệ với toàn bộ các miền giá trị của các cột có mặt trong bảng đều chỉ chứa các giá trị nguyên tố.

⇒ Cơ sở dữ liệu đạt chuẩn 1 do chứa các giá trị nguyên tố.



The screenshot shows a MySQL Workbench interface with a query editor containing the command: `SELECT * FROM bigdatabase.vehicle;`. Below the editor is a result grid titled "Result Grid". The grid has columns: Registration, ImmobiliserAlarm, Tracker, DriverSide, SeatCount, VehicleValue, Manufacturer, Model, EngineCC, and Transmission. The data consists of approximately 20 rows of vehicle information, such as registration numbers like aa48wgt, immobiliser alarm types like FactoryFittedImmobiliserAndAlarm, and various vehicle specifications like engine capacity and transmission type.

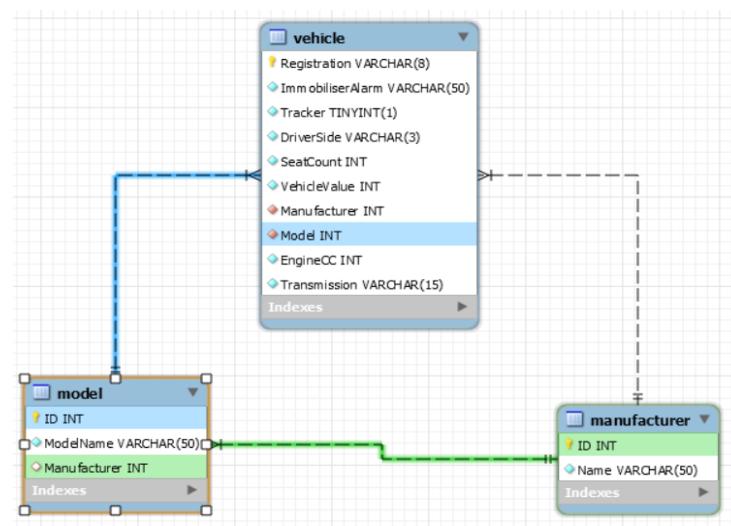
- Chuẩn 2

Là quan hệ chuẩn 1 không có các thuộc tính mà phụ thuộc hàm bộ phận vào khóa chính.

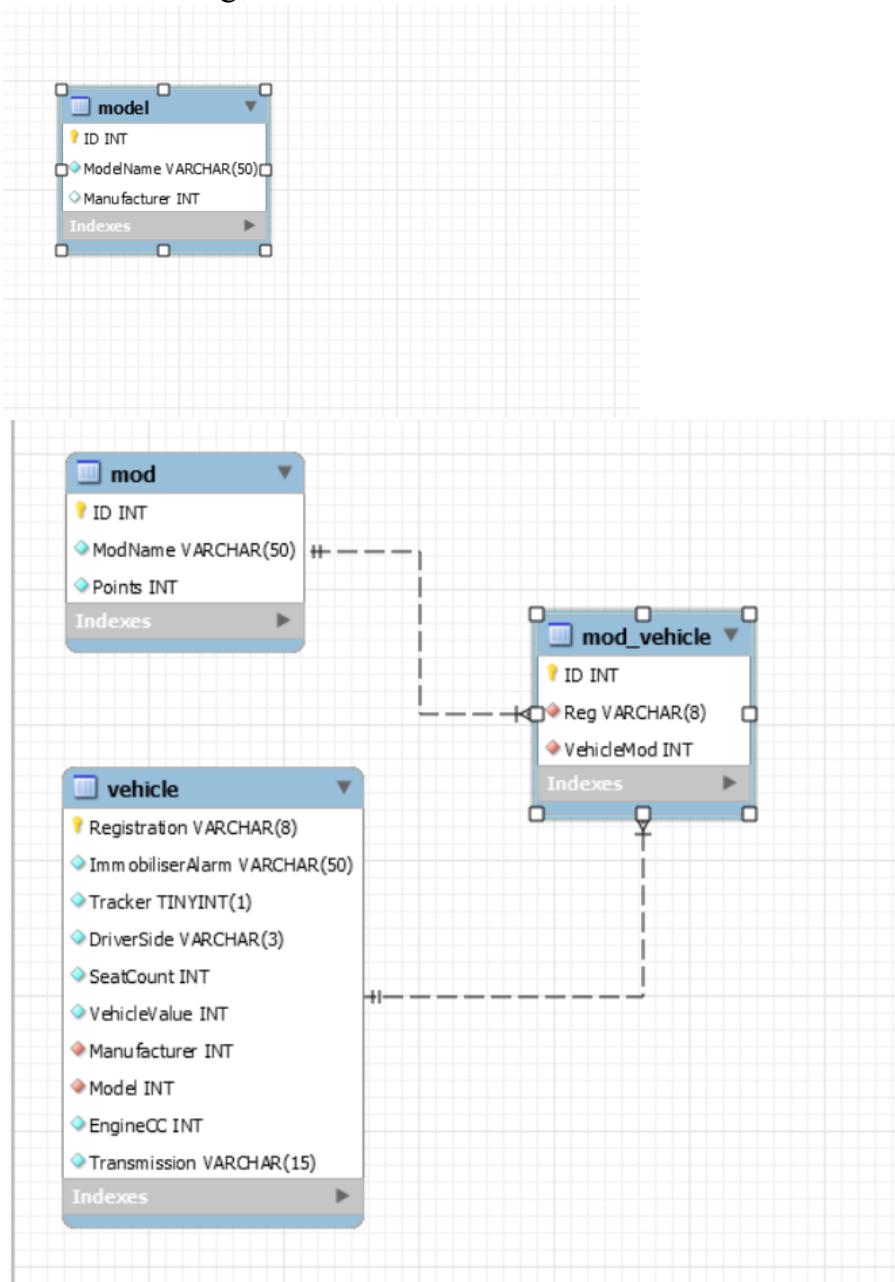
⇒ Cơ sở dữ liệu đã đạt chuẩn 2

- Chuẩn 3

Là quan hệ chuẩn 2 mà không có phụ thuộc hàm bắc cầu vào khóa chính.



- ⇒ Xuất hiện bảng vehicle có sự phụ thuộc hàm bắc cầu thông qua 2 bảng model và manufacturer. Các bảng còn lại đều đạt chuẩn 3.
- Chuẩn BNCF
Là chuẩn 3 mà không có trường hợp thuộc tính khóa phụ thuộc hàm vào thuộc tính không khóa.



- ⇒ Từ các bảng đạt chuẩn 3 ở trên, xuất hiện bảng modvehicle và bảng model. Bảng model có khóa ID phụ thuộc hàm vào thuộc tính manufacturer. Bảng modvehicle có khóa ID phụ thuộc hàm vào thuộc tính Reg và VehicleMod.

Vậy cơ sở dữ liệu trên được đánh giá là đạt chuẩn 2.

PHẦN 2: TRUY VẤN DỮ LIỆU

2.1 Các câu lệnh truy vấn

➤ Truy vấn 1

- Yêu cầu: Lấy thông tin customer có Licence Type là “Lack”, sắp xếp ID tăng dần
- Câu lệnh & Kết quả:

The screenshot shows the MySQL Workbench interface with a query editor and a result grid. The query is:

```

1 • SELECT*
2   FROM customer
3   WHERE LicenceType LIKE '%Lack%'
4   ORDER BY ID ASC
    
```

The result grid displays 46 rows of customer data, filtered by LicenceType like '%Lack%'. The columns include ID, Title, FName, LName, DOB, Homeowner, Dependents, MaritalStatus, Occupation, ClaimsIn5Years, TimeInUsSince, AddressID, LicenceType, LicenceLocation, LicenceNumber, AdditionalQualification, and MedicalC.

➤ Truy vấn 2

- Yêu cầu: Hiển thị danh sách khách hàng có ClaimStatus là “Paid”
- Câu lệnh & Kết quả:

The screenshot shows the MySQL Workbench interface with a query editor and a result grid. The query is:

```

1 • SELECT*
2   FROM claim
3   WHERE ClaimStatus = 'Paid'
4
    
```

The result grid displays 43 rows of claim data, filtered by ClaimStatus = 'Paid'. The columns include ID, PolicyNumber, Description, FaultyParty, AmountPaidOut, DateMade, DatePaid, and ClaimStatus.

➤ Truy vấn 3

- Yêu cầu: Tìm mã bưu điện của khách hàng Ms.Kieran McDermott
- Câu lệnh & Kết quả:

```

1 •  SELECT customer.Title, customer.FName, customer.LName, address.PostCode
2   FROM customer
3   INNER JOIN address
4     ON customer.AddressID = address.ID
5   WHERE customer.Title = 'Ms.' AND customer.FName = 'Kieran' AND customer.LName = 'McDermott'

```

The screenshot shows a MySQL Workbench interface. The SQL editor contains the provided query. The results grid shows one row of data:

	Title	FName	LName	PostCode
▶	Ms.	Kieran	McDermott	uv525ga

➤ Truy vấn 4

- Yêu cầu: Lấy thông tin các khách hàng có giấy phép ở Pháp
- Câu lệnh & Kết quả:

```

1 •  SELECT customer.Title, customer.FName, customer.LName, customer.DOB, occupation.OccupationDescription
2   FROM customer
3   INNER JOIN occupation
4     ON customer.Occupation = occupation.ID
5   WHERE customer.LicenceLocation = 'FR'
6
7

```

The screenshot shows a MySQL Workbench interface. The SQL editor contains the provided query. The results grid shows multiple rows of data:

	Title	FName	LName	DOB	OccupationDescription
▶	Prof.	Mikel	Runolfsdottir	1996-01-10	Photographer
	Dr.	Doris	Buckridge	1978-09-17	Unemployed
	Prof.	Walton	Fahay	1972-10-28	DatabaseArchitect
	Dr.	Cornelius	Simonis	2003-04-15	WebDeveloper
	Miss	Eloy	Goldner	2020-10-14	Fishmonger
	Prof.	Kamille	Terry	1977-12-16	BicycleRepairer
	Prof.	Nichole	Kilback	1974-06-21	Engineer
	Ms.	Sandy	Hansen	1974-02-24	Soldier
	Prof.	Quinton	Prosacco	1999-10-13	Actor
	Miss	Amelie	Kihn	2012-07-23	ComputerProgrammer
	Prof.	Alba	Johnston	1999-06-03	FulltimeStudent
	Prof.	Camille	Fahay	2020-12-18	Lawyer
	Mr.	Izaiah	Graham	1972-02-18	DatabaseArchitect
	Mrs.	Camrvn	Hoeder	2018-03-30	Babysitter

➤ Truy vấn 5

- Yêu cầu: Hiển thị danh sách khách hàng trả tiền theo từng tháng
- Câu lệnh & Kết quả:

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows: |

```

1   SELECT * FROM policy
2   INNER JOIN customer
3   ON policy.CustomerID = customer.ID
4   WHERE PaymentType LIKE '%Monthly%'
5

```

ID	Title	FName	LName	DOB	Homeowner	Dependants	MaritalStatus	Occupation	ClaimsIn5Years	TimeInUkSince	AddressID	LicenceType	LicenceLocation	LicenceNumber	AdditionalQualification	Medic
134643	Prof.	Bo	Ratke	1978-11-11	0	2	engaged	43	0	1989-12-19	42062	Full	IT	554099251	None	No
159965	Prof.	Hilma	Will	1975-11-21	0	2	single	6	1	1984-06-02	78304	Full	IT	691714802	None	No
125604	Dr.	Aurelie	Witting	2018-04-29	1	0	single	30	1	1985-08-25	6784	Full	RU	752414245	None	Yes
110815	Dr.	Katharina	Haag	2020-08-28	1	0	engaged	19	0	1980-03-20	42356	Full	GB	405626263	None	Yes
26166	Mr.	Paxton	Torphy	1981-04-16	0	2	single	10	0	1970-12-03	74983	Lack	FR	163385067	None	Yes
166275	Dr.	Valentine	Wolff	2013-08-14	2	1	single	10	1	1980-10-28	92010	Full	IN	268221131	None	Yes
50131	Mr.	Natalia	Kuhic	2002-10-02	0	2	engaged	20	0	1978-11-30	50512	Full	ES	677051968	None	No
170484	Miss	Ode	Kiehn	1998-03-03	1	2	single	9	0	1970-07-25	98456	Lack	IT	720289461	None	No
43123	Dr.	Grayson	Jones	1991-09-08	1	2	engaged	20	0	1971-08-20	61525	Full	IN	709693716	None	No
79239	Mr.	Edward	Pfannerstill	1993-10-25	2	1	single	22	1	1971-01-25	7137	Lack	PT	14749278	None	Yes
91379	Miss	Reilly	Bayer	1970-10-26	2	1	married	7	1	1980-10-03	53455	Lack	IE	168737475	None	No
41295	Dr.	Wava	Batz	1986-11-23	0	1	engaged	32	2	1975-08-18	27770	Lack	RU	679564130	None	Yes
107640	Mr.	Brandy	Roob	1977-08-27	0	0	married	43	2	1984-03-08	61462	Lack	PT	504927049	None	No
178744	Ms.	Michale	Keeling	1997-07-22	1	0	single	27	1	1978-09-21	67023	Lack	GB	926063197	None	No
893	Mrs.	Octavia	Hahn	1988-09-22	2	0	married	26	2	1986-05-04	359	Full	CN	978819112	None	Yes
125575	Prof.	Eileen	Lynch	2000-03-03	0	2	engaged	29	2	1970-08-28	97708	Lack	US	232105911	None	Yes
144074	Dr.	Marta	McGlynn	1990-09-11	2	0	engaged	5	0	1989-05-09	33930	Lack	GB	754464242	None	Yes
16477	Dr.	Rebekah	Welch	2019-06-06	2	0	engaged	16	1	1975-10-10	96077	Full	MX	395459117	None	Yes
35528	Mrs.	Karolann	Will	2019-06-09	2	2	married	33	1	1985-10-18	80667	Lack	DE	566645937	None	No
152065	Dr.	Peter	Wintheiser	1984-04-17	1	1	engaged	5	0	1984-11-27	25175	Full	DE	295548154	None	Yes
64578	Prof.	Hassie	Koss	1974-03-22	2	2	married	22	2	1971-04-21	84853	Full	IT	520026309	None	No

➤ Truy vấn 6

- Yêu cầu: Truy vấn dữ liệu của bảng Claim liên kết với bảng Policy
- Câu lệnh & Kết quả:

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows: |

```

1   SELECT *
2   FROM claim
3   INNER JOIN policy
4   ON claim.PolicyNumber = policy.ID
5

```

ID	PolicyNumber	Description	FaultyParty	AmountPaidOut	DateMade	DatePaid	ClaimStatus	ID	CustomerID	VehicleReg	VehicleUsage	EstimatedMileage	PolicyStart	Result Grid
1	108563	Reversed into another vehicle in the parking lot...	Noone	900	2022-05-19	2022-12-14	Paid	108563	119129	uh42tb	Commuting Only	8000	2017-07-22	Form Editor
2	165046	Skidded on black ice and hit a lampost destroy...	Client	1700	2022-07-26	2022-10-11	Paid	165046	143771	hm56aqx	SocialOnly	3000	2018-10-24	Field Types
3	188665	Reversed into another vehicle in the parking lot...	Client	1700	2022-07-18	2021-07-05	Pending	188665	140195	lv670zg	SocialAndComuting	5000	2019-11-13	Query Stats
4	71599	Reversed into another vehicle in the parking lot...	Client	200	2022-12-07	2022-12-24	Pending	71599	64202	x19wtf	Commuting Only	4000	2017-06-03	Execution Plan
5	168248	Lost Control over a small hill in the road. Skidde...	Noone	400	2021-10-25	2021-10-28	Paid	168248	35061	nt47fm	Commuting Only	7000	2019-11-27	Read Only
6	152141	Scraped side of vehicle against a post in a park...	3rdPartyUnknown	500	2021-09-02	2021-08-30	Pending	152141	179115	ka54tvv	SocialOnly	5000	2020-05-05	Activate Windows
7	169577	Lost Control over a small hill in the road. Skidde...	Client	1700	2022-02-16	2022-02-18	Pending	169577	103305	ww55ed	SocialOnly	7000	2019-07-22	Go to Settings to activate Windows
8	33450	Reversed into another vehicle in the parking lot...	3rdPartyUnknown	100	2022-11-16	2022-08-09	Paid	33450	179689	vn15sztk	SocialOnly	9000	2017-06-21	2017-01-03
9	114789	Skidded on black ice and hit a lampost destroy...	3rdPartyUnknown	200	2021-09-23	2021-06-07	Paid	114789	127452	xq07qgc	SocialOnly	8000	2017-07-28	2018-02-26
10	34695	Skidded on black ice and hit a lampost destroy...	Client	1000	2022-02-05	2022-06-17	Pending	34695	179272	sd13dsy	SocialAndComuting	8000	2018-08-25	2018-12-07
11	36797	Lost control and hit a roundabout. Destroying f...	Client	800	2022-08-03	2022-12-15	Paid	36797	89508	qr03jd	Commuting Only	4000	2019-11-05	2019-12-14
12	195921	Reversed into another vehicle in the parking lot...	Client	0	2022-01-17	2022-02-08	Paid	195921	97782	dy76on	Commuting Only	7000	2020-07-07	2020-07-07
13	121381	Reversed into another vehicle in the parking lot...	3rdPartyUnknown	1500	2023-03-03	2022-02-07	Pending	121381	32432	eu88jld	SocialAndComuting	7000	2019-11-05	2019-11-05
14	104268	Lost control and hit a roundabout. Destroying f...	Noone	3000	2022-05-18	2022-01-24	Paid	104268	91583	uj75zer	SocialAndComuting	6000	2020-12-14	2020-07-07
15	106326	Wing mirror smashed off during the night while ...	Noone	300	2022-12-26	2022-04-24	Paid	106326	112603	kv97bf	SocialOnly	7000	2020-07-07	2020-07-07
16	76513	Wing mirror smashed off during the night while ...	Client	1200	2022-01-13	2022-08-11	Paid	76513	92017	gg15xgs	SocialAndComuting	6000	2018-08-25	2018-08-25
17	177164	lost control and hit a roundabout. Destroying f...	3rdPartyUnknown	100	2022-02-21	2023-03-28	Pending	177164	47408	ml71uf	SocialAndComuting	8000	2020-06-26	2020-06-26
18	60770	Scraped side of vehicle against a post in a park...	Noone	100	2021-10-20	2021-09-06	Pending	60770	156000	vv23mk	Commuting Only	8000	2017-07-07	2017-07-07

➤ Truy vấn 7

- Yêu cầu: Truy vấn dữ liệu của bảng Customer với bảng Occupation theo thứ tự sắp xếp giảm dần ID
- Câu lệnh & Kết quả:

The screenshot shows the MySQL Workbench interface. In the top-left pane, there is a SQL editor window titled "SQL File 4*" containing the following SQL code:

```

1 • SELECT *
2   FROM customer
3   INNER JOIN occupation
4     ON customer.Occupation = occupation.ID
5   ORDER BY occupation.ID DESC
    
```

In the bottom-right pane, there is a large "Result Grid" displaying the results of the query. The grid has 52 rows and 17 columns, corresponding to the fields listed in the query. The columns include: ID, Title, FName, LName, DOB, Homeowner, Dependents, MaritalStatus, Occupation, ClaimsIn5Years, TimeInUKSince, AddressID, LicenceType, LicenceLocation, LicenceNumber, AdditionalQualification, and MedicC.

➤ Truy vấn 8

- Yêu cầu: Truy vấn dữ liệu của liên kết 3 bảng Customer, Occupation và Address theo thứ tự tăng dần ID
- Câu lệnh & Kết quả:

The screenshot shows the MySQL Workbench interface. In the top-left pane, there is a SQL editor window titled "SQL File 3*" containing the following SQL code:

```

1 • SELECT * FROM ((customer
2   INNER JOIN occupation
3     ON customer.Occupation = occupation.ID)
4   INNER JOIN address
5     ON customer.AddressID = address.ID)
6   ORDER BY address.ID ASC
    
```

In the bottom-right pane, there is a large "Result Grid" displaying the results of the query. The grid has 52 rows and 17 columns, corresponding to the fields listed in the query. The columns include: ID, Title, FName, LName, DOB, Homeowner, Dependents, MaritalStatus, Occupation, ClaimsIn5Years, TimeInUKSince, AddressID, LicenceType, LicenceLocation, LicenceNumber, AdditionalQualification, and MedicC.

➤ Truy vấn 9

- Yêu cầu: Truy vấn Tên và Ngày Sinh của khách hàng theo liên kết hai bảng Customer và Occupation theo trình tự ID tăng dần
- Câu lệnh & Kết quả:

The screenshot shows the MySQL Workbench interface. In the SQL pane, there is a query:

```

1 SELECT FName, LName ,DOB
2 FROM customer
3 INNER JOIN occupation
4 ON customer.Occupation = occupation.ID
5 ORDER BY occupation.ID ASC

```

The results are displayed in the Result Grid:

FName	LName	DOB
Emil	Barrows	2002-09-17
Adela	O'Kon	2007-10-08
Brennon	Rowe	1988-03-13
Lonzo	Crist	2008-01-10
Guadalupe	Cormier	1977-08-12
Gabriel	Romaguera	1989-01-11
Marcella	Daniel	2018-11-12
Kaley	Schmitt	1986-06-14
Kareem	Nitzsche	1993-02-06
Abigayle	Boyer	1979-03-25

Below the grid, the message "1000 row(s) returned" is shown. The status bar indicates "Duration / Fetch 0.859 sec / 0.000 sec".

➤ Truy vấn 10

- Yêu cầu: Truy vấn khách hàng có linece loại full theo liên kết của bảng Customer với Occupation
- Câu lệnh & Kết quả:

The screenshot shows the MySQL Workbench interface. In the SQL pane, there is a query:

```

1 • SELECT * FROM customer
2 INNER JOIN occupation
3 ON customer.Occupation = occupation.ID
4 WHERE customer.LicenceType ='Full'

```

The results are displayed in the Result Grid:

ID	Title	FName	LName	DOB	Homeowner	Dependants	MaritalStatus	Occupation	ClaimsIn5Years	TimeInRiskSince	AddressID	LicenceType	LicenceLocation	LicenceNumber	AdditionalQualification	Medi
2	Mr.	Kassandra	Konopelski	2014-05-21	0	2	single	43	1	1987-03-26	11782	Full	IN	760910580	None	No
4	Mr.	Finn	Sauer	1974-04-29	0	1	married	3	1	1980-07-27	48352	Full	GB	102020722	None	Yes
5	Prof.	Mikel	Runolfsdottir	1996-01-10	2	1	married	13	2	1989-08-26	71216	Full	FR	141243881	None	Yes
7	Prof.	Tia	Shields	1987-08-06	2	1	married	20	2	1990-07-16	5449	Full	RU	433919659	None	Yes
9	Prof.	Jairo	Jacobs	2008-01-19	1	2	engaged	26	2	1983-10-11	9373	Full	IN	784051166	None	No
10	Dr.	Flavie	Abshire	2000-08-26	0	2	engaged	6	0	1981-12-10	85799	Full	CN	546356314	None	Yes
11	Mrs.	Adalberto	Ric	2010-06-17	2	2	married	6	1	1970-11-13	11293	Full	MX	839854515	None	No
12	Dr.	Joany	Rutche	1988-08-09	0	2	married	45	1	1971-07-14	62784	Full	US	392163561	None	No
14	Mr.	Antonietta	Krajcik	1986-03-28	2	2	single	26	1	1971-12-13	39362	Full	US	730316906	None	Yes
15	Prof.	Veronica	Heathcot	2010-03-22	0	2	engaged	35	1	1986-02-27	61059	Full	RU	253388523	None	Yes
16	Mr.	Leslie	Schriner	1987-01-08	0	2	married	44	0	1985-04-19	55584	Full	IT	118139151	None	No
17	Mr.	Vivien	Borer	2011-12-14	0	0	engaged	33	2	1982-09-29	70738	Full	US	760612545	None	No
19	Mr.	Jose	Graham	1978-11-22	0	0	engaged	29	1	1990-05-14	71478	Full	MX	815261994	None	No
20	Miss	Crystel	Weber	1986-08-10	2	1	engaged	3	1	1974-09-29	82168	Full	MX	351920000	None	No
22	Ms.	Alf	White	2005-12-06	0	0	married	50	0	1987-07-27	82578	Full	IT	95297269	None	No

Below the grid, the message "Query Completed" is shown. The status bar indicates "Duration / Fetch 0.859 sec / 0.000 sec".

➤ Truy vấn 11

- Yêu cầu: Tìm ra đại diện của hộ gia đình đang trong thời hạn hưởng chính sách bảo hiểm. Hộ gia đình được xác định bởi nhóm các khách hàng có cùng AddressId(số lượng lớn hơn 1).
- Câu lệnh:

```

SELECT
    CustomerID,
    Title,
    FName AS FirstName,
    LName AS LastName,
    EmailAddress,
    TelephoneNumber,
    VehicleReg,
    HouseNumber,
    PostCode,
    Price AS 'Price($)'
From tbl_customer
LEFT JOIN tbl_policy ON tbl_customer.ID =
tbl_policy.CustomerID
LEFT JOIN tbl_address ON tbl_customer.AddressID =
tbl_address.ID
WHERE GETDATE() BETWEEN PolicyStartDate AND
PolicyEndDate
AND AddressID IN (
    SELECT tbl_customer.AddressID
    FROM tbl_customer
    GROUP BY AddressID
    HAVING COUNT(AddressID) > 1
)
ORDER BY CustomerID;

```

- Kết quả:

	CustomerID	Title	FirstName	LastName	EmailAddress	TelephoneNumber	VehicleReg	HouseNumber	PostCode	Price(\$)
▶	3	Mr.	Akeem	Graham	viola85@example.org	1-817-384-1168	uu26dpj	196	pz728ly	1138
	5	Prof.	Mikel	Runolfsdottir	webster61@example.com	(149)298-5357	xp90ggv	34	bx986eh	948
	9	Prof.	Jairo	Jacobs	yhoeger@example.net	08155297980	nv85bbi	132	np709se	1824
	10	Dr.	Flavie	Abshire	mathias.simonis@example.org	611.954.2066	np62ivu	60	tk538lq	503
	11	Mrs.	Adalberto	Rice	daniella24@example.org	433-423-4870x31	qu93ars	44	jl058yb	1173
	12	Dr.	Joany	Ritchie	lockman.tyrell@example.net	992.066.8610x64	pp81xfi	137	nv916ms	567
	14	Mr.	Antonietta	Krajcik	gutmann.maurine@example.com	115.818.1089x63	qn56rvl	10	jq406tm	1770
	16	Mr.	Leslie	Schinner	earline89@example.com	149-063-7331	kt25jte	85	ll582aa	827
	16	Mr.	Leslie	Schinner	earline89@example.com	149-063-7331	yw61suu	85	ll582aa	1520
	17	Mr.	Vivien	Borer	horacio.schimmel@example.org	(736)023-9209	ny27eop	114	uq415gs	715
	18	Ms.	Gisselle	Erdman	gkessler@example.net	1-119-754-7264x	lb04eat	73	wg628mh	1241
	19	Mr.	Jose	Graham	virginia.prohaska@example.com	(258)486-0209	uq28wrc	65	cz663oa	803
	19	Mr.	Jose	Graham	virginia.prohaska@example.com	(258)486-0209	yp05iqc	65	cz663oa	812
	22	Ms.	Alf	White	mckenzie51@example.org	156.475.0411	wz55dja	144	gz900ol	1880
	22	Ms.	Alf	White	mckenzie51@example.org	156.475.0411	fb08lvi	144	gz900ol	1118
	26	Dr.	Lionel	Harber	fbalistreri@example.net	484-121-5559x99	au66mqz	193	rk725bn	1360

Result 1 ×

➤ Truy vấn 12

- Yêu cầu: Lấy ra danh sách những chính sách bảo hiểm có cùng hiệu lực ở thời điểm hiện tại của từng khách hàng.
- Câu lệnh:

```
SELECT
```

```

x.pol1 AS PolicyID,
x.cus1 AS CustomerID,
x.pol2 AS PolicyID2,
x.cus2 AS CustomerID2
FROM(
SELECT
    P1.ID AS pol1,
    P1.CustomerID AS cus1,
    P1.PolicyStartDate,
    P1.PolicyEndDate,
    P2.CustomerID AS cus2,
    P2.ID AS pol2
FROM tbl_policy AS P1, tbl_policy AS P2
WHERE P1.ID < P2.ID
    AND P1.PolicyStartDate BETWEEN P2.PolicyStartDate AND
P2.PolicyEndDate
    OR P1.PolicyEndDate BETWEEN P2.PolicyStartDate AND
P2.PolicyEndDate
)AS X
WHERE pol1 <> pol2
AND cus1 = cus2
AND GETDATE() BETWEEN X.PolicyStartDate AND
X.PolicyEndDate;

```

- Kết quả:

	PolicyID	CustomerID	PolicyID2	CustomerID2
▶	9	179414	79626	179414
	13	48509	196330	48509
	16	90323	145509	90323
	17	89042	121298	89042
	21	40437	185231	40437
	33	6626	106719	6626
	33	6626	111184	6626
	33	6626	182867	6626
	35	171765	10569	171765
	36	134643	29335	134643
	39	36968	164670	36968
	40	154966	85497	154966

Result 2 ×

- Truy vấn 13
 - Yêu cầu: Tính chi phí bảo hiểm của tất cả các chính sách.
 - Câu lệnh:

USE nguyenhieu;

```

SELECT DISTINCT
    PolicyID,
    BasePrice,
    Metric,
    round(((Baseprice/100*metric)/100)*(PaymentTypePercentageChange(%)+100),2)
    as 'totalPrice($)'

FROM (
    SELECT DISTINCT
        PolicyID,
        (x.VehicleValue / 100) * 25 AS BasePrice,
        (EngineClass + OccupationClass + AgeClass + VehicleModClass +
        MedicalConditionClass)+100 AS Metric,
        PaymentTypePercentageChange(%)

    FROM (
        SELECT tbl_policy.ID AS PolicyID, Fname, VehicleReg,
        VehicleValue,
        CASE
            WHEN EngineCC BETWEEN 1000 AND 1199 THEN 0
            WHEN EngineCC BETWEEN 1200 AND 1399 THEN 10
            WHEN EngineCC BETWEEN 1400 AND 1599 THEN 20
            WHEN EngineCC BETWEEN 1600 AND 1799 THEN 30
            WHEN EngineCC BETWEEN 1800 AND 1999 THEN 40
            WHEN EngineCC BETWEEN 2000 AND 2199 THEN 50
            WHEN EngineCC BETWEEN 2200 AND 2399 THEN 60
            WHEN EngineCC BETWEEN 2400 AND 2599 THEN 70
            WHEN EngineCC BETWEEN 2600 AND 2799 THEN 80
            WHEN EngineCC BETWEEN 2800 AND 2999 THEN 90
            WHEN EngineCC BETWEEN 3000 AND 3199 THEN

```

```

WHEN EngineCC BETWEEN 3200 AND 3399 THEN
110
WHEN EngineCC BETWEEN 3400 AND 3599 THEN
120
WHEN EngineCC BETWEEN 3600 AND 3799 THEN
130
WHEN EngineCC BETWEEN 3800 AND 3999 THEN
140
ELSE 200
END AS EngineClass,
CASE
    WHEN FLOOR(DATEDIFF(DAY, PolicyStartDate,
DOB) / 365) <= 16 THEN 30
        WHEN FLOOR(DATEDIFF(DAY, PolicyStartDate,
DOB) / 365) BETWEEN 17 AND 19 THEN 30
            WHEN FLOOR(DATEDIFF(DAY, PolicyStartDate,
DOB) / 365) BETWEEN 20 AND 25 THEN 20
                WHEN FLOOR(DATEDIFF(DAY, PolicyStartDate,
DOB) / 365) BETWEEN 26 AND 29 THEN 10
                    WHEN FLOOR(DATEDIFF(DAY, PolicyStartDate,
DOB) / 365) BETWEEN 30 AND 35 THEN 0
                    WHEN FLOOR(DATEDIFF(DAY, PolicyStartDate,
DOB) / 365) BETWEEN 35 AND 39 THEN -10
                    WHEN FLOOR(DATEDIFF(DAY, PolicyStartDate,
DOB) / 365) BETWEEN 40 AND 49 THEN -20
                    WHEN FLOOR(DATEDIFF(DAY, PolicyStartDate,
DOB) / 365) BETWEEN 50 AND 59 THEN -30
                    WHEN FLOOR(DATEDIFF(DAY, PolicyStartDate,
DOB) / 365) BETWEEN 60 AND 90 THEN 20
                ELSE 0
            END AS AgeClass,
            Modifier AS OccupationClass,
            CASE

```

```

WHEN SUM(Points) > 1 THEN SUM(Points)
ELSE 0
END AS VehicleModClass,
CASE
    WHEN MedicalCondition = 'No' THEN 0
    WHEN MedicalCondition = 'Yes' THEN 10
    ELSE 20
END AS MedicalConditionClass,
CASE
    WHEN PaymentType = 'Annual' THEN 0
    WHEN PaymentType = 'Monthly' THEN 20
    ELSE 10
END AS 'PaymentTypePercentageChange(%)'
FROM tbl_vehicle
JOIN tbl_policy ON tbl_vehicle.Registration =
tbl_policy.VehicleReg
JOIN tbl_customer ON tbl_policy.CustomerID =
tbl_customer.ID
JOIN tbl_occupation ON tbl_customer.Occupation =
tbl_occupation.ID
LEFT JOIN tbl_mod_vehicle ON tbl_vehicle.Registration =
tbl_mod_vehicle.Reg
LEFT JOIN tbl_mod ON tbl_mod_vehicle.VehicleMod =
tbl_mod.ID
GROUP BY PolicyID
) AS X
) AS Y
WHERE PolicyID BETWEEN 10000 AND 12000;
• Kết quả:

```

	PolicyID	BasePrice	Metric	totalPrice(\$)
▶	10000	297.7500	180	589.55
	10001	147.2500	141	207.62
	10002	1423.5000	127	1988.63
	10003	948.5000	204	2128.43
	10004	1489.5000	219	3262.01
	10005	2097.5000	206	4752.94
	10006	787.5000	184	1449.00
	10007	2033.0000	226	5054.04
	10008	219.7500	150	362.59
	10009	1843.7500	187	3447.81
	10010	1838.7500	163	3296.88
	10011	1776.0000	162	3164.83

Result 3 ×

➤ Truy vấn 14

- Yêu cầu: Nhóm theo các năm, số tiền phải trả cho khách hàng và lợi nhuận.
- Câu lệnh:

SELECT

sales.FinancialYear,

sales.Gross,

CASE

WHEN expenses.expense IS NULL THEN 0

ELSE expenses.expense

END AS AmountSpentOnClaims,

CASE

WHEN sales.gross - expenses.expense IS NULL THEN sales.gross

ELSE sales.gross - expenses.expense

END AS 'Profit/Loss (\$)'

FROM (

SELECT

CASE

WHEN MONTH(tbl_policy.PolicyStartDate) >= 6

THEN concat(YEAR(tbl_policy.PolicyStartDate), ' to ',

YEAR(tbl_policy.PolicyStartDate) + 1)

```

        ELSE concat(YEAR(tbl_policy.PolicyStartDate) - 1, ' to ',
YEAR(tbl_policy.PolicyStartDate))

        END AS FinancialYear,
        SUM(tbl_policy.Price) AS Gross

        FROM tbl_policy
        GROUP BY FinancialYear
    ) AS sales

LEFT JOIN (
    SELECT
        CASE
            WHEN MONTH(tbl_claim.DatePaid) >= 6
            THEN CONCAT(YEAR(tbl_claim.DatePaid), ' to '
            ',YEAR(tbl_claim.DatePaid) + 1)
            ELSE CONCAT(YEAR(tbl_claim.DatePaid)-1, ' to ',
YEAR(tbl_claim.DatePaid))
        END AS FinancialYear,
        SUM(tbl_claim.AmountPaidOut) AS expense

        FROM tbl_claim
        GROUP BY FinancialYear
        HAVING FinancialYear IS NOT NULL
    ) AS expenses ON expenses.FinancialYear = sales.FinancialYear;

```

- Kết quả:

	FinancialYear	Gross	AmountSpentOnClaims	Profit/Loss (\$)
▶	2018 to 2019	49925861	0	49925861
	2020 to 2021	46472570	4274600	42197970
	2016 to 2017	49929158	0	49929158
	2017 to 2018	50158635	0	50158635
	2019 to 2020	50033904	0	50033904
	2015 to 2016	3509966	0	3509966

➤ Truy vấn 15

- Yêu cầu: Số ngày còn lại cho đến hạn cuối của chính sách bảo hiểm của mỗi khách hàng.
- Câu lệnh:

SELECT

```
Title,
FName AS 'First Name',
LName AS 'Last Name',
EmailAddress,
TelephoneNumber,
    CONVERT(varchar, GETDATE(), 105) AS 'Today Date',
    CONVERT(varchar, PolicyEndDate, 105) AS 'Policy End Date',
    DATEDIFF(DAY, PolicyEndDate, GETDATE()) AS 'Days left',
tbl_policy.ID AS PolicyID,
VehicleReg
FROM tbl_customer
LEFT JOIN tbl_policy ON tbl_customer.ID = tbl_policy.CustomerID
WHERE PolicyEndDate BETWEEN GETDATE() AND DATEADD(month, 30,
GETDATE());
```

- Kết quả:

	Title	First Name	Last Name	EmailAddress	TelephoneNumber	Today Date	Policy End Date	Days left	PolicyID	VehicleReg
▶	Dr.	Danielle	King	gregg29@example.com	1-273-979-5717	17/08/2021	15/09/2021	29	28	ye31tev
	Prof.	Howell	Runolfsson	misty08@example.org	1-969-035-2747x	17/08/2021	09/09/2021	23	31	kt62udm
	Dr.	Sabrina	Labadie	upfeffer@example.org	+83(8)034555879	17/08/2021	28/08/2021	11	114	wq63mbz
	Prof.	Damian	Hermann	gutmann.shemar@example.com	+11(8)798858297	17/08/2021	29/08/2021	12	169	zd96dat
	Miss	Oral	VonRueden	luettgen.anahi@example.org	(198)543-2353x8	17/08/2021	30/08/2021	13	212	va00snc
	Mrs.	Thalia	Rowe	gertrude20@example.com	834.787.9069	17/08/2021	03/09/2021	17	238	ib14tyq
	Dr.	Caroline	Shields	pbashirian@example.org	1-925-953-0023	17/08/2021	05/09/2021	19	269	jv67xi
	Prof.	Zita	Little	tania97@example.com	+29(6)569748969	17/08/2021	01/09/2021	15	282	pt59cto
	Dr.	Gunner	Gaylord	oceane69@example.org	05783992586	17/08/2021	15/09/2021	29	331	hn52fw
	Dr.	Effie	Johnston	rempel.julie@example.net	676-305-2516x28	17/08/2021	27/08/2021	10	381	tu29wj
	Prof.	Marie	Funk	lorenza01@example.org	421.387.0163x57	17/08/2021	05/09/2021	19	433	zp65psc
	Dr.	Reginald	Torr	randall00@example.net	1-772-151-8607	17/08/2021	02/09/2021	17	514	mmar31knk

2.2 Thực hiện các công việc để tối ưu thời gian truy vấn và đánh giá

- Index
 - Câu lệnh truy vấn

SELECT

```
tbl_policy.ID AS PolicyID,
Fname,
VehicleReg,
```

VehicleValue,

CASE

WHEN EngineCC BETWEEN 1000 AND 1199 THEN 0
 WHEN EngineCC BETWEEN 1200 AND 1399 THEN 10
 WHEN EngineCC BETWEEN 1400 AND 1599 THEN 20
 WHEN EngineCC BETWEEN 1600 AND 1799 THEN 30
 WHEN EngineCC BETWEEN 1800 AND 1999 THEN 40
 WHEN EngineCC BETWEEN 2000 AND 2199 THEN 50
 WHEN EngineCC BETWEEN 2200 AND 2399 THEN 60
 WHEN EngineCC BETWEEN 2400 AND 2599 THEN 70
 WHEN EngineCC BETWEEN 2600 AND 2799 THEN 80
 WHEN EngineCC BETWEEN 2800 AND 2999 THEN 90
 WHEN EngineCC BETWEEN 3000 AND 3199 THEN 100
 WHEN EngineCC BETWEEN 3200 AND 3399 THEN 110
 WHEN EngineCC BETWEEN 3400 AND 3599 THEN 120
 WHEN EngineCC BETWEEN 3600 AND 3799 THEN 130
 WHEN EngineCC BETWEEN 3800 AND 3999 THEN 140
 ELSE 200

END AS EngineClass,

CASE

WHEN FLOOR(DATEDIFF(DAY, PolicyStartDate, DOB) / 365) <= 16 THEN 30

WHEN FLOOR(DATEDIFF(DAY, PolicyStartDate, DOB) / 365) BETWEEN 17 AND 19 THEN 30

WHEN FLOOR(DATEDIFF(DAY, PolicyStartDate, DOB) / 365) BETWEEN 20 AND 25 THEN 20

WHEN FLOOR(DATEDIFF(DAY, PolicyStartDate, DOB) / 365) BETWEEN 26 AND 29 THEN 10

WHEN FLOOR(DATEDIFF(DAY, PolicyStartDate, DOB) / 365) BETWEEN 30 AND 35 THEN 0

WHEN FLOOR(DATEDIFF(DAY, PolicyStartDate, DOB) / 365)
BETWEEN 35 AND 39 THEN -10

WHEN FLOOR(DATEDIFF(DAY, PolicyStartDate, DOB) / 365)
BETWEEN 40 AND 49 THEN -20

WHEN FLOOR(DATEDIFF(DAY, PolicyStartDate, DOB) / 365)
BETWEEN 50 AND 59 THEN -30

WHEN FLOOR(DATEDIFF(DAY, PolicyStartDate, DOB) / 365)
BETWEEN 60 AND 90 THEN 20

ELSE 0

END AS AgeClass,

Modifier AS OccupationClass,

CASE

WHEN SUM(Points) > 1 THEN SUM(Points)

ELSE 0

END AS VehicleModClass,

CASE

WHEN MedicalCondition = 'No' THEN 0

WHEN MedicalCondition = 'Yes' THEN 10

ELSE 20

END AS MedicalConditionClass,

CASE

WHEN PaymentType = 'Annual' THEN 0

WHEN PaymentType = 'Monthly' THEN 20

ELSE 10

END AS 'PaymentTypePercentageChange(%)'

FROM tbl_vehicle

JOIN tbl_policy ON tbl_vehicle.Registration = tbl_policy.VehicleReg

JOIN tbl_customer ON tbl_policy.CustomerID = tbl_customer.ID

JOIN tbl_occupation ON tbl_customer.Occupation = tbl_occupation.ID

LEFT JOIN tbl_mod_vehicle ON tbl_vehicle.Registration =
tbl_mod_vehicle.Reg

LEFT JOIN tbl_mod ON tbl_mod_vehicle.VehicleMod = tbl_mod.ID

WHERE

VehicleValue BETWEEN 5000 AND 8000

GROUP BY PolicyID;

- Trước khi thêm index

The screenshot shows the MySQL Workbench interface with the 'EXPLAIN SELECT' results for the initial query. The 'key' column lists several indexes: PRIMARY, vln_engineCC, Mod_Vehicle_Vehicle_Reg, Policy_Vehicle_Reg, Policy_Vehicle_Reg, PRIMARY, and PRIMARY. This indicates that the query is using multiple indexes, which may be inefficient.

- Sau khi thêm index

The screenshot shows the MySQL Workbench interface with the 'EXPLAIN SELECT' results after creating the 'vln_engineCC' index. The 'key' column now shows a single index named 'vln_engineCC'. This indicates that the query is using a single index, which is more efficient than using multiple indexes.

- Partition
 - Câu lệnh truy vấn

```
EXPLAIN SELECT
    PolicyID,
    FName,
    PolicyStartDate,
    YearsNCB
FROM part_Table
    JOIN policy ON part_table.PolicyID = policy.ID
WHERE EngineClass IN (10, 50, 80, 140);
```

- Bảng để partition

CREATE TABLE part_Table

SELECT

```
policy.ID AS PolicyID,
Fname,
VehicleReg,
VehicleValue,
```

CASE

```
WHEN EngineCC BETWEEN 1000 AND 1199 THEN 0
WHEN EngineCC BETWEEN 1200 AND 1399 THEN 10
WHEN EngineCC BETWEEN 1400 AND 1599 THEN 20
WHEN EngineCC BETWEEN 1600 AND 1799 THEN 30
WHEN EngineCC BETWEEN 1800 AND 1999 THEN 40
WHEN EngineCC BETWEEN 2000 AND 2199 THEN 50
WHEN EngineCC BETWEEN 2200 AND 2399 THEN 60
WHEN EngineCC BETWEEN 2400 AND 2599 THEN 70
WHEN EngineCC BETWEEN 2600 AND 2799 THEN 80
WHEN EngineCC BETWEEN 2800 AND 2999 THEN 90
WHEN EngineCC BETWEEN 3000 AND 3199 THEN 100
WHEN EngineCC BETWEEN 3200 AND 3399 THEN 110
WHEN EngineCC BETWEEN 3400 AND 3599 THEN 120
```

```

WHEN EngineCC BETWEEN 3600 AND 3799 THEN 130
WHEN EngineCC BETWEEN 3800 AND 3999 THEN 140
ELSE 200
END AS EngineClass,
CASE
    WHEN FLOOR(DATEDIFF(PolicyStartDate, DOB) / 365) <=
16 THEN 30
        WHEN FLOOR(DATEDIFF(PolicyStartDate, DOB) / 365)
BETWEEN 17 AND 19 THEN 30
        WHEN FLOOR(DATEDIFF(PolicyStartDate, DOB) / 365)
BETWEEN 20 AND 25 THEN 20
        WHEN FLOOR(DATEDIFF(PolicyStartDate, DOB) / 365)
BETWEEN 26 AND 29 THEN 10
        WHEN FLOOR(DATEDIFF(PolicyStartDate, DOB) / 365)
BETWEEN 30 AND 35 THEN 0
        WHEN FLOOR(DATEDIFF(PolicyStartDate, DOB) / 365)
BETWEEN 35 AND 39 THEN -10
        WHEN FLOOR(DATEDIFF(PolicyStartDate, DOB) / 365)
BETWEEN 40 AND 49 THEN -20
        WHEN FLOOR(DATEDIFF(PolicyStartDate, DOB) / 365)
BETWEEN 50 AND 59 THEN -30
        WHEN FLOOR(DATEDIFF(PolicyStartDate, DOB) / 365)
BETWEEN 60 AND 90 THEN 20
    ELSE 0
END AS AgeClass,
Modifier AS OccupationClass,
CASE
    WHEN SUM(Points) > 1 THEN SUM(Points)
    ELSE 0
END AS VehicleModClass,
CASE
    WHEN MedicalCondition = 'No' THEN 0

```

```

WHEN MedicalCondition = 'Yes' THEN 10
ELSE 20

END AS MedicalConditionClass,
CASE
    WHEN PaymentType = 'Annual' THEN 0
    WHEN PaymentType = 'Monthly' THEN 20
    ELSE 10
END AS 'PaymentTypePercentageChange(%)'

FROM vehicle
JOIN policy ON vehicle.Registration = policy.VehicleReg
JOIN customer ON policy.CustomerID = customer.ID
JOIN occupation ON customer.Occupation = occupation.ID
LEFT JOIN mod_vehicle ON vehicle.Registration = mod_vehicle.Reg
LEFT JOIN `mod` ON mod_vehicle.VehicleMod = `mod`.ID
GROUP BY PolicyID;

```

- Câu lệnh partition

```

ALTER TABLE part_Table
PARTITION BY RANGE (EngineClass) (
    PARTITION p00 VALUES LESS THAN (20),
    PARTITION p01 VALUES LESS THAN (40),
    PARTITION p02 VALUES LESS THAN (60),
    PARTITION p03 VALUES LESS THAN (80),
    PARTITION p04 VALUES LESS THAN (100),
    PARTITION p05 VALUES LESS THAN (120),
    PARTITION p06 VALUES LESS THAN MAXVALUE
);

```

- Trước khi thêm partition

```

1 EXPLAIN SELECT
2   PolicyID,
3   FName,
4   PolicyStartDate,
5   YearsNCB
6   FROM part_Table
7   JOIN policy ON part_table.PolicyID = policy.ID
8   WHERE EngineClass IN (10, 50, 80, 140);

```

ID	Select_Type	Table	Partitions	Type	Possible_Keys	Key	Key_Len	Ref	Rows	Filtered	Extra
1	SIMPLE	part_Table	ALL	eq_ref	PRIMARY	PRIMARY	4	test.part_Table.PolicyID	1	100.00	Using where
1	SIMPLE	policy	ALL	eq_ref	PRIMARY	PRIMARY	4	test.part_Table.PolicyID	1	100.00	

Result 24 x

Action Output

#	Time	Action	Message	Duration / Fetch
1	2 21:46:31	EXPLAIN SELECT PolicyID, FName, PolicyStartDate, YearsNCB FROM part_Table JOIN policy ON...	2 row(s) returned	0.000 sec / 0.000 sec
2	3 21:47:49	DROP TABLE test.part_table	0 rows affected	0.125 sec
3	4 21:47:57	CREATE TABLE part_Table SELECT policy.ID AS PolicyID, FName, VehicleReg, VehicleValue, CASE WHEN...	70326 row(s) affected Records: 70326 Duplicates: 0 Warnings: 0	60.015 sec
4	5 21:49:28	DROP TABLE test.part_table	0 rows affected	0.063 sec
5	6 21:49:33	CREATE TABLE part_Table SELECT policy.ID AS PolicyID, FName, VehicleReg, VehicleValue, CASE WHEN...	200001 row(s) affected Records: 200001 Duplicates: 0 Warnings: 0	176.578 sec
6	7 21:53:00	SELECT PolicyID, FName, PolicyStartDate, YearsNCB FROM part_Table JOIN policy ON part_table...	28783 row(s) returned	0.337 sec / 0.016 sec
7	8 21:53:27	EXPLAIN SELECT PolicyID, FName, PolicyStartDate, YearsNCB FROM part_Table JOIN policy ON...	2 row(s) returned	0.000 sec / 0.000 sec

- Sau khi thêm partition

```

1 EXPLAIN SELECT
2   PolicyID,
3   FName,
4   PolicyStartDate,
5   YearsNCB
6   FROM part_Table
7   JOIN policy ON part_table.PolicyID = policy.ID
8   WHERE EngineClass IN (10, 50, 80, 140);

```

ID	Select_Type	Table	Partitions	Type	Possible_Keys	Key	Key_Len	Ref	Rows	Filtered	Extra
1	SIMPLE	part_Table	p0,p02,p04,p05	eq_ref	PRIMARY	PRIMARY	4	test.part_Table.PolicyID	1	100.00	Using where
1	SIMPLE	policy	ALL	eq_ref	PRIMARY	PRIMARY	4	test.part_Table.PolicyID	1	100.00	

Result 25 x

Action Output

#	Time	Action	Message	Duration / Fetch
1	5 21:49:28	DROP TABLE test.part_table	0 rows affected	0.063 sec
2	6 21:49:33	CREATE TABLE part_Table SELECT policy.ID AS PolicyID, FName, VehicleReg, VehicleValue, CASE WHEN...	200001 row(s) affected Records: 200001 Duplicates: 0 Warnings: 0	176.578 sec
3	7 21:53:00	SELECT PolicyID, FName, PolicyStartDate, YearsNCB FROM part_Table JOIN policy ON part_table...	28783 row(s) returned	5.937 sec / 0.016 sec
4	8 21:53:27	EXPLAIN SELECT PolicyID, FName, PolicyStartDate, YearsNCB FROM part_Table JOIN policy ON...	2 row(s) returned	0.000 sec / 0.000 sec
5	9 21:56:21	ALTER TABLE part_Table PARTITION BY RANGE (EngineClass) (PARTITION p00 VALUES LESS THAN...)	200001 row(s) affected Records: 200001 Duplicates: 0 Warnings: 0	9.000 sec
6	10 21:56:45	SELECT PolicyID, FName, PolicyStartDate, YearsNCB FROM part_Table JOIN policy ON part_table...	28783 row(s) returned	5.844 sec / 0.016 sec
7	11 21:57:01	EXPLAIN SELECT PolicyID, FName, PolicyStartDate, YearsNCB FROM part_Table JOIN policy ON...	2 row(s) returned	0.000 sec / 0.000 sec

PHẦN 3: CẬP NHẬT DỮ LIỆU

3.1 Thủ tục (Procedure) cập nhật dữ liệu

Nhóm chúng em thực hiện các Procedure: insert, update, delete, trên từng bảng dữ liệu.

Sau đây là câu lệnh thực hiện trên các bảng và kết quả:

3.1.1 Procedure Insert

- Customer

```

@ClaimsIn2Years int,
@TimeInUKSince date,
@AddressID int,
@LicenceType varchar(50),
@LicenceLocation varchar(20),
@LicenceNumber varchar(20),
@AdditionalQualification varchar(30),
@MedicalCondition varchar(30),
@UnspentMotoringConvictions int,
@PhoneNumber varchar(15),
@EmailAddress varchar(100),
@Id int
AS
BEGIN
    INSERT INTO tbl_customer (Title, FName, LName, DOB, Homeowner, Dependents, MaritalStatus, Occupation, ClaimsIn5Years, TimeInUKS
    VALUES (@Title, @FName, @LName, @DOB, @Homeowner, @Dependents, @MaritalStatus, @Occupation, @ClaimsIn2Years, @TimeInUKSince, @
END
----- RUN -----
EXEC InsertToCustomer 'Mr.', 'Nguyen', 'Trien', '2011-08-15', 2, 2, 'single', 36, 0, '1987-10-14', 71, 'Lack', 'ES', '012187234', 'None', ''
SELECT * FROM tbl_customer
) % - <

```

ID	Title	FName	LName	DOB	Homeowner	Dependants	MaritalStatus	Occupation	ClaimsIn5Years	TimeInUKSince	AddressID	LicenceType	LicenceLocation	L
16941	Prof. Kade	Haley	1976-05-16	2	1	engaged	10	2	1983-09-27	12184	Full	US	1	
16942	Prof. Alanis	Spencer	1970-12-25	1	0	single	41	1	1987-03-22	7341	Lack	IE	1	
16943	Miss Ashly	Gerlach	1980-11-06	2	1	married	24	1	1975-07-27	15995	Full	GB	1	
16944	Mr. Isaiah	Thiel	1985-08-23	0	0	engaged	13	2	1989-11-05	9774	Full	IN	1	
16945	Miss Michel	Dare	1973-02-21	1	2	single	34	2	1990-09-29	9622	Full	CA	1	
17000	Mr. Nguyen	Trien	2011-08-15	2	2	single	36	0	1987-10-14	71	Lack	ES	1	

- Claim

```

--- claim ---
CREATE PROCEDURE InsertToClaim
@Id int,
@PolicyNumber int,
@Description varchar(500),
@FaultyParty varchar(20),
@AmountPaidOut int,
@DateMade date,
@DatePaid date,
@ClaimStatus varchar(20)
AS
BEGIN
    INSERT INTO tbl_claim (ID, PolicyNumber, Description, FaultyParty, AmountPaidOut, DateMade, DatePaid, ClaimStatus)
    VALUES (@Id, @PolicyNumber, @Description, @FaultyParty, @AmountPaidOut, @DateMade, @DatePaid, @ClaimStatus)
END
----- RUN -----
EXEC InsertToClaim 19000, 8415, 'Skidded on black ice and hit a lamppost destroying front bumper and crash bar', '3
SELECT * FROM tbl_claim

```

ID	PolicyNumber	Description	FaultyParty	AmountPaidOut	DateMade	DatePaid	ClaimStatus
1...	18993	5932 Skidded on black ice and hit a lamppost destroy...	Client	1200	2021-03-31	2023-02-26	Paid
1...	18994	17874 Wing mirror smashed off during the night while v...	3rdPartyUnknown	1200	2021-05-11	2022-04-12	Pending
1...	18995	13428 Wing mirror smashed off during the night while v...	3rdPartyUnknown	400	2022-08-11	2022-03-03	Paid
1...	18996	11583 lost control and hit a roundabout. Destroying fro...	Noone	200	2023-04-04	2022-11-01	Paid
1...	18997	10420 Scrapped side of vehicle against a post in a parki...	3rdPartyUnknown	400	2022-01-17	2021-05-31	Paid
1...	18998	17024 lost control and hit a roundabout. Destroying fro...	Client	1700	2022-06-11	2022-04-22	Pending
1...	18999	8051 Lost Control over a small hill in the road. Skidded...	Noone	1500	2021-06-29	2021-10-08	Pending
1...	19000	8415 Skidded on black ice and hit a lamppost destroy...	3rdPartyUnknown	600	2021-05-05	2022-09-17	Pending

- Vehicle

```

--- Vehicle ---
CREATE PROCEDURE InsertToVehicle
@Registration varchar(8),
@ImmobiliserAlarm varchar(50),
@Tracker tinyint,
@DriverSide varchar(3),
@SeatCount int,
@VehicleValue int,
@Manufacturer int,
@Model int,
@EngineCC int,
@Transmission varchar(15)
AS
BEGIN
INSERT INTO tbl_vehicle(Registration, ImmobiliserAlarm, Tracker, DriverSide, SeatCount, VehicleValue, Manufactur
VALUES (@Registration, @ImmobiliserAlarm, @Tracker, @DriverSide, @SeatCount, @VehicleValue, @Manufacture, @Mo
END
----- RUN -----
EXEC InsertToVehicle 'th03toxer', 'FactoryFittedImmobiliserAndAlarm', 1, 'RHD', 2, 6467, 3826, 3624, 2500, 'manual'
SELECT * FROM tbl_vehicle

```

Results

Registration	ImmobiliserAlarm	Tracker	DriverSide	SeatCount	VehicleValue	Manufacturer	Model	EngineCC	Transmission
tg98thw	FactoryFittedImmobiliserAndAlarm	1	RHD	2	5266	5571	5861	1900	manual
th03oxo	FactoryFittedImmobiliserAndAlarm	0	RHD	3	4037	14427	15559	1400	manual
th03oxe	FactoryFittedImmobiliserAndAlarm	1	RHD	2	6467	3826	3624	2500	manual
th07dgr	FactoryFittedImmobiliserAndAlarm	0	RHD	3	8756	8309	15455	1000	manual
th14auv	FactoryFittedImmobiliserAndAlarm	0	RHD	3	6714	13496	10544	1700	manual
th16ozl	FactoryFittedImmobiliserAndAlarm	1	RHD	5	1878	13384	9242	1700	manual
th22vat	FactoryFittedImmobiliserAndAlarm	1	RHD	4	2120	9778	3152	2500	manual

- Policy

```

@CustomerID int,
@VehicleReg varchar(8),
@VehicleUsage varchar(30),
@EstimatedMileage int,
@PolicyStartDate date,
@PolicyEndDate date,
@CoverType varchar(30),
@PaymentType varchar(70),
@Excess int,
@YearsNCB int,
@OvernightStorage varchar(50),
@Price int
AS
BEGIN
INSERT INTO tbl_policy(ID, CustomerID, VehicleReg, VehicleUsage, EstimatedMileage, PolicyStartDate, PolicyEndDate, CoverType, Pa
VALUES (@id,@CustomerID, @VehicleReg, @VehicleUsage, @EstimatedMileage, @PolicyStartDate, @PolicyEndDate, @CoverType, @PaymentTy
END
----- RUN -----
EXEC InsertToPolicy 99999,2956,'xp20inn',' SocialOnly',6000,'2017-08-21','2022-06-08','FullyComprehensive',' Monthly',150.17,'Ga
SELECT * FROM tbl_policy

```

Results

ID	CustomerID	VehicleReg	VehicleUsage	EstimatedMileage	PolicyStartDate	PolicyEndDate	CoverType	PaymentType	Excess	YearsNCB	OvernightStorage	Price
18994	6748	il41qme	SocialOnly	9000	2017-05-27	2022-10-15	FullyComprehensive	Monthly	800	8	Garage	531
18995	2985	gj26mqa	Commuting Only	9000	2020-04-19	2025-08-10	FullyComprehensive	Monthly	400	7	Garage	190
18996	9116	jc56qsa	SocialAndComuting	5000	2019-11-18	2021-12-03	FullyComprehensive	Annual	800	8	Garage	728
18997	4852	sf48nax	SocialAndComuting	7000	2019-12-27	2021-04-28	FullyComprehensive	Annual	800	8	Driveway	160
18998	8666	xd69lod	SocialAndComuting	9000	2020-10-07	2019-10-11	FullyComprehensive	Annual	950	22	Garage	146
18999	7663	io78ppm	SocialAndComuting	8000	2019-12-28	2024-03-12	FullyComprehensive	Annual	400	13	Driveway	500
99999	2956	xo20inn	SocialOnly	6000	2017-08-21	2022-06-08	FullyComprehensive	Monthly	150	17	Garage	592

- Address

```

--- address ---
CREATE PROCEDURE InsertToAddress
@PostCode varchar(10),
@HouseNumber int,
@id int
AS
BEGIN
INSERT INTO tbl_address(PostCode, HouseNumber, ID)
VALUES (@PostCode, @HouseNumber, @id)
END

----- RUN -----
EXEC InsertToAddress test, 012345679, 16999
SELECT * FROM tbl_address

```

110 %

	ID	PostCode	HouseNumber
1...	16771	ta718uu	94
1...	16772	bt953gh	23
1...	16773	wv331nm	105
1...	16774	fl517mq	107
1...	16775	bl142oq	72
1...	16776	mh598uj	8
1...	16777	cg923dd	146
1...	16778	fd430aj	145
1...	16779	lj367yj	41
1...	16780	mx979uk	102
1...	16781	fp021bp	107
1...	16999	test	12345679

- Manufacturer

```

--- Manufacturer ---
CREATE PROCEDURE InsertToManufacturer
@id int,
@name varchar(50)
AS
BEGIN
INSERT INTO tbl_manufacturer(ID, Name)
VALUES (@id, @name)
END

----- RUN -----
EXEC InsertToManufacturer 0, Hieu
SELECT * FROM tbl_manufacturer

```

10 %

	ID	Name
1	0	Hieu
2	1	Lueilwitz Group
3	2	Dach-Blick
4	3	Bauch PLC
5	4	Mann and Sons

- Mod

```

-----+
--- Mod ---
CREATE PROCEDURE InsertToMod
@id int,
@ModName varchar(50),
@Points int
AS
BEGIN
INSERT INTO tbl_mod(ID, ModName, Points)
VALUES (@id, @ModName, @Points)
END
----- RUN -----
EXEC InsertToMod -1, DriverLights, 30
SELECT * FROM tbl_mod

```

110 %

	ID	ModName	Points
1	-1	DriverLights	30
2	1	Remapped/AftermarketUCE	50
3	2	AftermarketAlloys	50
4	3	DriverLights	10

- ModVehicle

```

--- ModVehicle ---
CREATE PROCEDURE InsertToModVehicle
@id int,
@Reg varchar(8),
@VehicleMod int
AS
BEGIN
INSERT INTO tbl_mod_vehicle(ID, Reg, VehicleMod)
VALUES (@id, @Reg, @VehicleMod)
END
----- RUN -----
EXEC InsertToModVehicle 20202, ad71brg, 10
SELECT * FROM tbl_mod_vehicle

```

110 %

	ID	Reg	VehicleMod
1...	19535	pg58srh	7
1...	19536	ad71brg	7
1...	19537	dc28pgm	10
1...	19538	xb51yzf	6
1...	19539	qw07eux	3
1...	19540	jz83tgi	7
1...	19541	ne59bdq	1
1...	20202	ad71brg	10

- Model

```

--- Model ---
CREATE PROCEDURE InsertToModel
@id int,
@modelName varchar(50),
@Manufacturer int
AS
BEGIN
INSERT INTO tbl_model(ID, ModelName, Manufacturer)
VALUES (@id, @modelName, @Manufacturer)
END
----- RUN -----
EXEC InsertToModel 9999, 'evolve 24/365 niches', 9677
SELECT * FROM tbl_model

```

110 %

	ID	modelName	Manufacturer
9...	9998	aggregate interactive action-items	8697
9...	9999	evolve 24/365 niches	9677
1...	10000	maximize end-to-end relationships	3800
1...	10001	iterate vertical vortals	12845
1...	10002	evolve clicks-and-mortar ROI	7544

- Occupation

```

--- Occupation ---
CREATE PROCEDURE InsertToOccupation
@id int,
@OccupationDescription varchar(100),
@Modifier int
AS
BEGIN
INSERT INTO tbl_occupation(ID, OccupationDescription, Modifier)
VALUES (@id, @OccupationDescription, @Modifier)
END
----- RUN -----
EXEC InsertToOccupation 0, 'Director', -40
SELECT * FROM tbl_occupation

```

10 %

	ID	OccupationDescription	Modifier
1	0	Director	-40
2	1	PartimeWorker	20
3	2	Interpreter	-32
4	3	Fundraiser	-1
5	4	Bartender	-22

3.1.1 Procedure Update

- Customer

```

@AddressID int,
@LicenceType varchar(50),
@LicenceLocation varchar(20),
@LicenceNumber varchar(20),
@AdditionalQualification varchar(30),
@MedicalCondition varchar(30),
@UnspentMotoriningConvictions int,
@TelephoneNumber varchar(15),
@EmailAddress varchar(100),
@id int
AS
BEGIN
UPDATE tbl_customer
SET ID = @id, Title = @Title, FName = @FName, LName = @LName, DOB = @DOB, Homeowner = @Homeowner, Dependants = @Dependants, Mar
WHERE ID = @id
END
----- RUN -----
EXEC UpdateToCustomer 7001,'Ms.','Bernie','Wyman','2011-08-15',2,2,'single',36,0,'1987-08-05',71,'Lack','ES','012187234','Nor
SELECT * FROM tbl_customer

```

Results

ID	Title	FName	LName	DOB	Homeowner	Dependants	MaritalStatus	Occupation	ClaimsIn5Years	TimeInUKSince	AddressID	LicenceType	LicenceLocation
6999	Dr.	Waldo	Langoh	2002-10-31	2	2	single	3	2	1982-01-07	15248	Lack	PT
7000	Prof.	Gina	Olson	1992-11-30	0	1	engaged	12	2	1970-02-03	9744	Lack	GB
7001	Ms.	Bernie	Wyman	2011-08-15	2	2	single	36	0	1987-10-14	71	Lack	ES
7002	Mrs.	Magali	Eber	1999-12-22	0	2	single	35	1	1979-04-18	1810	Full	GB
7003	Ms.	Buster	Romag...	1980-06-26	1	0	engaged	47	1	1987-04-10	16321	Full	FR
7004	Prof.	Coy	Beier	2020-05-07	2	0	married	23	2	1988-03-13	7497	Full	RU
7005	Prof.	Leora	Gutmann	1971-04-03	1	2	engaged	18	1	1982-01-08	15564	Lack	IE

- Claim

```

CREATE PROCEDURE UpdateToClaim
@PolicyNumber int,
@Description varchar(500),
@FaultyParty varchar(20),
@AmountPaidOut int,
@DateMade date,
@DatePaid date,
@ClaimStatus varchar(20),
@Id int
AS
BEGIN
UPDATE tbl_claim
SET PolicyNumber = @PolicyNumber, Description = @Description, FaultyParty = @FaultyParty, AmountPaidOut = @Am
WHERE ID = @ID
END
----- RUN -----
EXEC UpdateToClaim 3141,'Lost Control over a small hill in the road. Skidded off the road into a tree. ','3141
SELECT * FROM tbl_claim

```

Results

ID	PolicyNumber	Description	FaultyParty	AmountPaidOut	DateMade	DatePaid	ClaimStatus
8...	8886	5835	Noone	300	2023-02-03	2022-02-15	Pending
8...	8887	2196	Noone	3000	2022-11-18	2021-04-26	Paid
8...	8888	3141	3rdPartyUnknown	100	2021-04-26	2022-07-28	Pending
8...	8889	12422	Client	300	2022-03-14	2022-08-03	Pending
8...	8890	12614	3rdPartyUnknown	100	2023-01-29	2022-05-04	Paid
8...	8891	16953	Client	1200	2021-04-17	2022-07-07	Paid
o	8892	4700	2ndPartyUnknown	1500	2022-02-02	2022-02-11	Paid

- Vehicle

```

CREATE PROCEDURE UpdateToVehicle
    @Registration varchar(8),
    @ImmobiliserAlarm varchar(50),
    @Tracker tinyint,
    @DriverSide varchar(3),
    @SeatCount int,
    @VehicleValue int,
    @Manufacturer int,
    @Model int,
    @EngineCC int,
    @Transmission varchar(15)
AS
BEGIN
    UPDATE tbl_vehicle
    SET Registration = @Registration, ImmobiliserAlarm = @ImmobiliserAlarm, Tracker = @Tracker, DriverSide = @DriverSide
    WHERE Registration = @Registration
END
----- RUN -----
EXEC UpdateToVehicle 'hy19zyy', 'FactoryFittedImmobiliserAndAlarm', 1, 'RHD', 2, 2995, 6479, 10985, 1600, 'manual'
SELECT * FROM tbl_vehicle

```

Results

	Registration	ImmobiliserAlarm	Tracker	DriverSide	SeatCount	VehicleValue	Manufacturer	Model	EngineCC	Transmission
5...	hy03qdr	FactoryFittedImmobiliserAndAlarm	1	RHD	5	2454	2729	11618	1400	manual
5...	hy19yyj	FactoryFittedImmobiliserAndAlarm	0	RHD	3	7618	14750	174	1700	manual
5...	hy19zyy	FactoryFittedImmobiliserAndAlarm	1	RHD	2	2995	6479	10985	1600	manual
5...	hy23wjx	FactoryFittedImmobiliserAndAlarm	0	RHD	2	8873	3080	439	1200	manual
5...	hy24jbo	FactoryFittedImmobiliserAndAlarm	0	RHD	2	5687	2486	9623	1700	manual
5...	hy26whk	FactoryFittedImmobiliserAndAlarm	1	RHD	5	2263	6722	12860	1000	manual
5...	hy29dkv	FactoryFittedImmobiliserAndAlarm	1	RHD	2	3431	2123	9306	1000	manual

- Policy

```

@EstimatedMileage int,
@PolicyStartDate date,
@PolicyEndDate date,
@CoverType varchar(30),
@PaymentType varchar(70),
@Excess int,
@YearsNCB int,
@OvernightStorage varchar(50),
@Price int,
@id int
AS
BEGIN
    UPDATE tbl_policy
    SET CustomerID = @CustomerID, VehicleReg = @VehicleReg, VehicleUsage = @VehicleUsage, EstimatedMileage = @EstimatedMileage, Poli
    WHERE ID = @id
END
----- RUN -----
EXEC UpdateToPolicy 3117, 'lu65rvk', 'SocialAndComuting', 3000, '2019-08-28', '2021-01-08', 'FullyComprehensive', 'Annual', 950, 10, 'Garage'
SELECT * FROM tbl_policy

```

Results

	ID	CustomerID	VehicleReg	VehicleUsage	EstimatedMileage	PolicyStartDate	PolicyEndDate	CoverType	PaymentType	Excess	YearsNCB	OvernightStorage	Price
3...	3999	10181	mp45blo	Commuting Only	9000	2016-07-29	2024-05-19	FullyComprehensive	Annual	400	2	Garage	1135
4...	4000	6887	ub29nrf	SocialAndComuting	3000	2016-05-18	2024-05-07	FullyComprehensive	Monthly	300	8	Driveway	1921
4...	4001	3117	lu65rvk	SocialAndComuting	3000	2019-08-28	2021-01-08	FullyComprehensive	Annual	950	10	Garage	151
4...	4002	8245	mu66noi	SocialOnly	5000	2016-09-15	2020-04-18	FullyComprehensive	Annual	1000	1	Driveway	1435
4...	4003	5229	xj70lob	SocialOnly	6000	2020-08-10	2022-07-12	FullyComprehensive	Annual	150	11	Roadside	1515
4...	4004	9907	hp99jnh	Commuting Only	3000	2020-02-17	2020-12-28	FullyComprehensive	Monthly	300	2	Garage	1350
4...	4005	9858	ya05ai	SocialAndComuting	9000	2021-02-11	2025-09-04	FullyComprehensive	Monthly	1000	10	Roadside	1227

- Address

```

----- Address -----
CREATE PROCEDURE UpdateToAddress
@PostCode varchar(10),
@HouseNumber int,
@id int
AS
BEGIN
UPDATE tbl_address
SET PostCode = @PostCode, HouseNumber = @HouseNumber, ID = @id
WHERE ID = @id
END
----- RUN -----
EXEC UpdateToAddress test2, 0123456789, 99
SELECT * FROM tbl_address

```

110 %

	ID	PostCode	HouseNumber
97	97	cc089lb	96
98	98	eq894fy	101
99	99	test2	123456789
100	100	aa429uk	107
101	101	cn030ax	85

- Manufacturer

```

----- Manufacturer -----
CREATE PROCEDURE UpdateToManufacturer
@Name varchar(50),
@id int
AS
BEGIN
UPDATE tbl_manufacturer
SET Name = @Name, ID = @id
WHERE ID = @id
END
----- RUN -----
EXEC UpdateToManufacturer TrienNguyen, 55
SELECT * FROM tbl_manufacturer

```

110 %

	ID	Name
52	51	Kunze, Muller and Dibbert
53	52	Braun Inc
54	53	Hintz Ltd
55	54	Emmerich, Skiles and Kuhic
56	55	TrienNguyen
57	56	Grimes and Sons
58	57	Cartwright and Sons

- Mod

```

----- Mod -----
CREATE PROCEDURE UpdateToMod
@ModName varchar(50),
@Points int,
@id int
AS
BEGIN
UPDATE tbl_mod
SET ModName = @ModName, Points = @Points, ID = @id
WHERE ID = @id
END
----- RUN -----
EXEC UpdateToMod DenXe, 35, 6
SELECT * FROM tbl_mod

```

110 %

	ID	ModName	Points
4	3	DriverLights	10
5	4	SoundSystems	40
6	5	PerformanceExhaust	40
7	6	DenXe	35
8	7	TurboCharger	20
9	8	TurboCharger	50

- ModVehicle

```

----- ModVehicle -----
CREATE PROCEDURE UpdateToModVehicle
@Reg varchar(8),
@VehicleMod int,
@id int
AS
BEGIN
UPDATE tbl_mod_vehicle
SET Reg = @Reg, VehicleMod = @VehicleMod, ID = @id
WHERE ID = @id
END
----- RUN -----
EXEC UpdateToModVehicle jm13svv , 7, 9131
SELECT * FROM tbl_mod_vehicle

```

110 %

	ID	Reg	VehicleMod
9...	9129	mr03mtx	3
9...	9130	hm21fag	9
9...	9131	jm13svv	7
9...	9132	vh60bzz	9
9...	9133	jt10qpa	5
9...	9134	irR4tvl	9

- Model

```

----- Model -----
CREATE PROCEDURE UpdateToModel
@modelName varchar(50),
@Manufacturer int,
@id int
AS
BEGIN
UPDATE tbl_model
SET ModelName = @modelName, Manufacturer = @Manufacturer, ID = @id
WHERE ID = @id
END
----- RUN -----
EXEC UpdateToModel 'revolutionize web-enabled technologies', 5890, 222
SELECT * FROM tbl_model

```

110 %

	ID	ModelName	Manufacturer
220	220	exploit world-class metrics	4148
221	221	innovate value-added convergence	12090
222	222	revolutionize web-enabled technol...	5890
223	223	streamline frictionless web-readiness	11/29
224	224	cultivate mission-critical solutions	721
225	225	unleash B2C e-services	5113
226	226	expedite e-business convergence	5443
227	227	minimize transactional overheads	11016

- Occupation

```

----- Occupation -----
CREATE PROCEDURE UpdateToOccupation
@OccupationDescription varchar(100),
@Modifier int,
@id int
AS
BEGIN
UPDATE tbl_occupation
SET OccupationDescription = @OccupationDescription, Modifier = @Modifier, ID = @id
WHERE ID = @id
END
----- RUN -----
EXEC UpdateToOccupation 'Actor', 27, 24
SELECT * FROM tbl_occupation

```

110 %

	ID	OccupationDescription	Modifier
22	21	Tailor	8
23	22	Lawyer	0
24	23	Actuary	-20
25	24	Actor	27
26	25	Unemployed	-25
27	26	ConstructionManager	-28
28	27	Barber	-26

3.1.3 Procedure Delete

- Customer

```

----- Customer -----
CREATE PROCEDURE DeleteToCustomer
@id int
AS
BEGIN
DELETE FROM tbl_customer
WHERE id = @id
END
----- RUN -----
EXEC DeleteToCustomer 22
SELECT * FROM tbl_customer

```

110 %

	ID	Title	FName	LName	DOB	Homeowner	Dependants
19	19	Dr.	Candida	Mayer	2004-04-24	2	2
20	20	Prof.	Sophie	Wyman	1981-08-28	1	2
21	21	Dr.	Hattie	Tremblay	2009-11-06	2	0
22	23	Mr.	Marcell...	Ebert	1986-11-25	2	0
23	24	Dr.	Annabell	Poulos	2013-07-15	0	1

- Claim

```

----- Claim -----
CREATE PROCEDURE DeleteToClaim
@id int
AS
BEGIN
DELETE FROM tbl_claim
WHERE id = @id
END
----- RUN -----
EXEC DeleteToClaim 09
SELECT * FROM tbl_claim

```

110 %

	ID	PolicyNumber	Description	FaultyParty
7	7	15319	Scraped side of vehicle against a post in a parki...	Client
8	8	15495	lost control and hit a roundabout. Destroying fro...	3rdPartyUnknown
9	10	14816	Lost Control over a small hill in the road. Skidded...	3rdPartyUnknown

- Vehicle

```

----- Vehicle -----
CREATE PROCEDURE DeleteToVehicle
@Registration varchar(8)
AS
BEGIN
DELETE FROM tbl_vehicle
WHERE Registration = @Registration
END
----- RUN -----
EXEC DeleteToVehicle 'th03toxer'
SELECT * FROM tbl_vehicle

```

110 %

	Registration	ImmobiliserAlarm	Tracker	Drive
1...	tg88ucv	FactoryFittedImmobiliserAndAlarm	1	RHC
1...	tg90gtm	FactoryFittedImmobiliserAndAlarm	0	RHC
1...	tg98thw	FactoryFittedImmobiliserAndAlarm	1	RHC
1...	th03tox	FactoryFittedImmobiliserAndAlarm	0	RHC
1...	th07dgr	FactoryFittedImmobiliserAndAlarm	0	RHC
1...	th14auv	FactoryFittedImmobiliserAndAlarm	0	RHC
1...	th16ozl	FactoryFittedImmobiliserAndAlarm	1	RHC

- Policy

```

----- Policy -----
CREATE PROCEDURE DeleteToPolicy
@id int
AS
BEGIN
DELETE FROM tbl_policy
WHERE id = @id
END
----- RUN -----
EXEC DeleteToPolicy 44
SELECT * FROM tbl_policy

```

110 %

	ID	CustomerID	VehicleReg	VehicleUsage
43	43	7268	dw31dqx	SocialOnly
44	45	12745	wb83jni	Commuting Only
45	46	698	qh92glo	SocialOnly
46	47	1956	vx82ums	SocialAndComuting

- Address

```

----- Address -----
CREATE PROCEDURE DeleteToAddress
@id int
AS
BEGIN
DELETE FROM tbl_address
WHERE id = @id
END
----- RUN -----
EXEC DeleteToAddress 02
SELECT * FROM tbl_address

```

110 %

	ID	PostCode	HouseNumber
1	1	hj966ni	38
2	3	ta770wf	24
3	4	ok696ku	178
4	5	al131lb	7

- Manufacturer

```

----- Manufacturer -----
CREATE PROCEDURE DeleteToManufacturer
@id int
AS
BEGIN
DELETE FROM tbl_manufacturer
WHERE id = @id
END
----- RUN -----
EXEC DeleteToManufacturer 0
SELECT * FROM tbl_manufacturer

```

1 %

ID	Name
1	Lueilwitz Group
2	Dach-Blick
3	Bauch PLC

- Mod

```

----- Mod -----
CREATE PROCEDURE DeleteToMod
@id int
AS
BEGIN
DELETE FROM tbl_mod
WHERE id = @id
END
----- RUN -----
EXEC DeleteToMod 0
SELECT * FROM tbl_mod

```

110 %

	ID	ModName	Points
1	-1	DriverLights	30
2	1	Remapped/AftermarketUCE	50
3	2	AftermarketAlloys	50

- ModVehicle

```

----- ModVehicle -----
CREATE PROCEDURE DeleteToModVehicle
@id int
AS
BEGIN
DELETE FROM tbl_mod_vehicle
WHERE id = @id
END
----- RUN -----
EXEC DeleteToModVehicle 15432
SELECT * FROM tbl_mod_vehicle

```

110 %

	ID	Reg	VehicleMod
1...	15428	xj69ajg	9
1...	15429	gg83enj	8
1...	15430	yk79gpw	6
1...	15431	zp33hmb	2
1...	15433	ui62oay	2

- Model

```

----- Model -----
CREATE PROCEDURE DeleteToModel
@id int
AS
BEGIN
DELETE FROM tbl_model
WHERE id = @id
END
----- RUN -----
EXEC DeleteToModel 9999
SELECT * FROM tbl_model

```

110 %

	ID	ModelName	Manufacturer
9...	9996	seize 24/7 technologies	2426
9...	9997	repurpose compelling functionalities	6867
9...	9998	aggregate interactive action-items	8697
9...	10000	maximize end-to-end relationships	3800
1	10001	Reactor vertical website	120AF

- Occupation

```

----- Occupation -----
CREATE PROCEDURE DeleteToOccupation
@id int
AS
BEGIN
DELETE FROM tbl_occupation
WHERE id = @id
END
----- RUN -----
EXEC DeleteToOccupation 22
SELECT * FROM tbl_occupation

```

110 %

	ID	OccupationDescription	Modifier
19	19	WebDevelopper	-18
20	20	BicycleRepairer	36
21	21	Tailor	8
22	23	Actuary	-20
23	24	Actor	27

3.2 Thực hiện giao dịch (transaction)

3.2.1 Khái niệm transaction

- Là tiến trình thực hiện một nhóm các câu lệnh SQL. Các câu lệnh này được thực thi một cách tuần tự và độc lập. Một transaction được thực hiện thành công khi tất cả câu lệnh đều thành công, khi đó tất cả các thay đổi dữ liệu được thực hiện trong transaction được lưu vào cơ sở dữ liệu.

- Các thuộc tính của transaction:



3.2.2 Transaction nhóm thực hiện

Nhóm em thực hiện 3 transaction thông qua tình huống thực tế, xoay quanh việc mua bảo hiểm xe của một cá nhân.

- *UpdateCustomer* : Anh H trước đây từng mua bảo hiểm xe máy, anh H nay cưới vợ, có nhà nên ta cập nhật thông tin anh H.

```
-- Update Customer --

BEGIN TRANSACTION;

UPDATE tbl_address
SET PostCode = 'er235kc', HouseNumber = 36
WHERE ID = 1111

UPDATE tbl_occupation
SET OccupationDescription = 'Teacher', Modifier = 14
WHERE ID = 22

UPDATE tbl_customer
SET Homeowner = 1, Dependents = 0, MaritalStatus = 'married'
WHERE Occupation = 22 AND AddressID = 1111

COMMIT;
```

- *NewPolicy* : Sau một thời gian, anh H tiết kiệm tiền mua được con xe 4 bánh, nên anh đi đăng ký bảo hiểm xe.

```
BEGIN TRANSACTION;

DECLARE @AddID int
SELECT @AddID = MAX(ID) + 1 FROM tbl_address;

DECLARE @OccID int
SELECT @OccID = MAX(ID) + 1 FROM tbl_occupation;

DECLARE @CusID int
SELECT @CusID = MAX(ID) + 1 FROM tbl_customer;

DECLARE @PolID int
SELECT @PolID = MAX(ID) + 1 FROM tbl_policy;

INSERT INTO tbl_address(ID, PostCode, HouseNumber) VALUES (@AddID, 10000, 5);

INSERT INTO tbl_occupation(ID, OccupationDescription, Modifier) VALUES (@OccID, 'student', 20);

INSERT INTO tbl_customer(ID, Title, FName, LName, DOB, Homeowner, Dependants, MaritalStatus, Occupation, ClaimsIn5Years, TimeInUKSince, AddressID, LicenceType, VALUES (@CusID, 'Mr.', 'Hieu', 'Nguyen', '2001-02-12', 0, 0, 'single', @OccID, 0, GETDATE(), @AddID, 'Full', 'VN', 001201006959, 'None', 'No', 0, 0382128612, 't

INSERT INTO tbl_policy(ID, CustomerID, VehicleReg, VehicleUsage, EstimatedMileage, PolicyStartDate, PolicyEndDate, CoverType, PaymentType, Excess, YearsNCB, Ove
VALUES (@PolID, @CusID, 'aa31qcn', 'SocialOnly', 6000, GETDATE(), DATEADD(MONTH, 30 ,GETDATE()) , 'FullyComprehensive', 'Monthly', 1000, 5, 'DriveWay', 3457);

COMMIT;
```

Messages

1 row affected

1 row affected

1 row affected

- *DeletePolicy* : Khi đã có ô tô thì anh H không còn đi xe máy nữa và bán đi, vì thẻ bảo hiểm bị hủy.

```
---- Delete Policy --
BEGIN TRANSACTION;
DELETE FROM tbl_address
WHERE ID = 1111

DELETE FROM tbl_occupation
WHERE ID = 22
SAVE TRANSACTION SP1

DELETE FROM tbl_customer
WHERE Occupation = 22 AND AddressID = 1111

SAVE TRANSACTION SP2
DECLARE @id int
SELECT @id = ID FROM tbl_customer
WHERE Occupation = 22 AND AddressID = 1111

DELETE FROM tbl_policy
WHERE CustomerID = @id

ROLLBACK TRANSACTION SP1;
ROLLBACK TRANSACTION SP2;
COMMIT;
```

3.3 Sinh CSDL

Thực hiện tạo một database rỗng sau đó đổ dữ liệu từ database cũ qua database mới.

Các bước thực hiện :

- Tạo database rỗng có tên *DBempty*:



- Tạo procedure để dữ liệu vào theo từng bảng:
- address

```
-- Address --
CREATE PROCEDURE CopyTable_Address
AS
    DECLARE @AddressID int, @PostCode varchar(10), @HouseNumber int
    DECLARE Copy_Cursor CURSOR FOR
        SELECT ID, PostCode, HouseNumber FROM DBInsurance.dbo.tbl_address
    OPEN Copy_Cursor
    FETCH NEXT FROM Copy_Cursor INTO @AddressID, @PostCode, @HouseNumber
    WHILE @@FETCH_STATUS = 0
    BEGIN
        INSERT INTO DBempty.dbo.tbl_address (ID, PostCode, HouseNumber)
        VALUES(@AddressID, @PostCode, @HouseNumber)
        FETCH NEXT FROM Copy_Cursor INTO @AddressID, @PostCode, @HouseNumber
    END
    CLOSE Copy_Cursor DEALLOCATE Copy_Cursor
```

- occupation

```
-- Occupation-----
CREATE PROCEDURE CopyTable_Occupation
AS
    DECLARE @id int, @OccupationDescription varchar(100), @Modifier int
    DECLARE Copy_Cursor CURSOR FOR
        SELECT ID, OccupationDescription, Modifier FROM DBInsurance.dbo.tbl_occupation
    OPEN Copy_Cursor
    FETCH NEXT FROM Copy_Cursor INTO @id, @OccupationDescription, @Modifier
    WHILE @@FETCH_STATUS = 0
        BEGIN
            INSERT INTO DBempty.dbo.tbl_occupation (ID, OccupationDescription, Modifier)
            VALUES(@id, @OccupationDescription, @Modifier)
            FETCH NEXT FROM Copy_Cursor INTO @id, @OccupationDescription, @Modifier
        END
    CLOSE Copy_Cursor DEALLOCATE Copy_Cursor
```

- customer

```
----- Customer -----
CREATE PROCEDURE CopyTable_Customer
AS
    DECLARE @Title varchar(10),@FName varchar(50),@LName varchar(50),@DOB date,@Homeowner tinyint,@Dependants int,@MaritalStatus var
    DECLARE Copy_Cursor CURSOR FOR
    SELECT ID, Title, FName, LName, DOB, Homeowner, Dependants, MaritalStatus, Occupation, ClaimsIn5Years, TimeInUkSince, AddressID,
    FROM DBInsurance.dbo.tbl_customer
    OPEN Copy_Cursor
    FETCH NEXT FROM Copy_Cursor INTO @id, @Title, @FName, @LName, @DOB, @Homeowner, @Dependants, @MaritalStatus, @Occupation, @C
    WHILE @@FETCH_STATUS = 0
        BEGIN
            INSERT INTO DBempty.dbo.tbl_customer (ID, Title, FName, LName, DOB, Homeowner, Dependants, MaritalStatus, Occupation, Cl
            VALUES(@id, @Title, @FName, @LName, @DOB, @Homeowner, @Dependants, @MaritalStatus, @Occupation, @Claimsln2Years, @Timeln
            FETCH NEXT FROM Copy_Cursor INTO @id, @Title, @FName, @LName, @DOB, @Homeowner, @Dependants, @MaritalStatus, @Occupation
        END
    CLOSE Copy_Cursor DEALLOCATE Copy_Cursor
```

- manufacturer

```
----- Manufacturer -----
CREATE PROCEDURE CopyTable_Manufacturer
AS
    DECLARE @id int, @name varchar(50)
    DECLARE Copy_Cursor CURSOR FOR
    SELECT ID, Name   FROM DBInsurance.dbo.tbl_manufacturer
    OPEN Copy_Cursor
    FETCH NEXT FROM Copy_Cursor INTO @id, @name
    WHILE @@FETCH_STATUS = 0
        BEGIN
            INSERT INTO DBempty.dbo.tbl_manufacturer (ID, Name)
            VALUES(@id, @name)
            FETCH NEXT FROM Copy_Cursor INTO @id, @name
        END
    CLOSE Copy_Cursor DEALLOCATE Copy_Cursor
```

- model

```
-- Model --
CREATE PROCEDURE CopyTable_Model
AS

DECLARE @id int, @ModelName varchar(50), @Manufacturer int

DECLARE Copy_Cursor CURSOR FOR
SELECT ID, ModelName, Manufacturer FROM DBInsurance.dbo.tbl_model

OPEN Copy_Cursor

FETCH NEXT FROM Copy_Cursor INTO @id, @ModelName, @Manufacturer
WHILE @@FETCH_STATUS = 0

BEGIN
    INSERT INTO DBempty.dbo.tbl_model (ID, ModelName, Manufacturer)
    VALUES(@id, @ModelName, @Manufacturer)
    FETCH NEXT FROM Copy_Cursor INTO @id, @ModelName, @Manufacturer
END

CLOSE Copy_Cursor DEALLOCATE Copy_Cursor
```

- vehicle

```
-- Vehicle --
CREATE PROCEDURE CopyTable_Vehicle
AS

DECLARE @Registration varchar(8), @ImmobiliserAlarm varchar(50), @Tracker tinyint, @DriverSide varchar(3), @SeatCount int, @VehicleValue decimal(10,2), @Manufacturer nvarchar(50), @Model nvarchar(50), @EngineCC int, @Transmission nvarchar(50)
DECLARE Copy_Cursor CURSOR FOR

SELECT Registration, ImmobiliserAlarm, Tracker, DriverSide, SeatCount, VehicleValue, Manufacturer, Model, EngineCC, Transmission
FROM DBInsurance.dbo.tbl_vehicle

OPEN Copy_Cursor
FETCH NEXT FROM Copy_Cursor INTO @Registration, @ImmobiliserAlarm, @Tracker, @DriverSide, @SeatCount, @VehicleValue, @Manufacturer, @Model, @EngineCC, @Transmission
WHILE @@FETCH_STATUS = 0
BEGIN
    INSERT INTO DBempty.dbo.tbl_vehicle (Registration, ImmobiliserAlarm, Tracker, DriverSide, SeatCount, VehicleValue, Manufacturer, Model, EngineCC, Transmission)
    VALUES(@Registration, @ImmobiliserAlarm, @Tracker, @DriverSide, @SeatCount, @VehicleValue, @Manufacturer, @Model, @EngineCC, @Transmission)
    FETCH NEXT FROM Copy_Cursor INTO @Registration, @ImmobiliserAlarm, @Tracker, @DriverSide, @SeatCount, @VehicleValue, @Manufacturer, @Model, @EngineCC, @Transmission
END

CLOSE Copy_Cursor DEALLOCATE Copy_Cursor
```

- mod

```
-- Mod --
CREATE PROCEDURE CopyTable_Mod
AS

DECLARE @id int, @ModName varchar(50), @Points int

DECLARE Copy_Cursor CURSOR FOR
SELECT ID, ModName, Points FROM DBInsurance.dbo.tbl_mod

OPEN Copy_Cursor

FETCH NEXT FROM Copy_Cursor INTO @id, @ModName, @Points
WHILE @@FETCH_STATUS = 0

BEGIN
    INSERT INTO DBempty.dbo.tbl_mod (ID, ModName, Points)
    VALUES(@id, @ModName, @Points)
    FETCH NEXT FROM Copy_Cursor INTO @id, @ModName, @Points
END

CLOSE Copy_Cursor DEALLOCATE Copy_Cursor
```

- modvehicle

```
-- ModVehicle --
CREATE PROCEDURE CopyTable_ModVehicle
AS

DECLARE @id int, @Reg varchar(8), @VehicleMod int

DECLARE Copy_Cursor CURSOR FOR
SELECT ID, Reg, VehicleMod FROM DBInsurance.dbo.tbl_mod_vehicle
OPEN Copy_Cursor

FETCH NEXT FROM Copy_Cursor INTO @id, @Reg, @VehicleMod
WHILE @@FETCH_STATUS = 0

BEGIN
    INSERT INTO DBempty.dbo.tbl_mod_vehicle (ID, Reg, VehicleMod)
    VALUES(@id, @Reg, @VehicleMod)
    FETCH NEXT FROM Copy_Cursor INTO @id, @Reg, @VehicleMod
END

CLOSE Copy_Cursor DEALLOCATE Copy_Cursor
```

- policy

```
-- Policy -----
CREATE PROCEDURE CopyTable_Policy
AS

DECLARE @id int, @CustomerID int, @VehicleReg varchar(8), @VehicleUsage varchar(30), @EstimatedMileage int, @PolicyStartDate date
DECLARE Copy_Cursor CURSOR FOR
SELECT ID, CustomerID, VehicleReg, VehicleUsage, EstimatedMileage, PolicyStartDate, PolicyEndDate, CoverType, PaymentType, Excess
FROM DBInsurance.dbo.tbl_policy

OPEN Copy_Cursor

FETCH NEXT FROM Copy_Cursor INTO @id,@CustomerID, @VehicleReg, @VehicleUsage, @EstimatedMileage, @PolicyStartDate, @PolicyEndD...
WHILE @@FETCH_STATUS = 0

BEGIN
    INSERT INTO DBempty.dbo.tbl_policy (ID, CustomerID, VehicleReg, VehicleUsage, EstimatedMileage, PolicyStartDate, PolicyEndD...
    VALUES (@id,@CustomerID, @VehicleReg, @VehicleUsage, @EstimatedMileage, @PolicyStartDate, @PolicyEndDate, @CoverType, @P...
    FETCH NEXT FROM Copy_Cursor INTO @id,@CustomerID, @VehicleReg, @VehicleUsage, @EstimatedMileage, @PolicyStartDate, @Policy...
END

CLOSE Copy_Cursor DEALLOCATE Copy_Cursor
```

- claim

```
-- Claim -----
CREATE PROCEDURE CopyTable_Claim
AS

DECLARE @id int, @PolicyNumber int, @Description varchar(500), @FaultyParty varchar(20), @AmountPaidOut int, @DateMade date, @Da...
DECLARE Copy_Cursor CURSOR FOR
SELECT ID, PolicyNumber, Description, FaultyParty, AmountPaidOut, DateMade, DatePaid, ClaimStatus FROM DBInsurance.dbo.tbl_claim

OPEN Copy_Cursor

FETCH NEXT FROM Copy_Cursor INTO @id, @PolicyNumber, @Description, @FaultyParty, @AmountPaidOut, @DateMade, @DatePaid, @Claim...
WHILE @@FETCH_STATUS = 0

BEGIN
    INSERT INTO DBempty.dbo.tbl_claim (ID, PolicyNumber, Description, FaultyParty, AmountPaidOut, DateMade, DatePaid, Claims...
    VALUES(@id, @PolicyNumber, @Description, @FaultyParty, @AmountPaidOut, @DateMade, @DatePaid, @ClaimStatus)
    FETCH NEXT FROM Copy_Cursor INTO @id, @PolicyNumber, @Description, @FaultyParty, @AmountPaidOut, @DateMade, @DatePaid, @C...
END

CLOSE Copy_Cursor DEALLOCATE Copy_Cursor
```

- Chạy các procedure – complete

RUN

```
EXEC CopyTable_Address
EXEC CopyTable_occupation
EXEC CopyTable_Customer
EXEC CopyTable_Manufacturer
EXEC CopyTable_Model
EXEC CopyTable_Vehicle
EXEC CopyTable_Mod
EXEC CopyTable_ModVehicle
EXEC CopyTable_Policy
EXEC CopyTable_Claim
```

Messages

```
[1 row affected]
```

KẾT LUẬN

Những vấn đề nhóm đã làm được và học được:

- Vận dụng được các kiến thức đã học để tiến hành nghiên cứu database trên phần mềm SQL.
- Nắm được các khái niệm cơ bản cũng như thực hiện các thao tác trên database như chuẩn hóa dữ liệu, thiết kế dữ liệu, mô tả dữ liệu thông qua các diagram.
- Thực hiện tối ưu các câu lệnh truy vấn, thực hiện các câu lệnh truy vấn có tính nghiệp vụ thực tế, thực hiện nghiệp vụ thực tế trên các transaction.
- Nhóm cũng đã hiểu rõ hơn về cách làm việc trên một cơ sở dữ liệu cụ thể, biết cách áp dụng vào thực tế thông qua các nghiệp vụ, các câu lệnh truy vấn trên một cơ sở dữ liệu nhất định.
- Học được kỹ năng làm việc nhóm, phân chia công việc và thu thập dữ liệu cho database.

Một số vấn đề nhóm còn hạn chế:

- Các procedure vẫn làm tay, chưa có code sinh tự động.
- Các ý tưởng transaction còn hạn chế.

Trên đây là tổng kết hoạt động nhóm của chúng em, trong quá trình tổng hợp khó có thể tránh được những sai sót, mong thầy bỏ qua. Nhóm chúng em chân thành cảm ơn thầy Nguyễn Danh Tú vì những kiến thức bổ ích, cũng như những hướng dẫn tận tình của thầy đã giúp chúng em tiếp thu kiến thức một cách hiệu quả nhất, có thể đem áp dụng vào những vấn đề thực tiễn cũng như cho công việc sau này.

Tài liệu tham khảo

- Slide thầy Nguyễn Danh Tú cung cấp.
- Web: <https://www.mysqltutorial.org/>
- Các nguồn internet khác.