|  |  |
| --- | --- |
| **Project Case** |  |
| ISYS6169 | ISYS6169001  Database Systems |
| **Information Systems** | **O222-ISYS6169-WM06-00** |
| ***Valid on*** *Odd Semester Year 2021/2022* | **Revision 00** |

1. Seluruh kelompok tidak diperkenankan untuk:

*The whole group is not allowed to:*

* + - Melihat sebagian atau seluruh proyek kelompok lain,

*Seeing a part or the whole project from another groups*

* + - Menyadur sebagian maupun seluruh proyek dari buku,

*Adapted a part or the whole project from the book*

* + - Mendownload sebagian maupun seluruh proyek dari internet,

*Downloading a part or the whole project from the internet,*

* + - Mengerjakan soal yang tidak sesuai dengan tema yang ada di soal proyek,

*Working with another theme which is not in accordance with the existing theme in the matter of the project,*

* + - Melakukan tindakan kecurangan lainnya,

*Committing other dishonest actions,*

* + - Secara sengaja maupun tidak sengaja melakukan segala tindakan kelalaian yang menyebabkan hasil karyanya berhasil dicontek oleh orang lain / kelompok lain.

*Accidentally or intentionally conduct any failure action that cause the results of the project was copied by someone else / other groups.*

1. Jika kelompok terbukti melakukan tindakan seperti yang dijelaskan butir 1 di atas, maka **nilai kelompok** yang melakukan kecurangan (menyontek maupun dicontek) akan di – **NOL** – kan.

*If the group is proved to the actions described in point 1 above, the score of the group which committed dishonest acts (cheating or being cheated) will be “Zero”*

1. Perhatikan jadwal pengumpulan proyek, segala jenis pengumpulan proyek di luar jadwal tidak dilayani.

*Pay attention to the submission schedule for the project, all kinds of submission outside the project schedule will not be accepted*

1. Bila Anda tidak membaca peraturan ini, maka Anda dianggap telah membaca dan menyetujuinya

*If you have missed to read these regulations, so you are considered to have read and agreed on it*

1. Persentase penilaiaan untuk matakuliah ini adalah sebagai berikut:

*Marking percentage for this subject is described as follows:*

|  |  |  |
| --- | --- | --- |
| **Tugas Mandiri**  *Assignment* | **Proyek**  *Project* | **UAP**  *Final Exam* |
| 30% | 30% | 40% |

1. Software yang digunakan pada matakuliah ini adalah sebagai berikut:

*Software will be used in this subject are described as follows:*

|  |
| --- |
| **Software**  *Software* |
| SQL Server Management Studio 18.5.1  SQL Server Developer 2019  Microsoft Office 365  Visual Paradigm Community Edition 16.1 |

## Ekstensi file yang harus disertakan dalam pengumpulan tugas mandiri, proyek dan uap untuk matakuliah ini adalah sebagai berikut:

*File extensions should be included in assignment, project, and final exam collection for this subject are described as follows:*

|  |  |  |
| --- | --- | --- |
| **Tugas Mandiri**  *Assignment* | **Proyek**  *Project* | **UAP**  *Final Exam* |
| SQL | SQL, VPP, Image Files (JPG / PNG) | SQL |

## Soal

*Case*

**Bakmi WM**

**Bakmi WM** is one of the popular restaurants in Indonesia which sells many kinds of food, especially noodles. The owner, Martin, manages transactions occurred in **Bakmi WM** like **serving menu and selling souvenirs to customers**.

Every staff in **Bakmi WM** have a task to **serve a customer who wants to order some menu** and **sell the souvenirs to the customers**. Every staff must be following the procedures to become a staff, which are:

* Every **staff** must have a personal information like name, gender, date of birth, phone number, email, address, and salary. Every staff has an identification number with the following format:

“SFXXX”

X => number between 0 – 9

* Every **staff** has its own **position** data and the position data has an identification id with the following format:

“SPXXX”

X => number between 0 – 9

* Staff can sell souvenirs to the customers.
* Every **souvenir transaction** occurred have all the information about staff, customer, transaction date, souvenirs purchased, and the quantity of each souvenir. Every **souvenir transaction** has an identification number with the following format:

“STXXX”

X => number between 0 – 9

* Every **souvenir** purchased by the customer have its own name, buy price, and sell price. Every souvenir has an identification number with the following format:

“SOXXX”

X => number between 0 – 9

* Staff can also serve a customer who wants to order some menu.
* Every **menu transaction** made by the customer have all the information about staff, customer, transaction date, menu ordered, and the quantity of each menu. Every **menu transaction** has an identification number with the following format:

“MTXXX”

X => number between 0 – 9

* Every **menu** sold by **Bakmi WM** have its own name and price. Every menuhas an identification number with the following format:

“MEXXX”

X => number between 0 – 9

* Every **menu** has its own **category** data, and the category data has an identification id with the following format:

“MCXXX”

X => number between 0 – 9

Every customer that wants to order menu or buy souvenirs at **Bakmi WM** must be following the **sales transaction procedures**, those are:

* Every **customer** that wants to purchase a product must already completed personal information like name, gender, phone number, email, and address. Every customer has an identification number with the following format:

“CUXXX”

X => number between 0 – 9

* Customer can purchase **more than one product** in every transaction.

**Notes:**

* Menu price must be between 1000 and 10000000
* Staff address must be more than 15 characters.
* Staff gender must be either “Male” or “Female” (without quote).
* Staff year of birth must be less than 2005.
* Staff phone number must be numeric.
* Customer gender must be either “Male” or “Female” (without quote).
* Customer phone number must be numeric.

Now **Bakmi WM** is still using manual management system to maintain the **menu** and **souvenir transactions**. You as hired programmers are asked to help **Bakmi WM** to create a database system that can store data and maintain the **menu** and **souvenir transactions**. The tasks that you must do are:

1. Create Entity Relationship Diagram to maintain **menu** and **souvenir transactions**.
2. Create a database system using DDL syntax that relevant with **menu** and **souvenir transactions**.
3. Create query using DML syntax to fill the tables in database systems with data based on the following conditions:

* **Master** table must be filled with more than or equals 10 data.
* **Transaction** table must be filled with more than or equals 15 data.
* **Transaction detail** table must be filled with more than or equals 25 data.
* For the **Staff Position** table, the table must be filled with the following data:

|  |  |
| --- | --- |
| Staff Position Names | |
| Chairman | Accountant |
| Director | Cashier |
| Assistant Manager | Chef |
| Secretary | Waiter |
| Treasurer | Dishwasher |

* For the **Menu Category** table, the table must be filled with the following data:

|  |  |
| --- | --- |
| Menu Categories | |
| Food | Beverage |
| Dessert | Topping |

1. Create query using DML syntax to simulate the transactions process for **menu** and **souvenir transactions**.

**Note**: DML syntax to **fill database** and DML syntax to **simulate** the **transactions process** should be a **different query**.

1. To support database management process in **Bakmi WM**,Martin asked you to provide some query to obtain some important data. The requirements asked are as follows:
2. Display CustomerID, CustomerName, and Total Transaction (obtained from the total transaction and ended with ' purchase(s)') for each customer whose name contains 'e' and served by a staff whose name ends with 'n'.
3. Display SouvenirTransactionID, StaffID, CustomerID, and Total Price (obtained by adding 'Rp. ' in front of the sum of multiplication of the souvenir sell price and quantity) for each purchase which customer's name length is more than 10 and the souvenir sell price is greater than 35000.
4. Display MenuTransactionID, CustomerName, MenuTransactionDate (obtained from MenuTransactionDate with 'dd-mm-yyyy' format), Total Product (obtained from the number of products) and Total Quantity (obtained from the sum of all product's quantities) for every menu transaction which occurred at even day and the customer's name consists of at least 2 words
5. Display SouvenirTransactionID, Staff First Name (obtained from the first name of the staff), Total Product (obtained from the count of product), and Total Quantity (obtained from the sum of quantity) for every souvenir transaction which staff's name consists of more than one word and staff's salary is more than 10000000
6. Display unique SouvenirTransactionID, StaffName, SouvenirTransactionDate (obtained from SouvenirTransactionDate with 'dd-mm-yyyy' format), Salary (obtained by adding ‘Rp. ’ in front of the StaffSalary) for every menu transaction which has a souvenir which buy price is more than 10000 and handled by a staff whose salary is more than average.

(**alias subquery**)

1. Display StaffName, MenuName, MenuTransactionDate (obtained from SouvenirTransactionDate with 'dd-mm-yyyy' format), Staff Local Phone (obtained by replacing StaffPhone first character into ‘+62’) for every menu transaction which is served by female staff and menu price is higher than the average sell price of all souvenirs.

(**alias subquery**)

1. Display SouvenirTransactionID, Capitalized Customer Name (obtained from the uppercase of the customer's name), and Total Quantity (obtained from the sum of quantity purchased and ended with ' pc(s)') for every souvenir transaction which id number is odd and has total quantity purchased higher than the maximum quantity of all souvenir transaction.

(**alias subquery**)

1. Display Staff Number (obtained from replacing the 'SF' in StaffID with 'Staff '), StaffName, StaffPositionName, and Total Quantity (obtained from the sum of quantity purchased) for every menu transaction which total quantity is less than or equals to the minimum purchase quantity in every purchase that occurred between the 16th and 25th day of the month.

(**alias subquery**)

1. Create a view named 'CustomerMenuPurchaseViewer' to display CustomerID, CustomerName, CustomerEmail, Maximum Quantity (obtained from the maximum quantity purchased), and Minimum Quantity (obtained from the minimum quantity purchased) for every customer whose id is even and the maximum quantity doesn't equal to its minimum quantity.
2. Create a view named 'StaffSouvenirSellingViewer' to display StaffID, StaffName, StaffAddress, and Total Price (obtained by the sum of quantity purchased times souvenir sell price) for every staff whose address consists of at least 3 words and the minimum quantity purchased is more than 5.

**File that must be collected**:

1. Entity Relationship Diagram (.vpp, .png)
2. Query to create the database system. (.sql)
3. Query to insert data into tables. (.sql)
4. Query to simulate the transactions processes. (.sql)
5. Query to answer the 10 cases. (.sql)

Here are the rules that you must follow to create your project:

1. Use appropriate software for this subject based on **Sistem Praktikum** that can be downloaded from Binusmaya
2. Use the techniques taught during practicum
3. Collect appropriate files for this subject based on **Sistem Praktikum** that can be downloaded from Binusmaya
4. Include the other files that can support your project, such as:
   * All files in your project
   * Other files (image, audio, video, etc.) used in your project
   * \*.DOC file (documentation of your project) that contains the reference links of additional files (image, audio, video, etc.) used in your project