|  |  |
| --- | --- |
| **Project Case** |  |
| ISYS6084 | ISYS6123 | ISYS6123003 | ISYS6169 | ISYS6169001  Database | Introduction to Database Systems | Database Systems |
| **Information Systems** | **E222-ISYS6123003-FL05-00** |
| ***Valid on*** *Even Semester Year 2021/2022* | **Revision 00** |

1. Seluruh kelompok tidak diperkenankan untuk:

*The whole group is not allowed to:*

* + 1. Melihat sebagian atau seluruh proyek kelompok lain,

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

* + 1. Menyadur sebagian maupun seluruh proyek dari buku,

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

* + 1. Mendownload sebagian maupun seluruh proyek dari internet,

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

* + 1. 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,*

* + 1. Melakukan tindakan kecurangan lainnya,

*Committing other dishonest actions,*

* + 1. 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* |
| 40% | 60% |

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

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

|  |
| --- |
| **Software**  *Software* |
| Microsoft Office 365  SQL Server Developer 2019  SQL Server Management Studio 18.9.1  Visual Paradigm Community Edition 16.3 |

## Ekstensi file yang harus disertakan dalam pengumpulan tugas mandiri dan proyek 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* |
| SQL | SQL, VPP, Image Files (JPG / PNG) |

## Soal

*Case*

**Flokemon Store**

**Flokemon**are creatures that have various shapes and sizes and live in the wild or alongside humans. So as a programmer you are asked to manage the store database.

Every staff that hired by **Flokemon Store** have a task to **serve a customer who wants to buy a flokemon**. Every staff must be following the procedures to become a staff, which are:

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

“STXXX”

X => number between 0 – 9

* Staff can purchase flokemon from supplier
* Every **purchase transaction** made with the supplier have all the information about staff, supplier, transaction date, Flokemon purchased, and the quantity of each Flokemon. Every **purchase transaction** has an identification number with the following format:

“PUXXX”

X => number between 0 – 9

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

“SAXXX”

X => number between 0 – 9

* Everyflokemon sold by **Flokemon Store** have its own name, damage, and price. Every **flokemon** has an identification number with the following format:

“FLXXX”

X => number between 0 – 9

* Everyflokemon has its **type** data that store information about the type name. Every flokemon type has an identification number with the following format:

“FTXXX”

X => number between 0 – 9

Every customer that wants to buy flokemon at **Flokemon Store** must be following the **sales transaction procedures**, those are:

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

“CUXXX”

X => number between 0 – 9

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

Every supplier that wants to sell their flokemon must be following the **purchase transaction procedures**, those are:

* Every supplier that wants to sell their flokemon must already completed personal information like name, email, and phone number. Every supplier has an identification number with the following format:

“SUXXX”

X => number between 0 – 9

* Supplier can sell **more than one flokemon** in every transaction.

**Constraints:**

* Customer name must more than 7 characters.
* Customer Email must end with “.com” (without quote).
* Customer gender must be either “Male” or “Female” (without quote).
* Staff name must more than 7 characters.
* Staff Email must end with “.com” (without quote).
* Supplier name must more than 7 characters.
* Supplier Email must end with “.com” (without quote).
* Flokemon name must more than 5 characters.
* Flokemon price must be between 10000 and 1000000.
* Flokemon Damage must be equal or more than 100.
* Purchase Transaction date must be filled from past or same date as today.
* Purchase Quantity must be equal or more than 1.
* Sales Transaction date must be filled from past or same date as today.
* Sales Quantity must be equal or more than 1.

Now **FlokemonStore** still using manual management system to maintain the **sales** and **purchase transactions**. You as a programmer asked to create a database system that can store data and maintain the **sales** and **purchase transactions**. The tasks that you must do are:

1. Create Entity Relationship Diagram to maintain **sales** and **purchase transactions**.
2. Create a database system using DDL syntax that relevant with **sales** and **purchase 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.

1. Create query using DML syntax to simulate the transactions process for **sales** and **purchase 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 **Flokemon Store**,you are asked to provide some query that resulting important data. The requirements that asked from her are:
2. Display CustomerName, StaffName, and Total Transaction (obtained from the total of sales transaction) for each customer whose name contains character ‘a’ and transaction happened in 2020.
3. Display FlokemonName, FlokemonTypeName, and Total Purchased Flokemon (obtained from the sum of purchased flokemon quantity) for FlokemonName length more than 5 and FlokemonTypeName Contain character 'a'.
4. Display FlokemonTypeName, and Average Price (obtained from average flokemon price and start with ‘Rp. ’), Total Sales Flokemon (obtained from the sum of sold flokemon quantity) for FlokemonTypeName start with character 'p' and have damage more than 1000. Display the data in ascending order.
5. Display CustomerName, Total Transaction (obtain from the total number of different flokemon bought in one transaction), and Total Flokemon Bought (obtained from total transaction quantity of the sales transaction) for every transaction that happened in October and Total Flokemon quantity more than 10 with each customer whose name contains character ‘a’.
6. Display SalesID, CustomerName, StaffName, and Transaction Day (obtained from the day name of transaction date) for each transaction made by staff that has above average salary (obtained from average salary of all staffs) and was done in February.

**(alias subquery)**

1. Display StaffName (obtained from upper case format of StaffName), FlokemonName, Transaction Year (obtained from the year of transaction date) for each transaction that has a transaction quantity more than the average of all the transaction quantity in 2018 and staff whose name contain character ‘s’.

**(alias subquery)**

1. Display Transaction Date and Max Flokemon Sales (obtained from max quantity of flokemon sales transaction and ends with ' Flokemon') for each Flokemon that are sold between July and December in 2020 by staff name contain character 's'.

**(alias subquery)**

1. Display Transaction Date and Max Flokemon Purchased (obtained from max quantity of flokemon purchase transaction and ends with ' Flokemon') for each Flokemon that are bought between January and October in 2018 by staff name contain character 'o'.

**(alias subquery)**

1. Create a view named PurchaseView that display SupplierName, Supplier Phone (obtained by replacing 0 in front of the phone number with '+62'), Total Transactions (obtained from total number of different flokemon bought in one transaction), Total Flokemon Bought (obtained from total of transaction quantity in Purchase transaction), and for each Supplier that has made between 1 and 5 transactions and Supplier name length less than 15.
2. Create a view named StaffSalesView that display StaffName, Total Transactions (obtained from total number of different flokemon sold in one transaction), Total Flokemon Sold (obtained from total of transaction quantity in Sales transaction), and for each staff sell more than 10 Flokemon, StaffName length less than 15.

**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