|  |  |
| --- | --- |
| **Project Case** |  |
| ISYS6084 | ISYS6123 | ISYS6123003 | ISYS6169 | ISYS6169001  Database | Introduction to Database Systems | Database Systems |
| **Information Systems** | **E222-ISYS6123003-JP01-00** |
| ***Valid on*** *Even 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* |
| 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*

**JP Gaming Store**

**JP Gaming Store** is a local gaming store that just recently opened. JP Gaming Store uses an old filesystem to track transactions done in the store such as **selling products to a customer** and **purchasing components from a supplier**.

**Staff** in JP Gaming Store have a task to **serve a customer who wants to buy products** and **purchase components from suppliers**. Every staff must be following the procedures to become a staff, which are:

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

“STXXX”

X => number between 0 – 9

* Staff can purchase components with a vendor.
* Every **component transaction** made with the vendor has all the information about staff, vendor, transaction date, components purchased, and the quantity of each component. Every **component transaction** has an identification number with the following format:

“CTXXX”

X => number between 0 – 9

* Every component purchased from the supplier has its name, description, and price. Every **component** has an identification number with the following format:

“COXXX”

X => number between 0 – 9

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

“PTXXX”

X => number between 0 – 9

* Every **product** sold by JP Gaming Storehas its name, description, weight, and price. Every **Product** has an identification number with the following format:

“PRXXX”

X => number between 0 – 9

* Every product has its **product** **category** data that has its category name. Every **Product Category** has an identification number with the following format:

“PCXXX”

X => number between 0 – 9

Every customer that wants to buy a product at JP Gaming Storemust be following the **product transaction procedures**, those are:

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

“CUXXX”

X => number between 0 – 9

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

Every vendor that wants to sell their component must be following the **component transaction procedures**, those are:

* Every vendor that wants to sell their ingredients must have complete personal information like name, email, phone, date of birth, gender, and address. Every vendor has an identification number with the following format:

“VEXXX”

X => number between 0 – 9

* Vendors can sell **more than one component** in every transaction.

**Constraints:**

* Customer email must end with “gmail.com” or “yahoo.com” (without quote).
* Customer and staff addresses must have more than 1 word.
* Customer and vendor names must be more than 3 characters.
* Component and product price must be more than 5000.
* Product transaction and component transaction Quantity must be more than 0.
* Customer, vendor, and staff gender must be either “male” or “female” (without quote).
* Customer phone must start with “+62 “(without quote).
* Product weight must be between 1 and 40.

Now JP Gaming Storeisstill using an old file system to maintain the **product** and **component transactions**. You as his precious friend want to help **JP Gaming Store** create a database system that can store data and maintain the **product** and **component transactions**. The tasks that you must do are:

1. Create Entity Relationship Diagram to maintain **product** and **component** **transactions**.
2. Create a database system using DDL syntax that is relevant to **product** and **component** **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 to 10 data.
* **Transaction** table must be filled with more than or equals to 15 data.
* **Transaction detail** table must be filled with more than or equals to 25 data.
* For the **Product Category** table, the table must be filled with the following data:

|  |  |
| --- | --- |
| Product Category Names | |
| Chair | Keyboard |
| Mouse | Table |
| Headphone | Speaker |
| Monitor | Laptop |
| Computer | Accessories |

1. Create a query using DML syntax to simulate the transactions process for **product** and **component transactions**.

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

1. To support the database management process in **JP Gaming Store**,JP asked you to provide some queries, resulting in important data. The requirements are:
2. Display CustomerID, CustomerName(obtained from CustomerName in uppercase format), and TotalItemsBought (obtained from the sum of quantity) for every transaction that happens in the year 2021 and done by a customer that is born in 2001.
3. Display ProductID, ProductName, and ProductCategoryName for every transaction that occurred in the year 2020 and product that is purchased at least in 2 transactions.
4. Display ProductID, ProductName (obtained from ProductName in uppercase format), Income (obtained from the sum of ProductPrice times quantity), TotalTransaction (obtained from the total number of different products that have been bought) for every product which CategoryName is 'mouse' or 'keyboard' and ProductName contains ‘gaming’, ordered by Income in descending.
5. Display StaffID, StaffName, UserName (obtained from Email before ‘@’), TotalTransaction (obtained from the total number of different products sold by staff), MoneySpend (obtained from the sum of MaterialPrice times quantity) for every staff that is male and for every transaction that is done in 2019.
6. Display VendorNameID (obtained from VendorName and 3 last characters from VendorID), VendorPhone, YearOfBirth (obtained from the year of VendorDOB), and TransactionCount (obtained from the total number of different components sold by the vendor) for every transaction item that has a quantity of more than 60 and done by the female vendor, ordered by TransactionCount in descending.

(**alias subquery**)

1. Display ComponentID, ComponentKeyword (obtained from getting the first word from the component description) for every component that has a price more than the average price and for every transaction that happened in 2020.

(**alias subquery**)

1. Display ProductIdentifier (obtained from 3 last characters from ProductID and 3 first characters of ProductName), ProductWeight, ProductPrice, TransactionCount (obtained from the total number of different products sold) for every transaction that occurred after 2019 and has TransactionCount more than the average of TransactionCount.

(**alias subquery**)

1. Display CustomerName, EmailProvider (obtained from CustomerEmail after ‘@’ character), TotalQty (obtained from the sum of product quantity the customer has bought) for every male customer that has TotalQty more than average TotalQty.

**(alias subquery)**

1. Create view StaffCompactView to display StaffName, StaffPhone (obtained from replacing ‘+62’ from StaffPhone by ‘0’), YearOfBirth (obtained from the year of StaffDOB), Earning (obtained from the sum of ProductPrice times quantity), and TotalTransactions (Obtained from total number of different products sold by staff) for every female staff and product with weight more than 0.
2. Create view CustomerFLView to display CustomerID, CustomerName (obtained from CustomerName in a lowercase format), FirstTransaction (obtained from the Customer's first ProductTransactionDate), LastTransaction (obtained from the Customer's last ProductTransactionDate) for every customer that has more than one word in his/her name and that made transaction with staff with salary more than 7000000, ordered by CustomerName ascending.

**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