|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Practicum Final Exam – Odd Semester Year 2021/2022** | | | | | | |
| **Subject** | | | **ISYS6169001 – Database Systems** | | |  |
| **Class** | **:** | **BM01 / BN01** | | **Start Date** | **: 13 January 2022** |
| **Lecturer** | **:** | **D5104 - Dr. Robertus Nugroho Perwiro Atmojo, S.Kom., M.M.S.I, CSCA** | | **Start Time** | **: 11.20** |
| **End Date** | **: 13 January 2022** |
| **End Time** | **: 13.20** |

**PERATURAN UJIAN:**

*Exam Regulations:*

* Mahasiswa tidak diperbolehkan berdiskusi dan/atau bekerja sama dengan peserta ujian lainnya

*Student is not allowed to discuss and/or work together with other exam participants*

* Mahasiswa tidak diperbolehkan untuk membuka dan menyalin dari **BUKU** atau **CATATAN**, **VIDEO** dari pengajar (recording kelas, VBL, Youtube, dsb) dan **REFERENSI** lainnya

*Student isn't allowed to open and copy from any resources such as notes, videos (class recording, VBL, Youtube, etc) and other references*

* Mahasiswa tidak diperbolehkan membuka dan menyalin jawaban dari internet (google, stackoverflow, dsb)

*Student isn't allowed to open and copy answer from the internet (google, stackoverflow, etc)*

* Asisten **BERHAK** memberi nilai 0 **(NOL)** bagi peserta ujian yang melakukan segala bentuk kecurangan

*Assistant is able to give 0 (ZERO) score for exam participant who does any cheating actions*

* Kumpulkan jawaban tepat pada waktunya, apabila terlambat mengumpulkan maka jawaban tidak akan dikoreksi dan nilai mahasiswa adalah 0

*Submit the answer on time, if not, then the answer will not be checked, and the students will receive 0 (ZERO)*

* 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*



**SOFTWARE YANG DIGUNAKAN:**

*Software will be used:*

* SQL Server Developer 2019
* SQL Server Management Studio 18.9.1

**FILE YANG DIKUMPULKAN:**

*File must be collected:*

* SQL

**PERHATIAN!**

*Attention!*

* Bagi yang mengerjakan tidak sesuai dengan soal, maka akan diberikan nilai **NOL (0)**

*For those who do not work in accordance with the exam case will be marked as* ***ZERO (0)***

* Bagi yang mengerjakan tidak sesuai dengan software dan versi yang telah ditetapkan, maka akan tetap dikoreksi dengan software dan versi yang telah ditetapkan

*For those who do not work in accordance with the software and specific version will be corrected by the predefined software and version*

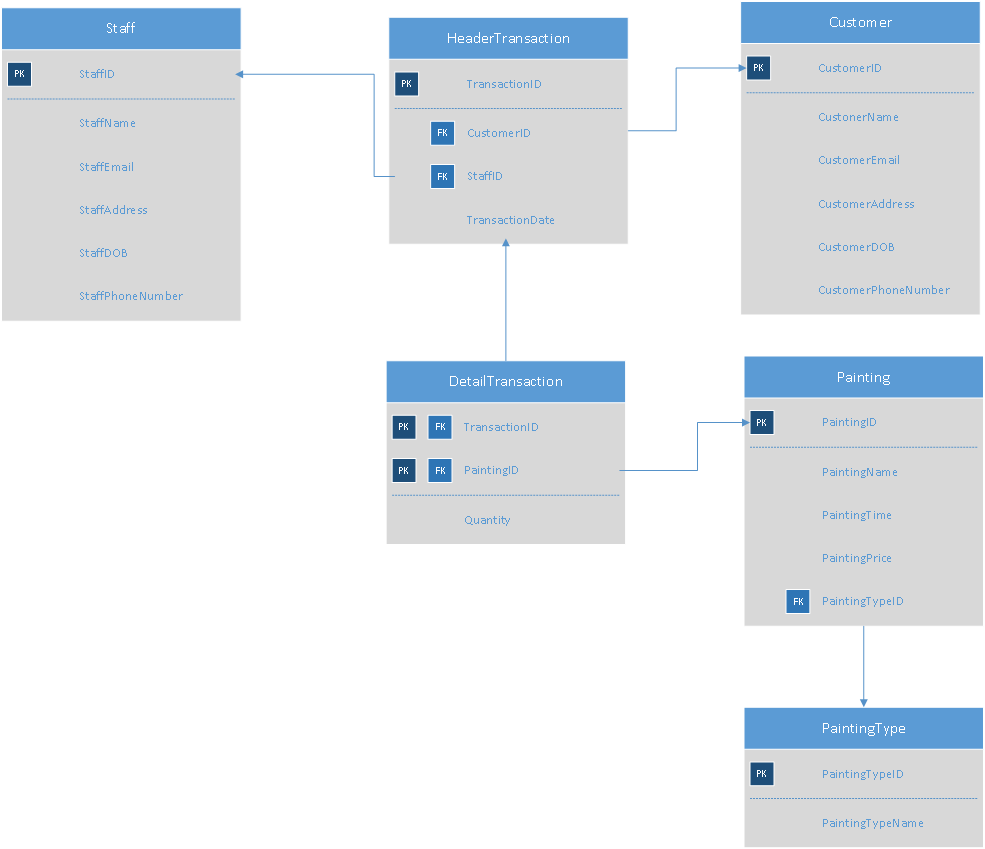
* Kompres semua jawaban yang akan diunggah. Pastikan format pengumpulan nama file dan ekstensi sesuai dengan format berikut: **[NIM]-[NAMA].zip**

*Compress all file that will be uploaded. Make sure the format for collecting file name and extension according to the following format:* ***[NIM]-[NAME].zip***

**Tabel Relasional**

*Relational Table*

**Bluejack Painting**



**Soal**

*Case*

1. Display **PaintingID**, **PaintingName**, and **PaintingPrice** for every Painting whose **name** **is more than 15 characters.**

(**len**)



1. Display **TransactionID**, **CustomerName**, and **TransactionDate** for every transaction which **occurs in** **September**.

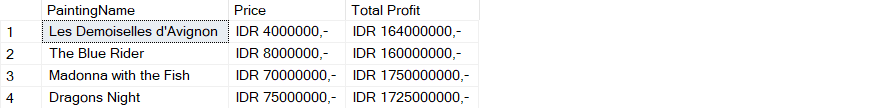
(**datename, month, join**)

A picture containing shape

Description automatically generated

1. Display **PaintingName**, **Price** (obtained from **adding ‘IDR ’ in front of** **PaintingPrice** and **‘,-’** **at the end of** **PaintingPrice**) and **Total Profit** (obtained from **total** **multiplication of quantity** **and PaintingPrice,** then **add** ‘**IDR ’** **in front of** **Total Profit** and **‘,-’** **behind** **Total Profit**) for every Painting which **name** **contains ‘d’ character.**

**(cast, sum, join, like)**

****

1. Display **CustomerName**, **TransactionID**, **TransactionDate**, **Total of** **Painting** (obtained from the **total number of paintings** bought per transaction), and **Total of** **Quantity** (obtained from the **total painting quantity** the customer bought per transaction) for every Customer who is **born in odd year**. Then, **combine** it with **CustomerName**, **TransactionID**, **TransactionDate**, **Total** **of** **Painting** (obtained from the **total number of paintings** bought per transaction), and **Total of Quantity** (obtained from the **total painting quantity** the customer bought per transaction) for every Customer whose **name is more than 10 characters**.

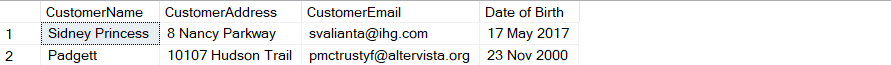
(**count, sum, year, union, len**)

Table

Description automatically generated

1. Display **CustomerName**, **CustomerAddress**, **CustomerEmail**, and **Date of Birth** (obtained from **CustomerDOB in ‘dd mon yyyy’ format)** for every customer who has done transaction in **July** and has purchased **painting** whose **type is ‘Abstract’**.

(**convert, month, in**)



1. Display **unique** **CustomerID**, **CustomerName**, **CustomerPhoneNumber**, and **CustomerEmail** for every customer who has purchased **any painting which quantity is more than** the **average** **of all** **painting** **purchase** and whose **name** **is more than one word.** After that, **sort** the data based on **CustomerName ascendingly.**

(**distinct, alias subquery, avg, charindex, order by**)

Graphical user interface

Description automatically generated with low confidence

1. Create a **view** named **‘StaffOddView’** to display **StaffID** (obtained from the **last** **3** **character** of **StaffID** and **add 'Staff '** **in front of** **it**), **StaffName**, and **StaffAddress** for every Staff whose **id is an odd number.**

(**create view,** **convert, right**)

Graphical user interface, application

Description automatically generated

1. Create a **view** named **‘TransactionDataOfModernismPainting’** to display **TransactionID**, **Day of Transaction** (obtained from the **weekday** of **transaction made**), **TransactionDate**, **Painting Title** (obtained by **PaintingName in uppercased** format) and **Quantity** (obtained from the **total of the painting quantity sold per transaction**) for every transaction which the **painting’s type** **name** is **‘Modernism’**.

(**create view, datename, weekday, upper, sum**)

Graphical user interface, table

Description automatically generated

1. Add a **column** named **‘CustomerSocialMedia’** on **Customer** table with **varchar** **(30)** data type. Then, add a **constraint** named **‘CheckCustomerSocialMedia’** in **Customer** table to check whether the **CustomerSocialMedia** is **between** **5** and **25** **characters**.

(**alter table, add, add constraint, between**)

Graphical user interface

Description automatically generated

1. **Delete** the datafrom **Customer** table for every customer which **the quantity of any painting purchased is** **more** **than** **6 in a transaction** and that **transaction** which **occurs** in **even** **month.**

(**delete, month**)

**Before:**

Table

Description automatically generated

**After:**

Table

Description automatically generated with medium confidence