|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Practicum Final Exam – Odd Semester Year 2021/2022** | | | | | | |
| **Subject** | | | **ISYS6169001 – Database Systems** | | | Diagram, schematic  Description automatically generated |
| **Class** | **:** | **BK01 / BL01** | | **Start Date** | **: 18 January 2022** |
| **Lecturer** | **:** | **D6526 - Nikolaus Permana Tri Rahmanto, S.Kom, M.M.S.I.** | | **Start Time** | **: 13:20** |
| **End Date** | **: 18 January 2022** |
| **End Time** | **: 15:20** |

**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*

Logo, company name

Description automatically generated

**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

**Tabel Relasional**

*Relational Table*

**ettayLor**

Diagram

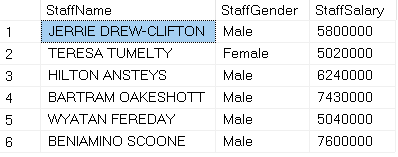
Description automatically generated

**Soal**

*Case*

1. Display **StaffName** (obtained from StaffName in **Uppercased Format**), **StaffGender**, and **StaffSalary** for every staff whose **salary** is **more than 5000000**.

(**upper**)



1. Display **SalesDate**, **CustomerName**, **CustomerAddress**, and **CustomerPhone** for every transaction that **occurs in ‘September’.**

(**join, datename, month**)



1. Display **ClothesId** (obtained from **replacing** ‘**CL’** to **‘CLOTHES ‘**), **ClothesName**, **ClothesPrice** (obtained from **adding ‘Rp. ’** in **front** of the **ClothesPrice**), and **Quantity** for every clothes that is **sold more than 3**.

(**replace, concat, join, sum**)

Table

Description automatically generated

1. Display **CustomerId**, **CustomerName**, **Average Quantity** (obtained from the **average** **clothes’ quantity** the customer bought), and **Total Transaction** (obtained from the **number of transactions** **made** by the customer) for every customer whose **home address** **ends with ‘Street’** and has **age more than 20** **at the time they do the transaction**. Then, **combine** it with **CustomerId**, **CustomerName**, **Average Quantity** (obtained from the **average clothes’ quantity** the customer bought), and **Total Transaction** (obtained from the **number of transactions made** by the customer) for every customer whose **home address ends with ‘Hill’** and has **age more than 20** **at the time they do the transaction.**

(**avg, count, like, datediff, year, union**)

Table

Description automatically generated

1. Display **ClothesName**, **ClothesTypeName**, **ClothesPrice**, and **Sold in Month** (obtained from the **name** ofthe **sales’ month**) for every clothes which **type** is either **Dress** or **Blouse**, and its **price** is **more than 300000.**

(**distinct, datename, month, in**)

Application, table, Teams

Description automatically generated

1. Display **StaffId**, **StaffName** (obtained from the **Staff’s last name**), **StaffSalary**, **SalesId**, **SalesDate**, and **Quantity** for every staff that has sell clothes **more than** the **average quantity of any clothes sold** in a single transaction and has **salary** **less than 5000000.**

(**substring, charindex, len, alias subquery, avg**)

Table

Description automatically generated

1. Create a **view** named ‘**Female Staffs**’ to display **StaffName** (obtained from **adding** **‘Ms. ’** and **Staff’s first name**), **StaffAddress**, **StaffPhone**, **StaffDOB**, and **StaffGender** for every **female staff.**

(**create view, left, charindex**)

Table

Description automatically generated

1. Create a **view** named **‘Customer that get discount’** to display **CustomerName**, **CustomerAddress**, **CustomerPhone**, **CustomerDOB**, and **Total Transaction** (obtained from the **calculation** **of clothes price and quantity** that has been bought) for every transaction that **occurs in 2021** and its **total transactions more than 700000**. After that, **sort** the data by **total transaction descendingly**.

(**create view, sum, year, order by**)

Table

Description automatically generated

1. **Add a column** named **ClothesDescription** on **Clothes** table with **varchar (255)** data type. Then, **add a constraint** named **DescriptionLength** to check the **length** **of ClothesDescription** must be **more than 10 characters**.

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

Table

Description automatically generated

1. Update **clothes** data by **reducing** its **price** **by** **20000** on **MsClothes** table for every clothes which **size** is **L, XL, or XXL** and the **quantity** **sold is less than 3**.

(**update, in**)

**Before** update:

Table

Description automatically generated

**After** update:

Table

Description automatically generated