Homework 5
Daniel Kim
Ruijiang Ma

1.

SHOW TABLES from information_schema;

SELECT TABLE_NAME, SUM(TABLE_ROWS)
      FROM information_schema.TABLES
      GROUP BY TABLE_NAME;

```
2 •     SELECT TABLE_NAME AS 'Table Name', SUM(TABLE_ROWS) AS 'Total Rows'
3           FROM information_schema.TABLES
4           GROUP BY TABLE_NAME;
5
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: IA

Table Name	Total Rows
DimAccount	99
DimCurrency	0
DimCustomer	18356
DimDepartmentGroup	7
DimEmployee	296
DimGeography	655
DimOrganization	14
DimProduct	158
DimProductCategory	4
DimProductSubcategory	37
DimPromotion	16
DimReseller	701
DimSalesReason	10
DimSalesTerritory	11
DimScenario	3
DimTime	1158
FactCurrencyRate	0
FactFinance	38480
FactInternetSales	59800
ADMINISTRABLE_ROLE...	0
APPLICABLE_ROLES	0
CHARACTER_SETS	0
CHECK_CONSTRAINTS	0
COLLATIONS	0
COLLATION_CHARACT...	0
COLUMNS	0

2.

SELECT TABLE_NAME AS 'Table Name', COLUMN_NAME AS 'Primary Key'
  FROM COLUMNS WHERE column_key = 'PRI';

```
5 •      use information_schema;
6 •      SELECT TABLE_NAME AS 'Table Name', COLUMN_NAME AS 'Primary Key'
7              FROM COLUMNS WHERE column_key = 'PRI';
```
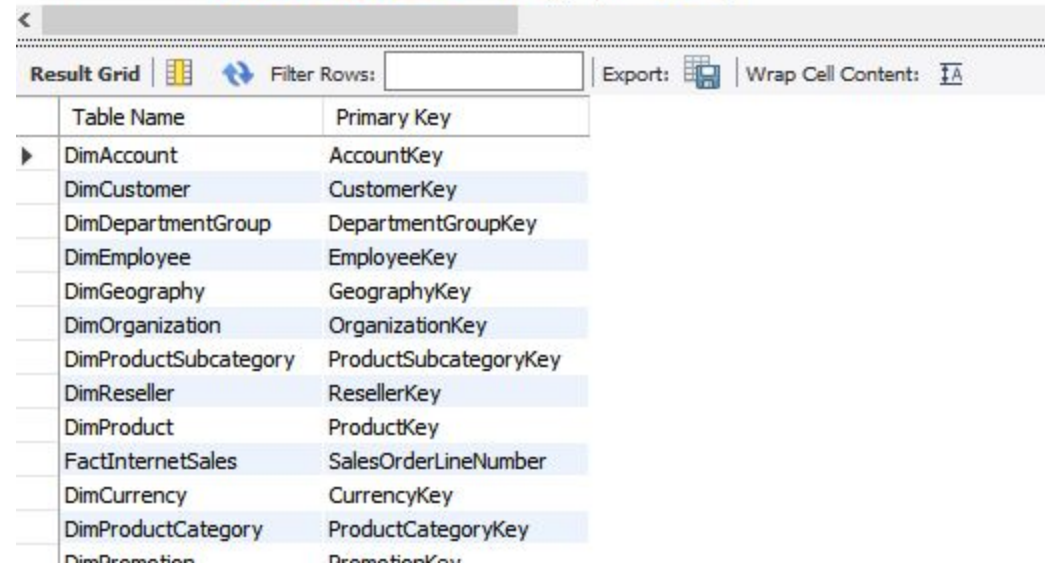
Result Grid | 🔢 | ↔ Filter Rows: [   ] | Export: 🔣 | Wrap Cell Content: 🄰

Table Name	Primary Key
DimAccount	AccountKey
DimCustomer	CustomerKey
DimDepartmentGroup	DepartmentGroupKey
DimEmployee	EmployeeKey
DimGeography	GeographyKey
DimOrganization	OrganizationKey
DimProductSubcategory	ProductSubcategoryKey
DimReseller	ResellerKey
DimProduct	ProductKey
FactInternetSales	SalesOrderLineNumber
DimCurrency	CurrencyKey
DimProductCategory	ProductCategoryKey
DimPromotion	PromotionKey

3.

They use standard table naming convention using prefix by adding in Dim in the front for dimension tables and adding fact in the front for fact table.
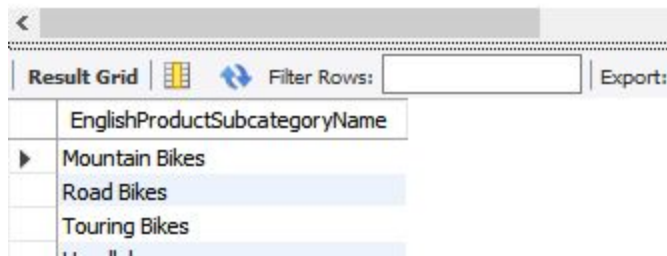
4.

The purpose of the recursive relationship of DimEmployee is that it is related to the employee-manager hierarchy because of all the information in each employee.

5.

```
USE aw;
SELECT EnglishProductSubcategoryName
   FROM DimProductSubcategory;
```

```
11 •    USE aw;
12 •    SELECT EnglishProductSubcategoryName
13          FROM DimProductSubcategory;
```

| Result Grid | | Filter Rows: | | Export: |

EnglishProductSubcategoryName
Mountain Bikes
Road Bikes
Touring Bikes

Mountain Bikes
Road Bikes
Touring Bikes

6.

USE aw;
SELECT d.CalendarYear, COUNT(p.Color) AS ColorCount, p.Color
   FROM DimTime d, FactInternetSales f, DimProduct p, DimProductSubcategory s
   WHERE d.CalendarYear BETWEEN "2001" AND "2004"
   AND p.ProductSubcategoryKey = s.ProductSubcategoryKey
   AND f.ProductKey = p.ProductKey
   AND d.TimeKey = f.OrderDateKey
   AND s.EnglishProductSubcategoryName LIKE "%Bikes%"
   GROUP BY d.CalendarYear, p.Color
   ORDER BY d.CalendarYear, count(p.Color) DESC;

```
13 ●    USE aw;
14 ●    SELECT d.CalendarYear, COUNT(p.Color) AS ColorCount, p.Color
15          FROM DimTime d, FactInternetSales f, DimProduct p, DimProductSubcategory s
16          WHERE d.CalendarYear BETWEEN "2001" AND "2004"
17          AND p.ProductSubcategoryKey = s.ProductSubcategoryKey
18          AND f.ProductKey = p.ProductKey
19          AND d.TimeKey = f.OrderDateKey
20          AND s.EnglishProductSubcategoryName LIKE "%Bikes%"
21          GROUP BY d.CalendarYear, p.Color
22          ORDER BY d.CalendarYear, count(p.Color) DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

CalendarYear	ColorCount	Color
2001	775	Red
2001	154	Black
2001	84	Silver
2002	1380	Red
2002	868	Black
2002	283	Silver
2002	146	Yellow
2003	2321	Black
2003	1268	Yellow
2003	1119	Silver
2003	501	Red
2003	501	Blue
2004	1966	Black
2004	1789	Yellow
2004	1205	Silver
2004	782	Blue
2004	63	Red

In 2001 and 2002, Red color was the most popular color of bikes, and in 2003 and 2004, Black color was the most popular color of bikes

7.

```
USE aw;
SELECT d.CalendarYear, g.StateProvinceName, SUM(f.UnitPrice) AS AmountSpent
   FROM DimTime d, FactInternetSales f, DimProduct p, DimProductSubcategory s,
      DimSalesTerritory t, DimGeography g
   WHERE d.CalendarYear BETWEEN "2001" AND "2004"
   AND p.ProductSubcategoryKey = s.ProductSubcategoryKey
   AND f.ProductKey = p.ProductKey
   AND d.TimeKey = f.OrderDateKey
   AND s.EnglishProductSubcategoryName LIKE "%Bikes%"
   AND f.SalesTerritoryKey = t.SalesTerritoryKey
   AND t.SalesTerritoryKey = g.SalesTerritoryKey
   GROUP BY d.CalendarYear, g.StateProvinceName
   ORDER BY d.CalendarYear, AmountSpent DESC;
```

In 2001, California, New South Wales, Washington, and Texas showed the highest sales volume in order

In 2002, California, New South Wales, Washington, and England showed the highest sales volume in order

In 2003, California, England, New South Wales, and Washing showed the highest sales volume in order

In 2004, California, England, Washington, and New South Wales showed the highest sales volume in order.

8.

```
USE aw;
SELECT p.ModelName, SUM(f.UnitPrice - f.ProductStandardCost) AS ProfitMargin
   FROM DimTime d, FactInternetSales f, DimProduct p, DimProductSubcategory s
   WHERE d.CalendarYear = "2002"
        AND p.ProductSubcategoryKey = s.ProductSubcategoryKey
   AND f.ProductKey = p.ProductKey
   AND d.TimeKey = f.OrderDateKey
   AND s.EnglishProductSubcategoryName LIKE "%Bikes%"
   GROUP BY p.ModelName
   ORDER BY ProfitMargin DESC;
```

```
37 •    SELECT p.ModelName, SUM(f.UnitPrice-f.ProductStandardCost) AS ProfitMargin
38          FROM DimTime d, FactInternetSales f, DimProduct p, DimProductSubcategory s
39          WHERE d.CalendarYear = "2002"
40          AND p.ProductSubcategoryKey = s.ProductSubcategoryKey
41          AND f.ProductKey = p.ProductKey
42          AND d.TimeKey = f.OrderDateKey
43          AND s.EnglishProductSubcategoryName LIKE "%Bikes%"
44          GROUP BY p.ModelName
45          ORDER BY ProfitMargin DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ModelName	ProfitMargin
Road-150	1159368.00
Road-250	610386.00
Mountain-200	371476.00
Mountain-100	330482.00
Road-650	117236.00
Road-550-W	57524.00

For the year 2002, Road-150 bike model yielded the highest margin for AdventureWorks