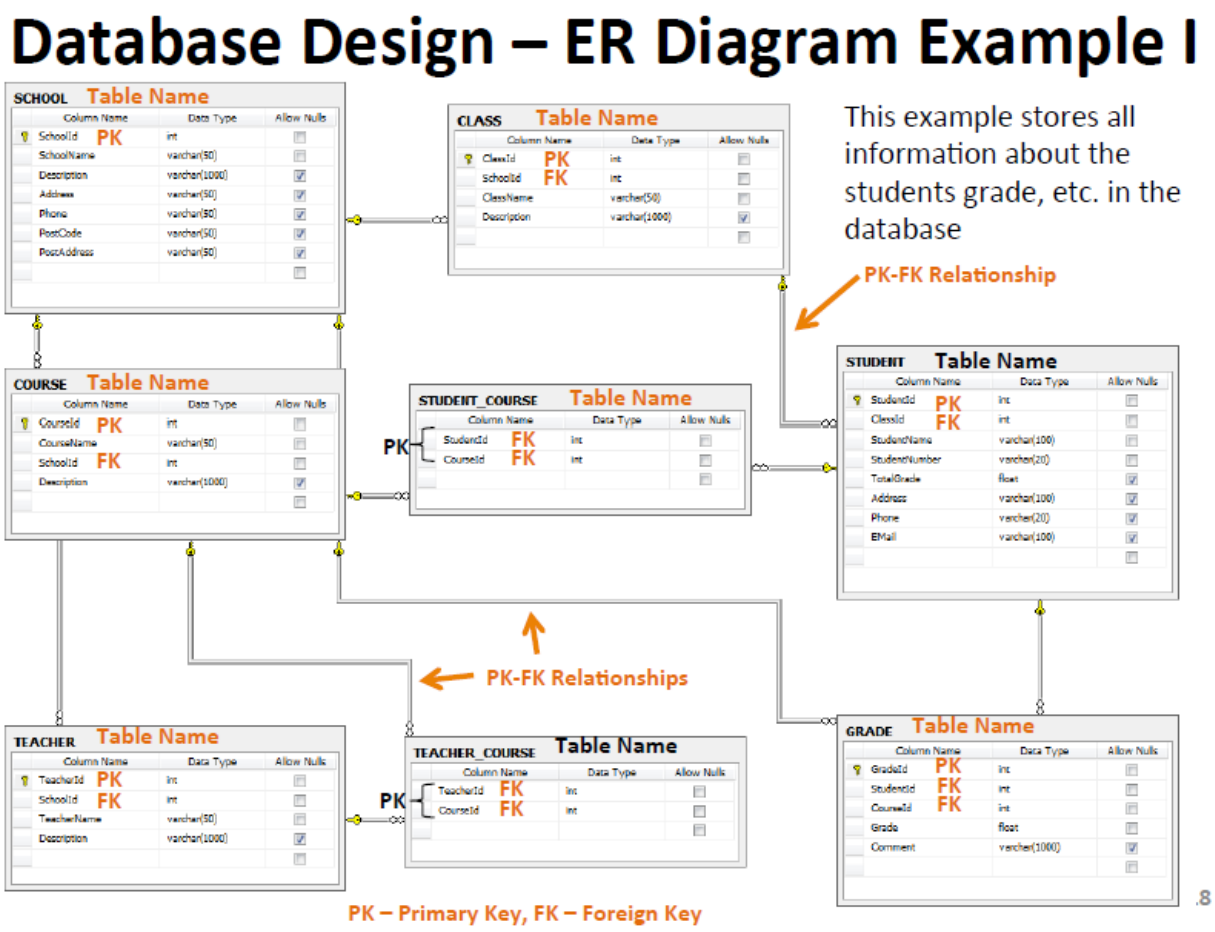
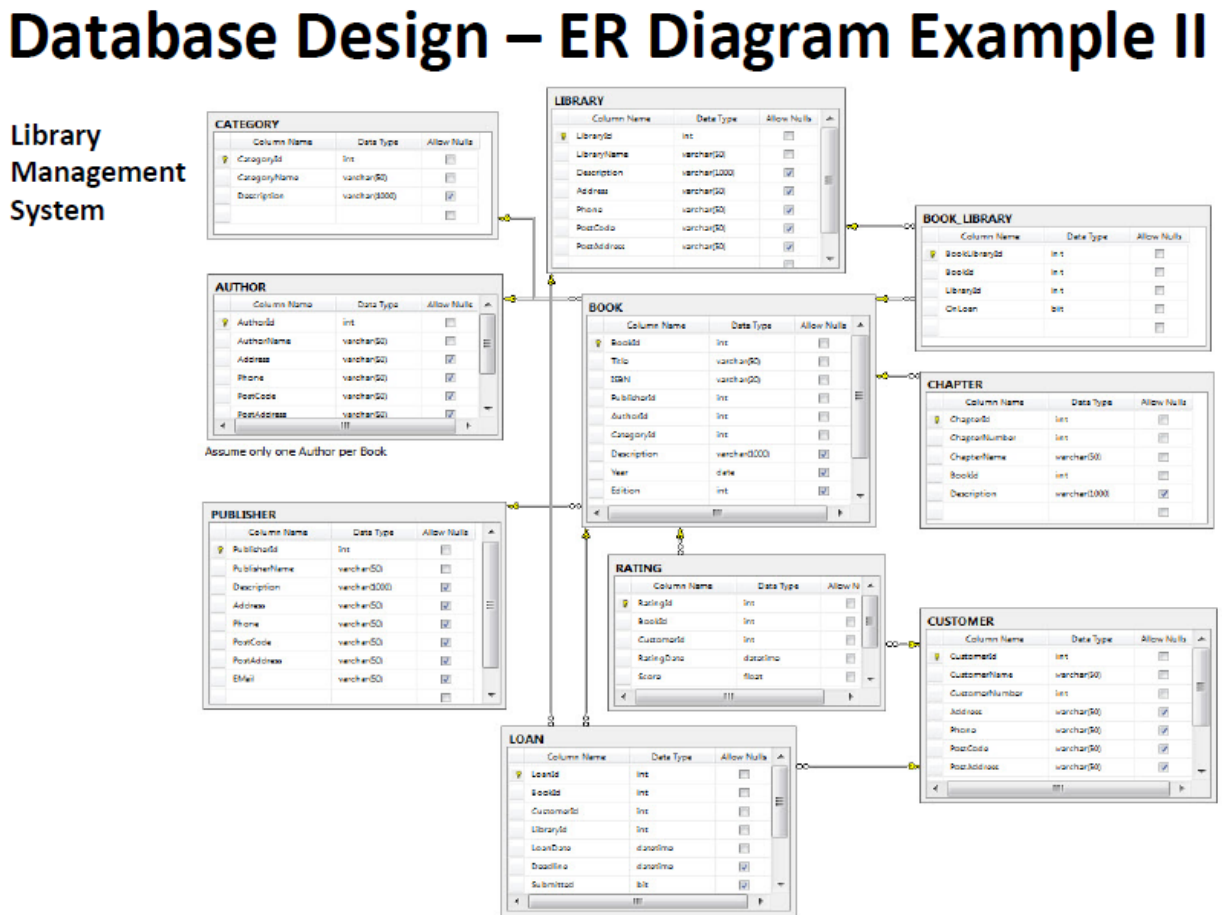
OPPORTUNITIES

CREATE A SCHOOL DATABASE SIMILAR TO THIS

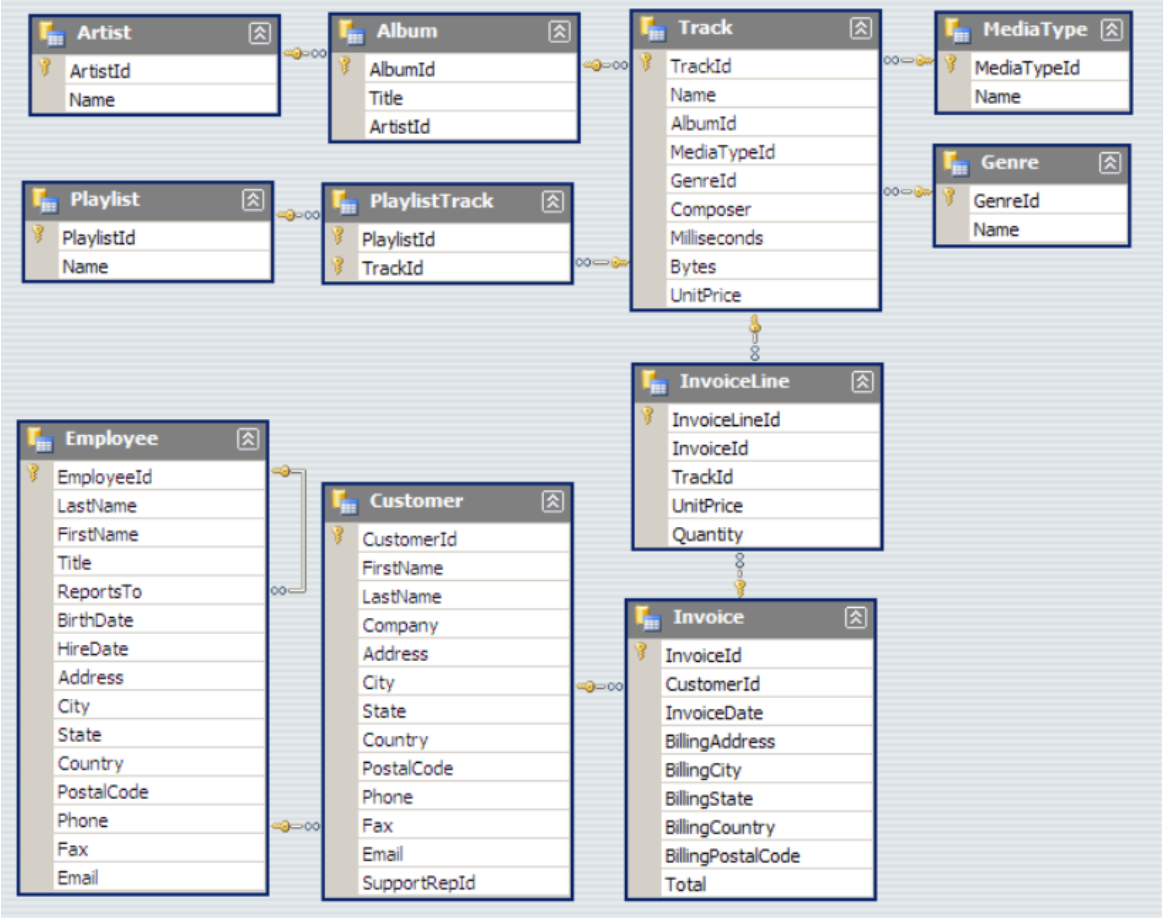


CREATE A LIBRARY MANAGEMENT SYSTEM DATABASE SIMILAR TO THIS.



TABLES ASSIGNMENT

CREATE A TABLE STRUCTURE SIMILAR TO THIS :



```
SELECT
C.EnglishProductCategoryName,S.EnglishProductSubcategoryName
,
P.EnglishProductName,F.SalesAmount
 FROM DimProductCategory C
INNER JOIN DimProductSubcategory S
ON C.ProductCategoryKey = S.ProductCategoryKey
INNER JOIN DimProduct P
ON S.ProductSubcategoryKey = P.ProductSubcategoryKey
INNER JOIN FactInternetSales F
ON P.ProductKey = F.ProductKey

SELECT C.EnglishProductCategoryName,
S.EnglishProductSubcategoryName,D.CalendarYear,
SUM(F.SALESAMOUNT) AS [TOTAL-SALES]
```

```
FROM DIMPRODUCTCATEGORY C
INNER JOIN DimProductSubcategory S ON
C.ProductCategoryKey=S.ProductCategoryKey
INNER JOIN DIMPRODUCT P ON
S.ProductSubcategoryKey=P.ProductSubcategoryKey
INNER JOIN FACTINTERNETSALES F ON F.PRODUCTKEY=P.PRODUCTKEY
INNER JOIN DIMDATE D ON D.DateKey = F.OrderDateKey
GROUP BY
C.EnglishProductCategoryName,S.EnglishProductSubcategoryName
,D.CalendarYear
```

QUERIES ASSIGNMENT [Using AdventureWorksDW database]

1. Write a query to find Sales Amount of each customer.
2. Write a query to find Sales Amount of each product.
3. Write a query to find no of customers for each product.

4. Write a query to find the top 3 customers with highest sales amount.
5. Write a query to find the 3rd highest customers (eg : the customer who is the third highest salesamount)
6. Write a query to find the total sales amount for each category

```
SELECT
C.EnglishProductCategoryName,SUM(F.SALESAMOU
NT) AS [TOTAL-SALES]
FROM DIMPRODUCTCATEGORY C
INNER JOIN DimProductSubcategory S ON
C.ProductCategoryKey=S.ProductCategoryKey
INNER JOIN DIMPRODUCT P ON
S.ProductSubcategoryKey=P.ProductSubcategory
Key
INNER JOIN FACTINTERNETSALES F ON
F.PRODUCTKEY=P.PRODUCTKEY
GROUP BY C.EnglishProductCategoryName
```

7. Write a query to find the total sales amount for each subcategory

```
SELECT
S.EnglishProductSubcategoryName,SUM(F.SALESAMO
UNT) AS [TOTAL-SALES] FROM
DimProductSubcategory S
    INNER JOIN DIMPRODUCT P ON
    S.ProductSubcategoryKey=P.ProductSubcategory
    Key
    INNER JOIN FACTINTERNETSALES F ON
    F.PRODUCTKEY=P.PRODUCTKEY
    GROUP BY S.EnglishProductSubcategoryName
```

8. Write a query to find the total sales amount for each product
9. Write a query to find the total sales amount for each category, subcategory & product.
10. Write a query to find the product having the highest sales amount.

11. Write a query to find the product having the fifth highest sales amount.
12. Write a query to find the product which have zero sales amount or not have been sold
13. Write a query to find the total sales amount for each Territory.

14. Write a query to find the total sales amount from 1st Jan 2005 to 31st Dec 2011.
15. Write a query to find the customer having highest sales amount from 1st Jan 2005 to 31st Dec 2011.
16. Write a query to find each year total sales amount.
17. Write a query to find each month total sales amount for each year.

FUNCTIONS ASSIGNMENT [Using AdventureWorksDW database]

---

1. Write a function to add two numbers.
2. Write a function to add /sub/mul/div based upon the mode passed. (Eg: user will pass two numbers and mode, (5, 6,"Add") accordingly it should return value).
3. Write a function to find the total number of customers for a product. User will enter a productname, based upon the name it should return the number of customers for that product.
4. Write a function to find the sales amount for given date range. User will enter the start date and end date and accordingly it should return the sales amount.
5. Write a function to find the sales amount for given Year. User will enter the Year accordingly it should return the sales amount.

---

6. Write a function to find the sales amount for given Year and Month. User will enter the Year and Month and accordingly it should return the total sales amount.
7. Write a function to get the top n product sales amount based upon number entered by user. For eg If user enter 7 it should be 7^{th} highest, if user enter 5 it should be 5^{th} highest.
8. Write a function to calculate profile. Profit will be salesamount – tax.

VIEWS ASSIGNMENT [Using AdventureWorksDW database]

---

1. Create a view for category, subcategory, and product sales details.
2. Create a view for customer, category, subcategory, product sales details.
3. Create a view for territory, category, subcategory, product sales details.
4. Create a View for Prospective Buyer which has owned a house.
5. Create a View for Prospective Buyer which have a single car.

SQL SERVER STORED PROCEDURE ASSIGNMENTS [eg : AdventureWorksDW database]

1. WRITE A STORED PROCEDURE TO FIND THE TOTAL SALES OF A PRODUCT.
   THE USER WILL ENTER THE PRODUCT NAME AND IT SHOULD RETURN THE TOTALSALESAMOUNT AS OUTPUT.

```
ALTER PROCEDURE SP_PRODUCTNAME
@NAME NVARCHAR(100)
AS
BEGIN
SELECT SUM(SalesAmount)  FROM FactInternetSales F INNER
JOIN DimProduct P
ON F.ProductKey= P.ProductKey WHERE P.EnglishProductName
= @NAME
END
```

2. WRITE A STORED PROCEDURE TO FIND THE PROUCT KEY AND PRODUCTNAME FOR THE CATEGORY.
   The user will enter the categoryname and it will return the list of products for that category.

```
CREATE PROCEDURE SP_GETPRODUCTS
@NAME NVARCHAR(100)
AS
BEGIN
SELECT * FROM DimProductCategory C INNER JOIN
DimProductSUBCategory S
ON C.ProductCategoryKey = S.ProductCategoryKey
INNER JOIN DimProduct P
ON S.ProductSubcategoryKey = P.ProductSubcategoryKey
```

```
WHERE C.EnglishProductCategoryName = @NAME
END
```

3. WRITE A STORED PROCEDURE TO FIND THE CUSTOMER KEY AND CUSTOMER NAME FOR THE PRODUCT.
   The user will enter the productname and it will return the list of customers for that product.

4. Write a procedure to find the sales amount of each category.

5. Write a procedure to enter a sales entry into FactInternetSales.

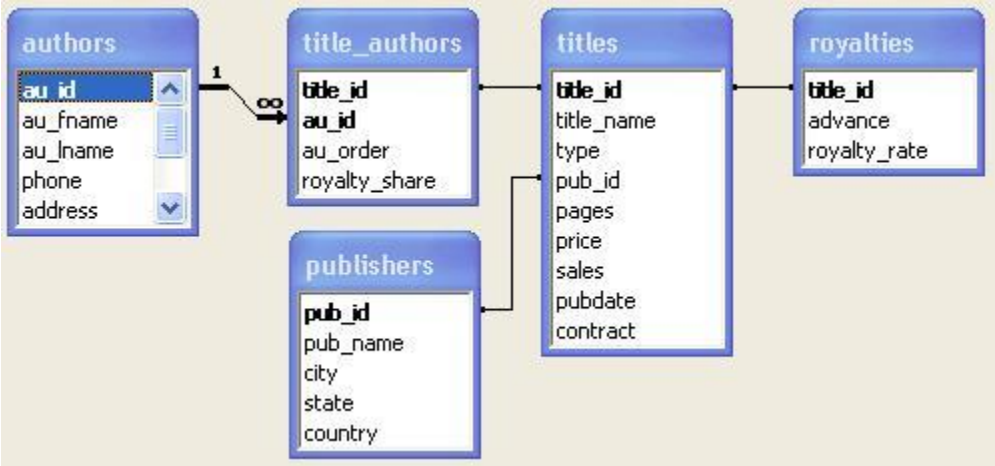
6. Write a procedure to find the sales amount for each sales territory.

7. Write a procedure to find the sales details for given date range. User will enter the start date and end date and based upon that the sales for the period should be displayed.

8. Write a procedure to get total sales amount for a category or sub category or product.
   If user enters a category name, it should give the totalsales for that category.
   If user enters subcategoryname, it should give the totalsales for that sub-category.
   If user enter product name, it should give the total sales for that product.
9. Write a procedure to find the number of males and females in employee table.
10. Write a procedure to find the number of employees in each territory.

SQL SERVER TRIGGER DATABASE

Create a SQL Server Trigger



Motivation

Publishers often have a multi-tiered royalty system. For *example* for the first 1000 books sold the author might get a certain royalty rate. For every book after the 1000th, the royalty rate might double. For example for one book, the author might get 9% for the first 1000 books sold and 18% for all subsequent sales. For another book the author might get 10% for the first 1000 books and 20% for all subsequent sales.

The Assignment

Write a trigger that doubles the royalty rate for a book when the sales for that book surpasses 1000. Specifically, when the "sales" column in the "titles" table for a particular row is updated from a value that was <= to 1000 to a value that is > 1000 the trigger should double the value of the "royalty_rate" column in the "royalties" table for the appropriate row .