Create Database LivreExercice

create table TYPE\_EXERCICE

(type\_exercice varchar(2) primary key not null,

libelle\_type varchar(25) not null)

create table EXERCICE

(numero\_exercice varchar(30) not null,

enonce varchar(30),

nombre\_page int,

type\_exercice varchar(2) not null,

primary key(numero\_exercice, type\_exercice),

foreign key (type\_exercice) references TYPE\_EXERCICE(type\_exercice))

create table ESTIMATION

(type\_exercice varchar(2) not null,

niveau varchar(30),

duree\_estime datetime,

primary key(type\_exercice , niveau),

foreign key (type\_exercice) references TYPE\_EXERCICE(type\_exercice))

drop table TYPE\_EXERCICE

drop table EXERCICE

drop table ESTIMATION

insert into TYPE\_EXERCICE Values

('M','Manipulation de BD')

insert into TYPE\_EXERCICE Values

('C','Conception de BD')

insert into TYPE\_EXERCICE Values

('P','Problem de BD')

insert into ESTIMATION values

('M','Moyen','03:15:00')

insert into ESTIMATION values

('M','Facile','02:30:00')

insert into ESTIMATION values

('C','Defficile','05:00:00')

insert into ESTIMATION values

('P','Facile','04:20:00')

insert into ESTIMATION values

('M','Defficile','04:30:00')

insert into ESTIMATION values

('P','Moyen','05:10:00')

insert into ESTIMATION values

('C','Facile','02:45:00')

insert into ESTIMATION values

('C','Moyen','04:00:00')

insert into ESTIMATION values

('P','Defficile','06:25:00')

insert into Exercice values

('EX001','Enonce EX001','3','C')

insert into Exercice values

('EX002','Enonce EX002','8','M')

insert into Exercice values

('EX003','Enonce EX003','6','P')

insert into Exercice values

('EX004','Enonce EX004','2','P')

insert into Exercice values

('EX005','Enonce EX005','5','C')

insert into Exercice values

('EX006','Enonce EX006','4','C')

-- 1

SELECT \* FROM EXERCICE WHERE nombre\_page < 4

-- 2

SELECT \* FROM ESTIMATION WHERE DATEPART(HOUR, duree\_estime) < 3

-- 3

SELECT TE.libelle\_type, ES.type\_exercice, MAX(ES.duree\_estime)

FROM TYPE\_EXERCICE AS TE

JOIN ESTIMATION AS ES ON TE.type\_exercice = ES.type\_exercice

WHERE ES.duree\_estime = (SELECT MAX(duree\_estime) FROM ESTIMATION)

GROUP BY TE.libelle\_type, ES.type\_exercice;

-- 4

SELECT

numero\_exercice,

enonce,

nombre\_page,

type\_exercice,

CASE

WHEN nombre\_page < 2 THEN 'small'

WHEN nombre\_page < 4 THEN 'Medium'

ELSE 'large'

END

FROM

EXERCICE

-- 5

DECLARE @TotalExercices INT;

DECLARE @i INT;

SET @TotalExercises = 0;

SET @i = 1;

WHILE @i <= (SELECT COUNT(\*) FROM EXERCICE)

BEGIN

SET @TotalExercices = @TotalExercices + 1;

SET @i = @i + 1;

END;

SELECT @TotalExercises AS TotalExercises;

--6

CREATE FUNCTION GetTotalPagesAndCount(

@TypeExercice varchar(2)

)

RETURNS TABLE

AS

RETURN

(

SELECT

SUM(nombre\_page) AS TotalPages,

COUNT(\*) AS ExerciseCount

FROM

EXERCICE

WHERE

type\_exercice = @TypeExercice

)

SELECT TotalPages, ExerciseCount

FROM dbo.GetTotalPagesAndCount('C');

-- 7

CREATE PROCEDURE GetExercisesByType

@TypeExercice varchar(2),

@ExerciseCount int OUTPUT

AS

BEGIN

SET NOCOUNT ON

SELECT \*

FROM EXERCICE

WHERE type\_exercice = @TypeExercice

SET @ExerciseCount = @@ROWCOUNT

END

DECLARE @Count int

EXEC GetExercisesByType 'P', @ExerciseCount = @Count OUTPUT;

SELECT @Count AS ExerciseCount

-- 8

CREATE TRIGGER trg\_estimation\_delete

ON ESTIMATION

FOR DELETE

AS

BEGIN

INSERT INTO DELETED\_ESTIMATION (type\_exercice, niveau, duree\_estime)

SELECT type\_exercice, niveau, duree\_estime

FROM DELETED

END

CREATE TABLE DELETED\_ESTIMATION (

type\_exercice VARCHAR(2) NOT NULL,

niveau VARCHAR(30),

duree\_estime DATETIME,

-- Additional columns if needed

)

DELETE ESTIMATION WHERE type\_exercice = 'P'

SELECT \* FROM ESTIMATION

SELECT \* FROM DELETED\_ESTIMATION

SELECT \* FROM TYPE\_EXERCICE

SELECT \* FROM EXERCICE