Bibloteka e Fakultetit

KERKESAT FUNKSIONALE DHE

JO-FUNKSIONALE

**Funksionale:**

1. Moduli Literatura

1.a) Sistemi duhet te mundesoje Shtimin e literatures

1.b) Sistemi duhet te mundesoje Fshirjen e literatures

1.c) Sistemi duhet te mundesoje Listimin e literatures

1.d) Sistemi duhet te mundesoje shfaqjen e Disponueshmeris se literatures

1.e)Sistemi duhet te mundesoje shfaqjen e informatave baze te nje libri

2) Moduli Studenti

2.a)Sistemi duhet t’I mundesoje studentit kerkimin e literatures permes Emrit, emrit te autorit dhe permes shtepise botuese

2.b)Sistemi duhet te I mendesoje studentit qe te shoh se a eshte I disponueshem literatura

2.b.I)Nese literatura nuk eshte e disponueshme sistemi duhet t’i tregoj studentit daten se kur do te kthehet literatura e kerkuar

2.b.II)Nese literatura eshte e disponueshme ateher sistemi duhet t’i mundesoje studentit huazimin e literatures

3)Moduli I menaxhimit te Autorve

3.a)Sistemi duhet te mundesoje shfaqjen e te dhenave bazike te autorit

3.b) Sistemi duhet te mundesoje kerkimin e literatures ne baze te Autorit

4)Moduli Menaxhimi I huazimeve

4.a)Sistemi duhet te mundesoje shfaqjen e Studentit I cili e posedon literaturen

4.b)Sistemi duhet te mundesoje shfaqjen e Disponueshmerise

4.c) Sistemi duhet te mundesoje shfaqjen e Dates se huazimit

4.d) Sistemi duhet te mundesoje shfaqjen e Dates se kthimit

5)Moduli i Shtepse Botuese

5.1)Sistemi duhet te mundesoje kerkimin ne baze te Shtepise Botuese.

**Jo-Funksionale:**

1)Fonti te jete ne formatin Calibri

2) Madhesia e fontit te jete 15px

3)Mundesine e sortimit ne menyre rritese te librave ne baze te emrit dhe dates se publikimit

4)Te perdoret MSSQL Server

5)Duhet ta menaxhoj shumegjuhesine

Entitetet

LIBRAT, AUTORËT, BOTUESIT, LEXUESIT, **HUAZIMET**.

**Relacioni Lexuesit:**

R (LexuesiID, Emri, Mbiemri, Gjinia, DataLindjes, Drejtimi)

**Relacioni Shtepia Botuese:**

R (ShBotuesID, Emri, DataThemelimit)

**Relacioni Librat:**

R (LibriID, Emri , ShBotueseID , DataBotimit, Disponueshmeria, ISBN)

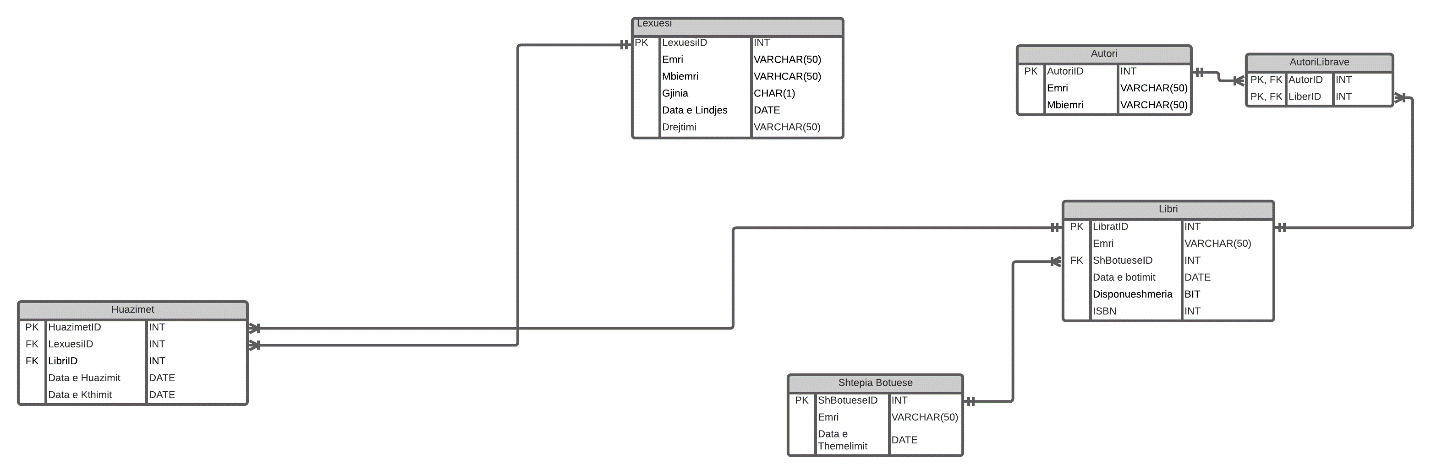
**Relacioni Autori:**

R(AutoriID,Emri,Mbiemri)

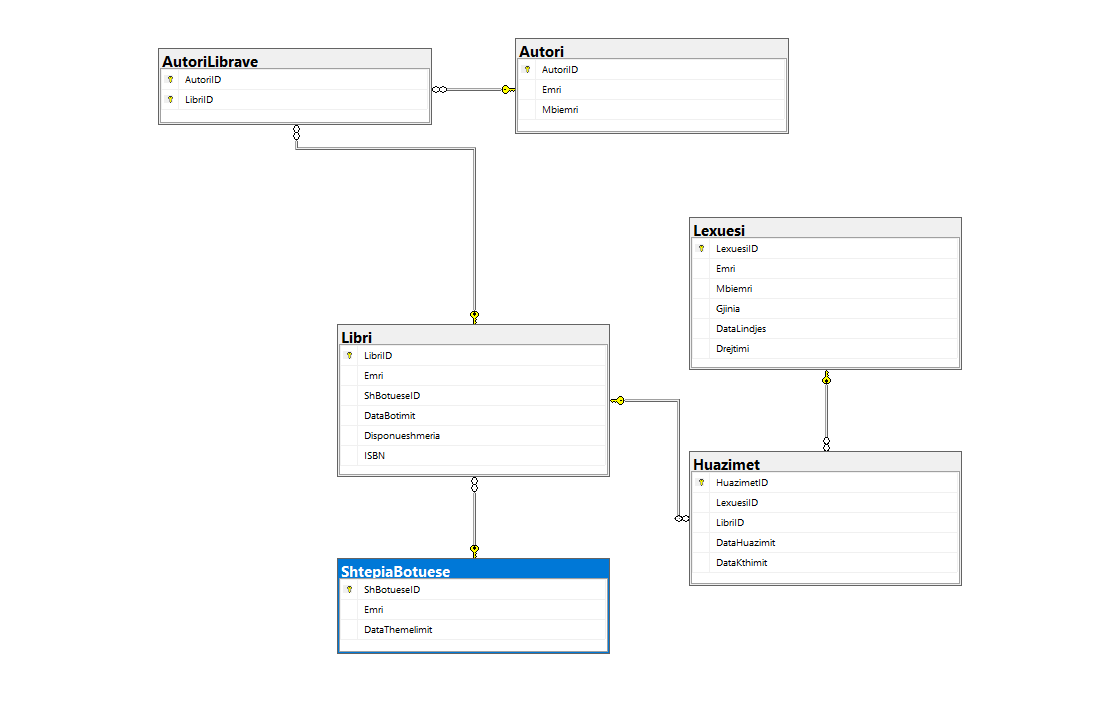
**Relacioni Huazimet:**

R(HuazimetID,LexuesitID,LibratID ,DataEHuazimit,DataKthimit)

E-R Diagram



Diagrami i Relacioneve



Query Per krijimin e tabelave

--CREATE TABLE Lexuesi (

-- LexuesiID INT PRIMARY KEY IDENTITY(1,1) NOT NULL,

-- Emri VARCHAR(50) NOT NULL,

-- Mbiemri VARCHAR(50) NOT NULL,

-- Gjinia CHAR(1) NOT NULL,

-- DataLindjes DATE NOT NULL,

-- Drejtimi VARCHAR(50) NOT NULL,

--)

--CREATE TABLE Autori (

-- AutoriID INT PRIMARY KEY IDENTITY(1,1) NOT NULL,

-- Emri VARCHAR(50) NOT NULL,

-- Mbiemri VARCHAR(50) NOT NULL

--)

--CREATE TABLE ShtepiaBotuese(

-- ShBotueseID INT PRIMARY KEY IDENTITY(1,1) NOT NULL,

-- Emri VARCHAR(50) NOT NULL,

-- DataThemelimit DATE NOT NULL

--)

--CREATE TABLE Libri (

-- LibriID INT PRIMARY KEY IDENTITY(1,1) NOT NULL,

-- Emri VARCHAR(50) NOT NULL,

-- ShBotueseID INT NOT NULL,

-- DataBotimit DATE NOT NULL,

-- Disponueshmeria BIT NOT NULL,

-- ISBN INT NOT NULL,

-- CONSTRAINT FK\_ShtepiaBotuese FOREIGN KEY (ShBotueseID) REFERENCES ShtepiaBotuese(ShBotueseID)

--)

--CREATE TABLE Huazimet(

-- HuazimetID INT PRIMARY KEY IDENTITY(1,1) NOT NULL,

-- LexuesiID INT NOT NULL,

-- LibriID INT NOT NULL,

-- DataHuazimit DATE NOT NULL,

-- DataKthimit DATE NOT NULL,

-- CONSTRAINT FK\_Lexuesi FOREIGN KEY (LexuesiID) REFERENCES Lexuesi(LexuesiID),

-- CONSTRAINT FK\_Libri FOREIGN KEY (LibriID) REFERENCES Libri(LibriID)

--)

--CREATE TABLE AutoriLibrave(

-- AutoriID INT NOT NULL,

-- LibriID INT NOT NULL,

-- CONSTRAINT FK\_Autori FOREIGN KEY (AutoriID) REFERENCES Autori(AutoriID),

-- CONSTRAINT FK\_LibriShkruar FOREIGN KEY (LibriID) REFERENCES Libri(LibriID),

-- CONSTRAINT PK\_AutoriLibrave PRIMARY KEY (AutoriID, LibriID)

--)

FUTJA E VLERAVE NEPER TABELA

--SELECT \* FROM Lexuesi

--INSERT INTO Lexuesi(Emri,Mbiemri,Gjinia,DataLindjes,Drejtimi)

--VALUES ('Bekim','Peci','M', '2001-08-03', 'SHK')

--INSERT INTO Lexuesi(Emri,Mbiemri,Gjinia,DataLindjes,Drejtimi)

--VALUES ('Syle','Sylani','M','1999-12-11','EK')

--INSERT INTO Lexuesi(Emri,Mbiemri,Gjinia,DataLindjes,Drejtimi)

--VALUES ('Betim','Betimi','M','1988-06-01','SHK')

--SELECT \* FROM Lexuesi

--SELECT \* FROM ShtepiaBotuese

--INSERT INTO ShtepiaBotuese(Emri,DataThemelimit)

--VALUES ('Lorea','2014-03-04')

--INSERT INTO ShtepiaBotuese(Emri,DataThemelimit)

--VALUES ('Dukagjini','1994-03-14')

--INSERT INTO ShtepiaBotuese(Emri,DataThemelimit)

--VALUES('Lora','2000-04-15')

--SELECT \* FROM ShtepiaBotuese

--SELECT \* FROM Libri

--INSERT INTO Libri(Emri,ShBotueseID,DataBotimit,Disponueshmeria,ISBN)

--VALUES ('Rruget e heshtura',1,'2015-08-04',1,123456789)

--INSERT INTO Libri(Emri,ShBotueseID,DataBotimit,Disponueshmeria,ISBN)

--VALUES ('Cicerima e nates',2,'2003-11-30',1,118456721)

--INSERT INTO Libri(Emri,ShBotueseID,DataBotimit,Disponueshmeria,ISBN)

--VALUES ('Ushtari i Vetrmuar',2,'2001-03-31',1,1115623465)

--INSERT INTO Libri(Emri,ShBotueseID,DataBotimit,Disponueshmeria,ISBN)

--VALUES('Shkretetira',3,'2001-09-03',1,134567892)

--SELECT \* FROM Autori

--INSERT INTO Autori(Emri,Mbiemri)

--VALUES('Filan','Fisteku')

--INSERT INTO Autori (Emri,Mbiemri)

--VALUES('Syle','Sylani')

--INSERT INTO Autori (Emri,Mbiemri)

--VALUES('Askush','Askushi')

--SELECT \* FROM Autori

--SELECT \* FROM Lexuesi

--SELECT \* FROM Libri

--INSERT INTO Huazimet(LexuesiID,LibriID,DataHuazimit,DataKthimit)

--VALUES (1,1,'2021-01-09','2021-01-23')

--INSERT INTO Huazimet(LexuesiID,LibriID,DataHuazimit,DataKthimit)

--VALUES (5,3,'2021-01-02','2021-01-16')

--INSERT INTO Huazimet(LexuesiID,LibriID,DataHuazimit,DataKthimit)

--VALUES(6,4,'2020-12-31','2021-01-14')

--SELECT \* FROM Huazimet

--SELECT \* FROM Autori

--SELECT \* FROM Libri

--INSERT INTO AutoriLibrave

--VALUES(1,1)

--INSERT INTO AutoriLibrave

--VALUES(1,4)

--INSERT INTO AutoriLibrave

--VALUES(2,1)

--INSERT INTO AutoriLibrave(AutoriID,LibriID)

--VALUES(3,5)

SELECT \* FROM AutoriLibrave

FUNKSIONET

--CREATE FUNCTION LlogaritMoshen (@DataLindjes DATE)

--RETURNS INT

--AS

--BEGIN

--DECLARE @Mosha INT

--SET @Mosha = DATEDIFF(YEAR,@DataLindjes,GETDATE()) -

-- CASE

-- WHEN (MONTH(@DataLindjes) > MONTH(GETDATE())) OR

-- (MONTH(@DataLindjes) = MONTH(GETDATE())) AND DAY(@DataLindjes) > DAY(GETDATE())

-- THEN 1

-- ELSE 0

-- END

-- RETURN @Mosha

--END

--SELECT \* FROM Lexuesi

--DECLARE @DateBirth DATE

--SELECT

-- @DateBirth = DataLindjes

-- FROM Lexuesi

-- WHERE LexuesiID = 5

--SELECT dbo.LlogaritMoshen(@DateBirth) AS Mosha

--CREATE FUNCTION NrLibrave(@AutoriID INT)

-- RETURNS INT

-- AS

-- BEGIN

-- DECLARE @NrLibrave INT

-- SET @NrLibrave = (SELECT COUNT(AutoriID) FROM AutoriLibrave WHERE AutoriID = @AutoriID)

-- return @NrLibrave

-- END

--SELECT \* FROM Autori

--SELECT \* FROM AutoriLibrave

--SELECT dbo.NrLibrave(3) AS NrLibrave

--CREATE FUNCTION DataKthimit(@DataHuazimit DATE)

-- RETURNS DATE

-- AS

-- BEGIN

-- DECLARE @DataKthimit DATE

-- SET @DataKthimit = DATEADD(WEEK,2,@DataHuazimit)

-- return @DataKthimit

-- END

--SELECT dbo.DataKthimit('2021-01-01') AS DataKthimit

--CREATE FUNCTION NdryshoDisponeshmerine(@LibriID INT)

-- RETURNS BIT

-- AS

-- BEGIN

-- DECLARE @Perfundimi VARCHAR(50)

-- DECLARE @Disponueshmeria BIT

-- SELECT

-- @Disponueshmeria = Disponueshmeria

-- FROM Libri

-- WHERE LibriID = @LibriID

-- IF @Disponueshmeria = 0

-- SET @Disponueshmeria = 1

-- ELSE

-- SET @Disponueshmeria = 0

-- RETURN @Disponueshmeria

-- END

--DECLARE @LibriID INT = 1

--UPDATE Libri

-- SET Disponueshmeria = dbo.NdryshoDisponeshmerine(@LibriID)

-- WHERE LibriID = @LibriID

--SELECT \* FROM Libri

--CREATE FUNCTION DiteParaPervjetorit(@DataThemelimit DATE)

-- RETURNS INT

-- AS

-- BEGIN

-- DECLARE @Differenca INT

-- SET @Differenca = DATEDIFF(DAY,@DataThemelimit,GETDATE())

-- WHILE (@Differenca >= 365)

-- BEGIN

-- SET @Differenca = @Differenca - 365

-- END

-- RETURN @Differenca

-- END

--DECLARE @DataThemelimit DATE

--SELECT

-- @DataThemelimit = DataThemelimit

-- FROM ShtepiaBotuese

-- WHERE ShBotueseID = 1

--SELECT dbo.DiteParaPervjetorit(@DataThemelimit) AS DiteParaPervjetorit

--SELECT \* FROM ShtepiaBotuese

VIEWS

--CREATE VIEW vStudentList

--AS

--SELECT Emri, Mbiemri, Drejtimi

--FROM Lexuesi

--SELECT \*

--FROM vStudentList

--CREATE VIEW vHuazimetList

--AS

--SELECT HuazimetID, Lexuesi.Emri, Lexuesi.Mbiemri, Libri.Emri AS 'Libri', DataHuazimit,DataKthimit

--FROM Huazimet, Lexuesi, Libri

--WHERE Huazimet.LexuesiID = Lexuesi.LexuesiID AND Huazimet.LibriID = Libri.LibriID

--SELECT \* FROM vHuazimetList

--CREATE VIEW vAutoriLibrave

--AS

--SELECT Autori.Emri, Autori.Mbiemri,Libri.Emri AS Libri,Libri.DataBotimit

--FROM AutoriLibrave

--JOIN Autori ON Autori.AutoriID = AutoriLibrave.AutoriID

--JOIN Libri ON Libri.LibriID = AutoriLibrave.LibriID

--SELECT \* FROM vAutoriLibrave

--CREATE VIEW vAutori

--AS

--SELECT Autori.Emri, Mbiemri, Libri.Emri AS 'Libri',dbo.NrLibrave(Autori.AutoriID) AS 'Numri i Librave'

--FROM Autori

--JOIN AutoriLibrave ON AutoriLibrave.AutoriID = Autori.AutoriID

--JOIN Libri ON Libri.LibriID = AutoriLibrave.LibriID

--SELECT \* FROM vAutori

--SELECT \* FROM Libri

--CREATE VIEW vLibri

--AS

--SELECT Libri.Emri, DataBotimit, ISBN, ShtepiaBotuese.Emri AS ShtepiaBotuese

--FROM Libri

--JOIN ShtepiaBotuese ON Libri.ShBotueseID = ShtepiaBotuese.ShBotueseID

--SELECT \* FROM vLibri

--CREATE VIEW vShtepiaBotuese

--AS

--SELECT ShtepiaBotuese.Emri, DataThemelimit, Libri.Emri AS 'Emri i librit'

--FROM ShtepiaBotuese

--JOIN Libri ON ShtepiaBotuese.ShBotueseID = Libri.ShBotueseID

--SELECT \* FROM vShtepiaBotuese

Transaksioni

--CREATE PROCEDURE LargoHuazimin

-- @LibriID INT

-- AS

-- BEGIN

-- BEGIN TRY

-- BEGIN TRANSACTION

-- DELETE FROM Huazimet

-- WHERE Huazimet.LibriID = @LibriID

-- UPDATE Libri

-- SET Disponueshmeria = 1

-- WHERE LibriID = @LibriID

-- COMMIT TRANSACTION

-- END TRY

-- BEGIN CATCH

-- ROLLBACK TRANSACTION

-- END CATCH

-- END

--SELECT \* FROM Lexuesi

--CREATE PROCEDURE LargoHuazimin

-- @LibriID INT

-- AS

-- BEGIN

-- BEGIN TRY

-- BEGIN TRANSACTION

-- DELETE FROM Huazimet

-- WHERE Huazimet.LibriID = @LibriID

-- UPDATE Libri

-- SET Disponueshmeria = 1

-- WHERE LibriID = @LibriID

-- COMMIT TRANSACTION

-- END TRY

-- BEGIN CATCH

-- ROLLBACK TRANSACTION

-- END CATCH

-- END

--SELECT \* FROM Lexuesi

--CREATE PROCEDURE ShtoHuazim

-- @LibriID INT,

-- @LexuesiID INT,

-- @DataHuazimit DATE

-- AS

-- BEGIN

-- DECLARE @Disponueshmeria BIT

-- BEGIN TRY

-- BEGIN TRANSACTION

-- SELECT

-- @Disponueshmeria = Disponueshmeria

-- FROM Libri

-- WHERE LibriID = @LibriID

-- IF @Disponueshmeria > 0

-- BEGIN

-- INSERT INTO Huazimet(LexuesiID,LibriID,DataHuazimit,DataKthimit)

-- VALUES (@LexuesiID,@LibriID,@DataHuazimit,dbo.DataKthimit(@DataHuazimit))

-- UPDATE Libri

-- SET

-- Disponueshmeria = 0

-- WHERE LibriID = @LibriID

-- COMMIT TRANSACTION

-- END

-- ELSE

-- ROLLBACK TRANSACTION

-- END TRY

-- BEGIN CATCH

-- ROLLBACK TRANSACTION

-- END CATCH

-- END

--SELECT \* FROM Libri

--SELECT \* FROM Lexuesi

----SELECT \* FROM Huazimet

--EXEC ShtoHuazim @LibriID = 1, @LexuesiID = 1, @DataHuazimit = '2021-01-05';

--SELECT \* FROM Huazimet

--EXEC LargoHuazimin @LibriID = 4;

--EXEC ShtoHuazim 3,6,'2021-01-10';

--SELECT \* FROM Huazimet

--SELECT \* FROM Libri