

מסד נתונים מטלה 4 :

```
IF OBJECT_ID('dbo.Roles', 'U') IS NOT NULL DROP TABLE dbo.Roles;
IF OBJECT_ID('dbo.UserRoles', 'U') IS NOT NULL DROP TABLE dbo.UserRoles;
IF OBJECT_ID('dbo.PaymentApprovals', 'U') IS NOT NULL DROP TABLE
dbo.PaymentApprovals;
IF OBJECT_ID('dbo.PaymentRequests', 'U') IS NOT NULL DROP TABLE
dbo.PaymentRequests;
IF OBJECT_ID('dbo.Files', 'U') IS NOT NULL DROP TABLE dbo.Files;
IF OBJECT_ID('dbo.BudgetCategories', 'U') IS NOT NULL DROP TABLE
dbo.BudgetCategories;
IF OBJECT_ID('dbo.ResearchMembers', 'U') IS NOT NULL DROP TABLE
dbo.ResearchMembers;
IF OBJECT_ID('dbo.Researches', 'U') IS NOT NULL DROP TABLE dbo.Researches;
IF OBJECT_ID('dbo.ResearchCenters', 'U') IS NOT NULL DROP TABLE
dbo.ResearchCenters;
IF OBJECT_ID('dbo.Users', 'U') IS NOT NULL DROP TABLE dbo.Users;
GO

CREATE TABLE dbo.Roles (
    RoleId INT IDENTITY(1,1) PRIMARY KEY,
    RoleName NVARCHAR(50) NOT NULL UNIQUE -- 'Researcher', 'CenterManager',
'Secretary'
);
GO

-----
-- Users - (ת"ז היא המזהה) משתמשים
-----
CREATE TABLE dbo.Users (
    IdNumber CHAR(9) NOT NULL PRIMARY KEY, -- תעודת זהות
    UserName VARCHAR(50) NOT NULL UNIQUE, -- באנגלית בלבד
    Email NVARCHAR(255) NOT NULL UNIQUE,
    PasswordHash NVARCHAR(255) NOT NULL,
    FirstName NVARCHAR(50) NOT NULL,
    LastName NVARCHAR(50) NOT NULL,
    IsActive BIT NOT NULL CONSTRAINT DF_Users_IsActive DEFAULT
(1),
    CreatedAt DATETIME2 NOT NULL CONSTRAINT DF_Users_CreatedAt DEFAULT
(SYSUTCDATETIME()),
    CONSTRAINT CK_Users_UserName_English
CHECK (UserName NOT LIKE '%[^A-Za-z0-9-._]%' ),
    CONSTRAINT CK_Users_Email
CHECK (Email LIKE '%_@%._%')
);
GO

-----
-- UserRoles - שיוך משתמשים לתפקידים
-----
CREATE TABLE dbo.UserRoles (
    IdNumber CHAR(9) NOT NULL,
    RoleId INT NOT NULL,
    CONSTRAINT PK_UserRoles PRIMARY KEY (IdNumber, RoleId),
    CONSTRAINT FK_UserRoles_Users FOREIGN KEY (IdNumber) REFERENCES
dbo.Users(IdNumber),
    CONSTRAINT FK_UserRoles_Roles FOREIGN KEY (RoleId) REFERENCES
dbo.Roles(RoleId)
);
GO

-----
```

-- ResearchCenters - מרכזי מחקר

```
-----  
CREATE TABLE dbo.ResearchCenters (  
    CenterId INT IDENTITY(1,1) PRIMARY KEY,  
    CenterName NVARCHAR(100) NOT NULL,  
    ManagerId CHAR(9) NOT NULL, -- ת"ז של מנהל מרכז המחקר  
    CONSTRAINT FK_ResearchCenters_Manager  
        FOREIGN KEY (ManagerId) REFERENCES dbo.Users(IdNumber)  
);  
GO
```

-- Researches - מחקרים

```
-----  
CREATE TABLE dbo.Researches (  
    ResearchId INT IDENTITY(1,1) PRIMARY KEY,  
    ResearchCode NVARCHAR(50) NOT NULL UNIQUE, -- קוד מחקר של רופין  
    ResearchName NVARCHAR(200) NOT NULL,  
    Description NVARCHAR(MAX) NULL,  
    TotalBudget DECIMAL(18,2) NOT NULL,  
    StartDate DATE NOT NULL,  
    EndDate DATE NULL,  
    IsUnderCenter BIT NOT NULL CONSTRAINT  
DF_Researches_IsUnderCenter DEFAULT (0),  
    CenterId INT NULL,  
    PrimaryResearcherId CHAR(9) NOT NULL, -- חוקר ראשי  
    CreatedAt DATETIME2 NOT NULL CONSTRAINT  
DF_Researches_CreatedAt DEFAULT (SYSUTCDATETIME()),  
    CONSTRAINT FK_Researches_Center  
        FOREIGN KEY (CenterId) REFERENCES dbo.ResearchCenters(CenterId),  
    CONSTRAINT FK_Researches_PrimaryResearcher  
        FOREIGN KEY (PrimaryResearcherId) REFERENCES dbo.Users(IdNumber),  
    CONSTRAINT CHK_Researches_Center  
        CHECK (IsUnderCenter = 0 OR CenterId IS NOT NULL)  
);  
GO
```

-- ResearchMembers - שיוך חוקרים למחקרים

```
-----  
CREATE TABLE dbo.ResearchMembers (  
    ResearchId INT NOT NULL,  
    IdNumber CHAR(9) NOT NULL,  
    ResearchRole NVARCHAR(50) NOT NULL, -- PI, Researcher וכו'  
    IsActive BIT NOT NULL CONSTRAINT DF_ResearchMembers_IsActive  
DEFAULT (1),  
    CONSTRAINT PK_ResearchMembers PRIMARY KEY (ResearchId, IdNumber),  
    CONSTRAINT FK_ResearchMembers_Research FOREIGN KEY (ResearchId) REFERENCES  
dbo.Researches(ResearchId),  
    CONSTRAINT FK_ResearchMembers_User FOREIGN KEY (IdNumber) REFERENCES  
dbo.Users(IdNumber)  
);  
GO
```

-- BudgetCategories - קטגוריות תקציב

```
-----  
CREATE TABLE dbo.BudgetCategories (  
    CategoryId INT IDENTITY(1,1) PRIMARY KEY,  
    CategoryName NVARCHAR(100) NOT NULL UNIQUE,  
    Description NVARCHAR(255) NULL  
);
```

GO

```
-----
-- Files - (משויכים למחקר) / מסמכים / קבצים
-----
CREATE TABLE dbo.Files (
    FileId          INT IDENTITY(1,1) PRIMARY KEY,
    ResearchId      INT              NOT NULL,
    OriginalFileName NVARCHAR(255) NOT NULL,
    StoredFileName  NVARCHAR(255) NOT NULL,
    RelativePath    NVARCHAR(500) NOT NULL,
    ContentType     NVARCHAR(100) NULL,
    UploadedById    CHAR(9)         NOT NULL,
    UploadedAt      DATETIME2       NOT NULL CONSTRAINT DF_Files_UploadedAt
    DEFAULT (SYSUTCDATETIME()),
    CONSTRAINT FK_Files_Research FOREIGN KEY (ResearchId) REFERENCES
    dbo.Researches(ResearchId),
    CONSTRAINT FK_Files_UploadedBy FOREIGN KEY (UploadedById) REFERENCES
    dbo.Users(IdNumber)
);
GO
```

```
-----
-- PaymentRequests - בקשות תשלום / הוצאות
-----
CREATE TABLE dbo.PaymentRequests (
    PaymentRequestId INT IDENTITY(1,1) PRIMARY KEY,
    ResearchId       INT              NOT NULL,
    RequestedById    CHAR(9)         NOT NULL,
    CategoryId       INT              NOT NULL,
    Amount           DECIMAL(18,2) NOT NULL,
    Currency         NVARCHAR(10) NOT NULL CONSTRAINT
    DF_PaymentRequests_Currency DEFAULT (N'ILS'),
    RequestDate      DATETIME2       NOT NULL CONSTRAINT
    DF_PaymentRequests_RequestDate DEFAULT (SYSUTCDATETIME()),
    Description      NVARCHAR(MAX) NULL,
    FileId           INT              NULL,
    Status           NVARCHAR(20) NOT NULL CONSTRAINT
    DF_PaymentRequests_Status DEFAULT (N'Pending'),
    -- 'Pending', 'Approved', 'Rejected', 'Cancelled'
    CONSTRAINT FK_PaymentRequests_Research FOREIGN KEY (ResearchId) REFERENCES
    dbo.Researches(ResearchId),
    CONSTRAINT FK_PaymentRequests_RequestedBy FOREIGN KEY (RequestedById)
    REFERENCES dbo.Users(IdNumber),
    CONSTRAINT FK_PaymentRequests_Category FOREIGN KEY (CategoryId) REFERENCES
    dbo.BudgetCategories(CategoryId),
    CONSTRAINT FK_PaymentRequests_File FOREIGN KEY (FileId) REFERENCES
    dbo.Files(FileId),
    CONSTRAINT CK_PaymentRequests_Status
    CHECK (Status IN (N'Pending', N'Approved', N'Rejected', N'Cancelled'))
);
GO
```

```
-----
-- PaymentApprovals - אישורי תשלום
-- למחקר תחת מרכז: CenterManager -> Secretary
-- בלבד למחקר פרטי: Secretary
-----
```

```

CREATE TABLE dbo.PaymentApprovals (
    ApprovalId          INT IDENTITY(1,1) PRIMARY KEY,
    PaymentRequestId    INT          NOT NULL,
    ApprovedById        CHAR(9)      NOT NULL,          -- מי אישר
    ApprovalRole        NVARCHAR(50) NOT NULL,          -- 'CenterManager' או
'Secretary'
    Status              NVARCHAR(20) NOT NULL,          -- 'Approved' או 'Rejected'
    ApprovalDate        DATETIME2    NOT NULL CONSTRAINT
DF_PaymentApprovals_ApprovalDate DEFAULT (SYSUTCDATETIME()),
    Comment             NVARCHAR(MAX) NULL,
    CONSTRAINT FK_PaymentApprovals_PaymentRequest FOREIGN KEY
(PaymentRequestId) REFERENCES dbo.PaymentRequests(PaymentRequestId),
    CONSTRAINT FK_PaymentApprovals_ApprovedBy FOREIGN KEY (ApprovedById)
REFERENCES dbo.Users(IdNumber),
    CONSTRAINT CK_PaymentApprovals_ApprovalRole
CHECK (ApprovalRole IN (N'CenterManager', N'Secretary')),
    CONSTRAINT CK_PaymentApprovals_Status
CHECK (Status IN (N'Approved', N'Rejected'))
);
GO

```

SP:

```
IF OBJECT_ID('dbo.spRegisterUser', 'P') IS NOT NULL
```

```
    DROP PROCEDURE dbo.spRegisterUser;
```

GO

```
CREATE PROCEDURE dbo.spRegisterUser
```

```
@ IdNumber  CHAR,(9)
```

```
@ UserName  VARCHAR,(50)
```

```
@ Email     NVARCHAR,(255)
```

```
@ PasswordHash NVARCHAR,(255)
```

```
@ FirstName NVARCHAR,(50)
```

```
@ LastName  NVARCHAR(50)
```

AS

```
BEGIN
```

```
    SET NOCOUNT ON;
```

```
    IF EXISTS (SELECT 1 FROM dbo.Users WHERE IdNumber = @IdNumber)
```

```
    BEGIN
```

```
RAISERROR) N;(1,16, 'משתמש עם תעודת זהות זו כבר קיים במערכת.')
```

```
RETURN;
```

```
END
```

```
IF EXISTS (SELECT 1 FROM dbo.Users WHERE UserName = @UserName)
```

```
BEGIN
```

```
RAISERROR) N;(1,16, 'שם המשתמש כבר תפוס. בחר שם אחר.')
```

```
RETURN;
```

```
END
```

```
IF EXISTS (SELECT 1 FROM dbo.Users WHERE Email = @Email)
```

```
BEGIN
```

```
RAISERROR) N;(1,16, 'כתובת המייל כבר רשומה במערכת.')
```

```
RETURN;
```

```
END
```

```
INSERT INTO dbo.Users
```

```
)      IdNumber, UserName, Email, PasswordHash, FirstName, LastName, IsActive,
```

```
CreatedAt(
```

```
VALUES
```

```
@)      IdNumber, @UserName, @Email, @PasswordHash, @FirstName, @LastName,
```

```
1, SYSUTCDATETIME;()
```

```
SELECT
```

```
IdNumber,
```

```
UserName,
```

```
Email,
```

```
FirstName,
```

```
LastName,
```

```
IsActive,
```

```
CreatedAt
```

```
FROM dbo.Users
```

```
WHERE IdNumber = @IdNumber;

END

GO
```

```
-----

--SP 2 – Login

-----
```

```
IF OBJECT_ID('dbo.spLoginUser', 'P') IS NOT NULL

    DROP PROCEDURE dbo.spLoginUser;

GO
```

```
CREATE PROCEDURE dbo.spLoginUser
```

```
@   UserName   VARCHAR(50)
```

```
@   PasswordHash NVARCHAR(255)
```

```
AS
```

```
BEGIN
```

```
    SET NOCOUNT ON;
```

```
    IF NOT EXISTS)
```

```
        SELECT 1
```

```
        FROM dbo.Users
```

```
        WHERE UserName = @UserName
```

```
        AND PasswordHash = @PasswordHash
```

```
        AND IsActive = 1
```

```
(
```

```
    BEGIN
```

```
        RAISERROR) N;(1 ,16 ,',' פעיל. ' או שהמשתמש אינו פעיל. ' ,16 ,1) N;
```

```
        RETURN;
```

```
    END
```

```
SELECT
```

```

        IdNumber,
        UserName,
        Email,
        FirstName,
        LastName,
        IsActive,
        CreatedAt
    FROM dbo.Users
    WHERE UserName = @UserName;
END
GO

-----
--SP 3 – GetUserWithRoles
-----

IF OBJECT_ID('dbo.spGetUserWithRoles', 'P') IS NOT NULL
    DROP PROCEDURE dbo.spGetUserWithRoles;
GO

CREATE PROCEDURE dbo.spGetUserWithRoles
@    UserName VARCHAR(50)
AS
BEGIN
    SET NOCOUNT ON;

    SELECT
        u.IdNumber,
        u.UserName,
        u.Email,
        u.FirstName,
        u.LastName,

```

```

        u.IsActive,
        r.RoleName
FROM dbo.Users u
LEFT JOIN dbo.UserRoles ur ON u.IdNumber = ur.IdNumber
LEFT JOIN dbo.Roles r ON ur.RoleId = r.RoleId
WHERE u.UserName = @UserName;
END
GO

```

```

===== */

```

.3 Stored Procedures – Research / Members

```

/* =====

```

```

-----

```

```

--SP 4 – CreateResearch

```

```

-----

```

```

IF OBJECT_ID('dbo.spCreateResearch', 'P') IS NOT NULL

```

```

    DROP PROCEDURE dbo.spCreateResearch;

```

```

GO

```

```

CREATE PROCEDURE dbo.spCreateResearch

```

```

@ ResearchCode    NVARCHAR(50)

```

```

@ ResearchName    NVARCHAR(200)

```

```

@ Description      NVARCHAR(MAX) = NULL,

```

```

@ TotalBudget      DECIMAL(18,2)

```

```

@ StartDate        DATE,

```

```

@ EndDate          DATE = NULL,

```

```

@ IsUnderCenter    BIT,

```

```

@ CenterId         INT = NULL,

```

```

@ PrimaryResearcherId CHAR(9)

```

```

AS

```

```

BEGIN

    SET NOCOUNT ON;

    IF EXISTS (SELECT 1 FROM dbo.Researches WHERE ResearchCode =
@ResearchCode)

        BEGIN

            RAISERROR(N;(1,16, 'קוד מחקר זה כבר קיים במערכת.',
            RETURN;

        END

    IF NOT EXISTS (SELECT 1 FROM dbo.Users WHERE IdNumber =
@PrimaryResearcherId AND IsActive = 1)

        BEGIN

            RAISERROR(N;(1,16, 'חוקר ראשי לא קיים או לא פעיל.',
            RETURN;

        END

    IF @IsUnderCenter = 1

        AND NOT EXISTS (SELECT 1 FROM dbo.ResearchCenters WHERE CenterId =
@CenterId)

        BEGIN

            RAISERROR(N;(1,16, 'מרכז מחקר לא קיים.',
            RETURN;

        END

    INSERT INTO dbo.Researches

    )    ResearchCode, ResearchName, Description, TotalBudget, StartDate, EndDate,

        IsUnderCenter, CenterId, PrimaryResearcherId, CreatedAt(

    VALUES

    @)    ResearchCode, @ResearchName, @Description, @TotalBudget, @StartDate,
@EndDate,

    @    IsUnderCenter, @CenterId, @PrimaryResearcherId, SYSUTCDATETIME;())

```

```
DECLARE @NewResearchId INT = SCOPE_IDENTITY;()
```

```
INSERT INTO dbo.ResearchMembers (ResearchId, IdNumber, ResearchRole, IsActive)  
VALUES (@NewResearchId, @PrimaryResearcherId, N'PI', 1);
```

```
SELECT*  
FROM dbo.Researches  
WHERE ResearchId = @NewResearchId;  
END  
GO
```

```
-----  
--SP 5 – GetResearchById  
-----
```

```
IF OBJECT_ID('dbo.spGetResearchById', 'P') IS NOT NULL  
    DROP PROCEDURE dbo.spGetResearchById;  
GO
```

```
CREATE PROCEDURE dbo.spGetResearchById  
@    ResearchId INT  
AS  
BEGIN  
    SET NOCOUNT ON;  
  
    SELECT*  
    FROM dbo.Researches  
    WHERE ResearchId = @ResearchId;  
END  
GO
```

```
-----
```

--SP 6 – GetResearchesByUser

IF OBJECT_ID('dbo.spGetResearchesByUser', 'P') IS NOT NULL

DROP PROCEDURE dbo.spGetResearchesByUser;

GO

CREATE PROCEDURE dbo.spGetResearchesByUser

@ IdNumber CHAR(9)

AS

BEGIN

SET NOCOUNT ON;

SELECT DISTINCT r*.

FROM dbo.Researches r

LEFT JOIN dbo.ResearchMembers rm

ON r.ResearchId = rm.ResearchId AND rm.IsActive = 1

WHERE rm.IdNumber = @IdNumber

OR r.PrimaryResearcherId = @IdNumber;

END

GO

--SP 7 – AddResearchMember

IF OBJECT_ID('dbo.spAddResearchMember', 'P') IS NOT NULL

DROP PROCEDURE dbo.spAddResearchMember;

GO

CREATE PROCEDURE dbo.spAddResearchMember

@ ResearchId INT,

@ IdNumber CHAR(9)

```

@ ResearchRole NVARCHAR(50)

AS

BEGIN

    SET NOCOUNT ON;

    IF NOT EXISTS (SELECT 1 FROM dbo.Researches WHERE ResearchId = @ResearchId)

    BEGIN

        RAISERROR) N;(1,16, 'מחקר לא קיים.')

        RETURN;

    END

    IF NOT EXISTS (SELECT 1 FROM dbo.Users WHERE IdNumber = @IdNumber AND
IsActive = 1)

    BEGIN

        RAISERROR) N;(1,16, 'משתמש לא קיים או לא פעיל.')

        RETURN;

    END

    IF EXISTS (SELECT 1 FROM dbo.ResearchMembers WHERE ResearchId =
@ResearchId AND IdNumber = @IdNumber)

    BEGIN

        UPDATE dbo.ResearchMembers

        SET ResearchRole = @ResearchRole,

            IsActive = 1

        WHERE ResearchId = @ResearchId AND IdNumber = @IdNumber;

    END

    ELSE

    BEGIN

        INSERT INTO dbo.ResearchMembers (ResearchId, IdNumber, ResearchRole,
IsActive)

        VALUES (@ResearchId, @IdNumber, @ResearchRole, 1);

    END

```

```
SELECT*
FROM dbo.ResearchMembers
WHERE ResearchId = @ResearchId AND IdNumber = @IdNumber;
END
GO
```

```
-----
--SP 8 – GetResearchMembers
-----
```

```
IF OBJECT_ID('dbo.spGetResearchMembers', 'P') IS NOT NULL
    DROP PROCEDURE dbo.spGetResearchMembers;
GO
```

```
CREATE PROCEDURE dbo.spGetResearchMembers
@    ResearchId INT
AS
BEGIN
    SET NOCOUNT ON;
```

```
    SELECT
        rm.ResearchId,
        rm.IdNumber,
        rm.ResearchRole,
        rm.IsActive,
        u.FirstName,
        u.LastName,
        u.Email
    FROM dbo.ResearchMembers rm
    JOIN dbo.Users u ON rm.IdNumber = u.IdNumber
    WHERE rm.ResearchId = @ResearchId;
```

END

GO

===== */

.4 Stored Procedures – Budget / Files / Payments

/* =====

--SP 9 – GetBudgetCategories

IF OBJECT_ID('dbo.spGetBudgetCategories', 'P') IS NOT NULL

DROP PROCEDURE dbo.spGetBudgetCategories;

GO

CREATE PROCEDURE dbo.spGetBudgetCategories

AS

BEGIN

SET NOCOUNT ON;

SELECT CategoryId, CategoryName, Description

FROM dbo.BudgetCategories

ORDER BY CategoryName;

END

GO

--SP 10 – AddFile

IF OBJECT_ID('dbo.spAddFile', 'P') IS NOT NULL

DROP PROCEDURE dbo.spAddFile;

GO

```

CREATE PROCEDURE dbo.spAddFile
@ ResearchId INT,
@ OriginalFileName NVARCHAR(255)
@ StoredFileName NVARCHAR(255)
@ RelativePath NVARCHAR(500)
@ ContentType NVARCHAR(100) = NULL,
@ UploadedById CHAR(9)
AS
BEGIN
    SET NOCOUNT ON;

    IF NOT EXISTS (SELECT 1 FROM dbo.Researches WHERE ResearchId = @ResearchId)
    BEGIN
        RAISERROR) N;(1 ,16 ,'.מחקר לא קיים.'
        RETURN;
    END

    IF NOT EXISTS (SELECT 1 FROM dbo.Users WHERE IdNumber = @UploadedById AND
IsActive = 1)
    BEGIN
        RAISERROR) N;(1 ,16 ,'.משתמש מעלה לא קיים או לא פעיל.'
        RETURN;
    END

    INSERT INTO dbo.Files
) ResearchId, OriginalFileName, StoredFileName, RelativePath, ContentType,
UploadedById, UploadedAt(
VALUES
@) ResearchId, @OriginalFileName, @StoredFileName, @RelativePath,
@ContentType, @UploadedById, SYSUTCDATETIME;((

```

```

        SELECT SCOPE_IDENTITY() AS NewFileId;
END
GO

-----

--SP 11 – CreatePaymentRequest
-----

IF OBJECT_ID('dbo.spCreatePaymentRequest', 'P') IS NOT NULL
    DROP PROCEDURE dbo.spCreatePaymentRequest;
GO

CREATE PROCEDURE dbo.spCreatePaymentRequest
@   ResearchId  INT,
@   RequestedById CHAR,(9)
@   CategoryId  INT,
@   Amount      DECIMAL,(18,2)
@   Description  NVARCHAR(MAX) = NULL,
@   FileId      INT = NULL
AS
BEGIN
    SET NOCOUNT ON;

    IF NOT EXISTS (SELECT 1 FROM dbo.Researches WHERE ResearchId = @ResearchId)
    BEGIN
        RAISERROR(N;(1,16, 'מחקר לא קיים.',16,1)
        RETURN;
    END

    IF NOT EXISTS (SELECT 1 FROM dbo.Users WHERE IdNumber = @RequestedById
    AND IsActive = 1)
    BEGIN
        RAISERROR(N;(1,16, 'משתמש לא קיים או לא פעיל.',16,1)

```

```
RETURN;  
END
```

```
IF NOT EXISTS (SELECT 1 FROM dbo.ResearchMembers  
WHERE ResearchId = @ResearchId  
AND IdNumber = @RequestedById  
AND IsActive = 1(
```

```
BEGIN  
RAISERROR) N;(1,16, 'המשתמש לא משויך למחקר הזה ולכן לא יכול להגיש בקשה.'  
RETURN;  
END
```

```
IF NOT EXISTS (SELECT 1 FROM dbo.BudgetCategories WHERE CategoryId =  
@CategoryId)  
BEGIN  
RAISERROR) N;(1,16, 'קטגוריית תקציב לא קיימת.'  
RETURN;  
END
```

```
IF @FileId IS NOT NULL  
BEGIN  
IF NOT EXISTS (SELECT 1 FROM dbo.Files WHERE FileId = @FileId AND ResearchId  
= @ResearchId)  
BEGIN  
RAISERROR) N;(1,16, 'הקובץ המצורף לא קיים או לא שייך למחקר הזה.'  
RETURN;  
END  
END
```

```
INSERT INTO dbo.PaymentRequests  
) ResearchId, RequestedById, CategoryId, Amount, Currency, RequestDate,  
Description, FileId, Status(
```

VALUES

@) ResearchId, @RequestedById, @CategoryId, @Amount, N'ILS',
SYSUTCDATETIME(), @Description, @FileId, N'Pending;('

SELECT SCOPE_IDENTITY() AS NewPaymentRequestId;

END

GO

--SP 12 – GetPaymentRequestsByResearch

IF OBJECT_ID('dbo.spGetPaymentRequestsByResearch', 'P') IS NOT NULL

DROP PROCEDURE dbo.spGetPaymentRequestsByResearch;

GO

CREATE PROCEDURE dbo.spGetPaymentRequestsByResearch

@ ResearchId INT

AS

BEGIN

SET NOCOUNT ON;

SELECT

pr.PaymentRequestId,

pr.ResearchId,

pr.RequestedById,

u.FirstName,

u.LastName,

pr.CategoryId,

bc.CategoryName,

pr.Amount,

pr.Currency,

pr.RequestDate,

```
        pr.Description,
        pr.FileId,
        pr.Status
FROM dbo.PaymentRequests pr
JOIN dbo.Users u ON pr.RequestedById = u.IdNumber
JOIN dbo.BudgetCategories bc ON pr.CategoryId = bc.CategoryId
WHERE pr.ResearchId = @ResearchId;
END
GO
```

```
-----
--SP 13 – GetPaymentRequestsByUser
-----
```

```
IF OBJECT_ID('dbo.spGetPaymentRequestsByUser', 'P') IS NOT NULL
    DROP PROCEDURE dbo.spGetPaymentRequestsByUser;
GO
```

```
CREATE PROCEDURE dbo.spGetPaymentRequestsByUser
```

```
@    RequestedById CHAR(9)
```

```
AS
```

```
BEGIN
```

```
    SET NOCOUNT ON;
```

```
    SELECT
```

```
        pr.PaymentRequestId,
```

```
        pr.ResearchId,
```

```
        r.ResearchName,
```

```
        pr.CategoryId,
```

```
        bc.CategoryName,
```

```
        pr.Amount,
```

```
        pr.Currency,
```

```

        pr.RequestDate,
        pr.Description,
        pr.FileId,
        pr.Status
    FROM dbo.PaymentRequests pr
    JOIN dbo.Researches r ON pr.ResearchId = r.ResearchId
    JOIN dbo.BudgetCategories bc ON pr.CategoryId = bc.CategoryId
    WHERE pr.RequestedById = @RequestedById;
END
GO

```

```

-----
--SP 14 – CreatePaymentApproval
-----

```

```

IF OBJECT_ID('dbo.spCreatePaymentApproval', 'P') IS NOT NULL
    DROP PROCEDURE dbo.spCreatePaymentApproval;
GO

```

```

CREATE PROCEDURE dbo.spCreatePaymentApproval
@    PaymentRequestId INT,
@    ApprovedById    CHAR,(9)
@    ApprovalRole    NVARCHAR(50), -- 'CenterManager' I\ 'Secretary'
@    Status          NVARCHAR(20), -- 'Approved' I\ 'Rejected'
@    Comment         NVARCHAR(MAX) = NULL
AS
BEGIN
    SET NOCOUNT ON;

    DECLARE @ResearchId INT;
    DECLARE @IsUnderCenter BIT;

```

```

SELECT
@      ResearchId = pr.ResearchId
FROM dbo.PaymentRequests pr
WHERE pr.PaymentRequestId = @PaymentRequestId;

IF @ResearchId IS NULL
BEGIN
    RAISERROR) N;(1 ,16 ,','נמצאה.'
    RETURN;
END

IF NOT EXISTS (SELECT 1 FROM dbo.Users WHERE IdNumber = @ApprovedById AND
IsActive = 1)
BEGIN
    RAISERROR) N;(1 ,16 ,','פעיל.' או לא קיים
    RETURN;
END

IF @Status NOT IN (N'Approved', N'Rejected')
BEGIN
    RAISERROR) N;(1 ,16 ,','תקין.'
    RETURN;
END

SELECT @IsUnderCenter = r.IsUnderCenter
FROM dbo.Researches r
WHERE r.ResearchId = @ResearchId;

IF @ApprovalRole = N'Secretary' AND @IsUnderCenter = 1
    AND NOT EXISTS)
    SELECT 1

```

```

FROM dbo.PaymentApprovals pa

WHERE pa.PaymentRequestId = @PaymentRequestId

AND pa.ApprovalRole = N'CenterManager'

AND pa.Status = N'Approved'

(
BEGIN

    RAISERROR) N;(1 ,16 ,','אשר לא ניתן למזכיר לפני שמנהל מרכז מחקר אישר.'

    RETURN;

END

INSERT INTO dbo.PaymentApprovals

)    PaymentRequestId, ApprovedById, ApprovalRole, Status, ApprovalDate,
Comment(

VALUES

(@)    PaymentRequestId, @ApprovedById, @ApprovalRole, @Status,
SYSUTCDATETIME(), @Comment;(

IF @Status = N'Rejected'

BEGIN

    UPDATE dbo.PaymentRequests

    SET Status = N'Rejected'

    WHERE PaymentRequestId = @PaymentRequestId;

END

ELSE IF @Status = N'Approved' AND @ApprovalRole = N'Secretary'

BEGIN

    UPDATE dbo.PaymentRequests

    SET Status = N'Approved'

    WHERE PaymentRequestId = @PaymentRequestId;

END

SELECT TOP 1*

FROM dbo.PaymentApprovals

```

```

WHERE PaymentRequestId = @PaymentRequestId

ORDER BY ApprovalId DESC;

END

GO

-----

--SP 15 – spGetResearchBudgetDetails
-- פירוט תקציב למחקר: סך הוצאות, לפי קטגוריה, פירוט, יתרה
-----

IF OBJECT_ID('dbo.spGetResearchBudgetDetails', 'P') IS NOT NULL

    DROP PROCEDURE dbo.spGetResearchBudgetDetails;

GO

CREATE PROCEDURE dbo.spGetResearchBudgetDetails

@    ResearchId INT

AS

BEGIN

    SET NOCOUNT ON;

    -- פרטי המחקר
    SELECT

        ResearchId,

        ResearchName,

        ResearchCode,

        TotalBudget,

        StartDate,

        EndDate,

        CreatedAt

    FROM dbo.Researches

    WHERE ResearchId = @ResearchId;

```

-- 1. סך כל ההוצאות שאושרו

```
SELECT  
  
    SUM(Amount) AS TotalApprovedExpenses  
  
FROM dbo.PaymentRequests  
  
WHERE ResearchId = @ResearchId  
  
    AND Status = 'Approved';
```

-- 2. הוצאות לפי קטגוריה

```
SELECT  
  
    pr.CategoryId,  
    bc.CategoryName,  
  
    SUM(pr.Amount) AS TotalByCategory  
  
FROM dbo.PaymentRequests pr  
  
JOIN dbo.BudgetCategories bc  
  
    ON pr.CategoryId = bc.CategoryId  
  
WHERE pr.ResearchId = @ResearchId  
  
    AND pr.Status = 'Approved'  
  
GROUP BY pr.CategoryId, bc.CategoryName;
```

-- 3. פירוט מלא של כל הבקשות (Approved / Pending / Rejected)

```
SELECT  
  
    pr.PaymentRequestId,  
  
    pr.Amount,  
  
    pr.RequestDate,  
  
    pr.Description,  
  
    pr.Status,  
  
    pr.FileId,  
  
    bc.CategoryName,
```

```

        u.FirstName + N' ' + u.LastName AS RequestedBy
FROM dbo.PaymentRequests pr
JOIN dbo.Users u ON pr.RequestedById = u.IdNumber
JOIN dbo.BudgetCategories bc ON pr.CategoryId = bc.CategoryId
WHERE pr.ResearchId = @ResearchId;

```

-- 4. יתרת התקציב

```

SELECT
    r.TotalBudget-
    ISNULL(
        SELECT SUM(pr.Amount)
        FROM dbo.PaymentRequests pr
        WHERE pr.ResearchId = @ResearchId
        AND pr.Status = 'Approved'
    ) AS RemainingBudget
FROM dbo.Researches r
WHERE r.ResearchId = @ResearchId;
END;
GO
CREATE OR ALTER PROC spDeletePaymentRequestIfPending
    @PaymentRequestId INT
AS
BEGIN
    -- כאן SET NOCOUNT ON בלי --

    DELETE FROM PaymentRequests
    WHERE PaymentRequestId = @PaymentRequestId
    AND Status = 'Pending';
END;
GO

```

CREATE OR ALTER PROC spUpdatePaymentRequestIfPending

 @PaymentRequestId INT,

 @CategoryId INT,

 @Amount DECIMAL(18,2),

 @Description NVARCHAR(MAX) = NULL,

 @FileId INT = NULL

AS

BEGIN

 -- בלי SET NOCOUNT ON

 UPDATE PaymentRequests

 SET CategoryId = @CategoryId,

 Amount = @Amount,

 Description = @Description,

 FileId = @FileId

 WHERE PaymentRequestId = @PaymentRequestId

 AND Status = 'Pending';

END;

GO