-- ================================================

-- Author:        Asmita Mahamuni

-- Create date:   2024-06-24

-- Description:   This stored procedure handles the workflow for updating

--                student subject allotments based on subject requests.

-- ================================================

CREATE PROCEDURE UpdateSubjectAllotments

AS

BEGIN

    -- Set NOCOUNT on to prevent extra result sets from interfering with SELECT statements.

    SET NOCOUNT ON;

    -- Variable declaration

    DECLARE @StudentID VARCHAR(50), @SubjectID VARCHAR(50), @CurrentSubjectID VARCHAR(50);

    -- Cursor to iterate through each record in SubjectRequest table

    DECLARE SubjectRequestCursor CURSOR FOR

    SELECT StudentId, SubjectId FROM SubjectRequest;

    OPEN SubjectRequestCursor;

    FETCH NEXT FROM SubjectRequestCursor INTO @StudentID, @SubjectID;

    -- Loop through each row in SubjectRequest table

    WHILE @@FETCH\_STATUS = 0

    BEGIN

        -- Check if the student already has an active subject

        SELECT @CurrentSubjectID = SubjectId

        FROM SubjectAllotments

        WHERE StudentId = @StudentID AND Is\_valid = 1;

        -- If the student exists in the SubjectAllotments table

        IF @CurrentSubjectID IS NOT NULL

        BEGIN

            -- Check if the requested subject is different from the current subject

            IF @CurrentSubjectID <> @SubjectID

            BEGIN

                -- Invalidate the current subject

                UPDATE SubjectAllotments

                SET Is\_valid = 0

                WHERE StudentId = @StudentID AND Is\_valid = 1;

                -- Insert the new requested subject as valid

                INSERT INTO SubjectAllotments (StudentId, SubjectId, Is\_valid)

                VALUES (@StudentID, @SubjectID, 1);

            END

        END

        ELSE

        BEGIN

            -- If the student does not exist in the SubjectAllotments table, insert the new requested subject as valid

            INSERT INTO SubjectAllotments (StudentId, SubjectId, Is\_valid)

            VALUES (@StudentID, @SubjectID, 1);

        END

        FETCH NEXT FROM SubjectRequestCursor INTO @StudentID, @SubjectID;

    END

    -- Cleanup

    CLOSE SubjectRequestCursor;

    DEALLOCATE SubjectRequestCursor;

END;

GO