**Stored Procedures:**

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspAdminDetailsInsert`(

IN i\_FirstName varchar(100) ,

IN i\_LastName varchar(100) ,

IN i\_EmailAddress varchar(100) ,

IN i\_contactNumber bigint ,

IN i\_UserName varchar(100) ,

IN i\_PasswordforUserName varchar(100),

OUT o\_AdminID int

)

Begin

Insert into tbladminregistration

(

FirstName,

LastName,

EmailAddress,

contactNumber,

UserName,

PasswordforUserName

)

values

(

i\_FirstName,

i\_LastName,

i\_EmailAddress,

i\_contactNumber,

i\_UserName,

i\_PasswordforUserName);

set o\_AdminID=(select AdminID from tbladminregistration order by 1 desc LIMIT 1);

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspAdminLoginGet`(

IN i\_UserName varchar(100),

IN i\_AdminID int,

IN i\_Password varchar(100)

)

BEGIN

Select \* from tblAdminRegistration where

username=i\_UserName and AdminID=i\_AdminID and PasswordforUserName=i\_Password;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspBookDelete`(

IN i\_BookCode varchar(100) ,

IN i\_BookName varchar(1000),

IN i\_BookAvailability int,

IN i\_AdminID int

)

BEGIN

declare i\_BookID int;

set i\_BookID=(Select BookID from tblBookDetails where BookCode=i\_BookCode);

Update tblBookDetails

set IsActive=0,

UpdatedUserID=i\_AdminID,

UpdatedDate=current\_timestamp()

where BookID=i\_BookID;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspBookDetailsByBookCodeGet`(IN i\_BookCode varchar(25))

BEGIN

Select BookCode,BookName,BookAvailability from tblbookdetails where BookCode=i\_BookCode and isactive=1;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspBookDetailsGet`(in\_Filter varchar(100))

Begin

IF(in\_Filter="ALL") THEN

Select BookCode,BookName,BookAvailability from tblBookDetails where isactive=1;

ELSE

Select BookCode,BookName,BookAvailability from tblBookDetails where

DepartmentID=Cast(in\_Filter as unsigned) and isactive=1;

end if;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspBookInsert`(

IN i\_BookCode varchar(100) ,

IN i\_DepartmentID varchar(100) ,

IN i\_BookName varchar(100) ,

IN i\_BookAvailability bigint ,

IN i\_AdminID varchar(100) ,

IN i\_MaxBookAvailability varchar(100),

OUT o\_BookID int

)

BEGIN

Insert into tblBookDetails

(

BookCode,

DepartmentID,

BookName,

BookAvailability,

UpdatedUserID,

UpdatedDate,

MaximumBookAvailable,

IsActive

)

values

( i\_BookCode,

i\_DepartmentID,

i\_BookName,

i\_BookAvailability,

i\_AdminID,

current\_timestamp(),

i\_MaxBookAvailability,

1);

set o\_BookID=(select BookID from tblBookDetails order by 1 desc LIMIT 1);

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspBookReturn`(

IN i\_BookCode varchar(100) ,

IN i\_StudentID int ,

OUT o\_BookAvailability int,

OUT o\_BookID int,

OUT o\_IsBookCanBeReturned int,

OUT o\_IsBookAlreadyReturned int

)

Begin

declare IsBookAlreadyReturned int;

declare MaximumAvailability int;

set o\_BookAvailability=(select BookAvailability from tblBookDetails where BookCode=i\_BookCode and isactive=1);

set o\_BookID=(select BookID from tblBookDetails where BookCode=i\_BookCode and isactive=1);

set MaximumAvailability=(Select MaximumBookAvailable from tblBookDetails where BookCode=i\_BookCode and isactive=1);

if(o\_BookAvailability<=MaximumAvailability) then

set o\_IsBookCanBeReturned=1;

else

set o\_IsBookCanBeReturned=0;

end if;

set IsBookAlreadyReturned=(Select count(\*) from tblStudentsXBooks where StudentID=i\_StudentID and BookID=o\_BookID and IsActive=1);

if(IsBookAlreadyReturned!=0) then

set o\_IsBookAlreadyReturned=1;

else

set o\_IsBookAlreadyReturned=0;

end if;

if(o\_IsBookAlreadyReturned=1 && o\_IsBookCanBeReturned=1) then

Update tblBookDetails set BookAvailability=(BookAvailability+1) where BookCode= i\_BookCode and isactive=1;

update tblStudentsXBooks set isActive=0 where StudentID=i\_StudentID and BookID=o\_BookID and isactive=1;

set o\_BookAvailability=(select BookAvailability from tblBookDetails where BookCode=i\_BookCode and isactive=1);

end if;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspBookTake1`(

IN i\_BookCode varchar(100) ,

IN i\_StudentID int ,

OUT o\_BookAvailability int,

OUT o\_IsMaxBookLimitReached int,

OUT o\_BookAlreadyTaken int,

OUT o\_BookID int)

BEGIN

declare Max\_limit int;

declare Same\_Book int;

set o\_BookAvailability=(select BookAvailability from tblBookDetails where BookCode=i\_BookCode and isactive=1);

set o\_BookID=(select BookID from tblBookDetails where BookCode=i\_BookCode and isactive=1);

set Max\_limit=(select count(\*) from tblStudentsXBooks where StudentID=i\_StudentID and isactive=1);

if(Max\_limit=5) then

set o\_IsMaxBookLimitReached=1;

else

set o\_IsMaxBookLimitReached=0;

End if;

set Same\_Book=(select count(\*) from tblStudentsXBooks where StudentID=i\_StudentID and BookID=o\_BookID and isactive=1);

if(Same\_Book=0) then

set o\_BookAlreadyTaken=0;

else

set o\_BookAlreadyTaken=1;

end if;

if(o\_BookAlreadyTaken=0 && o\_IsMaxBookLimitReached=0) then

Update tblBookDetails set BookAvailability=(BookAvailability-1) where BookCode= i\_BookCode and isactive=1;

Insert into tblStudentsXBooks(StudentID,BookID,IsActive) values (i\_StudentID,o\_BookID,1);

set o\_BookAvailability=(select BookAvailability from tblBookDetails where BookCode=i\_BookCode and isactive=1);

End if;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspBookUpdate`(

IN i\_BookCode varchar(100) ,

IN i\_BookName varchar(1000),

IN i\_BookAvailability int,

IN i\_AdminID int

)

BEGIN

declare i\_BookID int;

set i\_BookID=(Select BookID from tblBookDetails where BookCode=i\_BookCode and isactive=1);

Update tblBookDetails

set BookName=i\_BookName,

BookAvailability=i\_BookAvailability,

UpdatedUserID=i\_AdminID,

UpdatedDate=current\_timestamp()

where BookID=i\_BookID and isactive=1;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspCourseDelete`(

IN i\_CourseCode varchar(100) ,

IN i\_CourseType varchar(1000),

IN i\_CourseName varchar(1000),

IN i\_AdminID int,

OUT o\_iscoursecodeexists int

)

BEGIN

Declare i\_CourseID int;

set i\_CourseID=(Select CourseID from tblCourseDetails where Coursecode=i\_CourseCode);

IF EXISTS(Select CourseID from tblCourseDetails where Coursecode=i\_CourseCode) THen

set o\_iscoursecodeexists=1;

Update tblcoursedetails set IsActive=0 where CourseID=i\_CourseID;

else

set o\_iscoursecodeexists=0;

END IF;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspCourseDetailsGet`()

BEGIN

Select CourseCode,CourseType,CourseName from tblcoursedetails where IsActive=1;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspCourseInsert`(

IN i\_CourseCode varchar(100) ,

IN i\_CourseType varchar(100),

IN i\_CourseName varchar(100) ,

in\_Filter varchar(100),

IN i\_AdminID int,

OUT o\_CourseID int,

OUT o\_isSameCourseCodeExists int

)

BEGIN

IF EXISTS(Select CourseID from tblcoursedetails where CourseCode=i\_CourseCode) THEN

set o\_isSameCourseCodeExists=1;

ELSE

BEGIN

set o\_isSameCourseCodeExists=0;

Insert into tblcoursedetails

(

CourseCode,

CourseType,

DepartmentID,

IsActive,

UpdatedUserID,

UpdatedDate,

CourseName

)

values

( i\_CourseCode,

i\_CourseType,

Cast(in\_Filter as unsigned),

1,

i\_AdminID,

current\_timestamp(),

i\_CourseName

);

set o\_CourseID=(select CourseID from tblcoursedetails order by 1 desc LIMIT 1);

END;

END IF;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspCourseUpdate`(

IN i\_CourseCode varchar(100) ,

IN i\_CourseType varchar(100) ,

IN i\_CourseName varchar(1000),

IN i\_AdminID int

)

BEGIN

declare i\_CourseID int;

set i\_CourseID=(select CourseID from tblCourseDetails where CourseCode=i\_CourseCode);

Update tblcoursedetails

set CourseCode=i\_CourseCode,

CourseName=i\_CourseName,

UpdatedUserID=i\_AdminID,

UpdatedDate=current\_timestamp()

where CourseID=i\_CourseID and isactive=1;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspDepartmentDelete`(

IN i\_DepartmentCode varchar(100) ,

IN i\_DepartmentName varchar(1000),

IN i\_AdminID int,

OUT o\_isbookcodeexists int

)

BEGIN

Declare i\_DepartmentID int;

set i\_DepartmentID=(Select DepartmentID from tblDepartmentDetails where Departmentcode=i\_DepartmentCode);

IF EXISTS(Select DepartmentID from tblDepartmentDetails where Departmentcode=i\_DepartmentCode) THen

set o\_isbookcodeexists=1;

Update tblDepartmentDetails set IsActive=0 where DepartmentID=i\_DepartmentID;

else

set o\_isbookcodeexists=0;

END IF;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspDepartmentDetailsGet`()

BEGIN

Select DepartmentCode,DepartmentName from tblDepartmentDetails where IsActive=1;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspDepartmentInsert`(

IN i\_DepartmentCode varchar(100) ,

IN i\_DepartmentName varchar(100) ,

IN i\_AdminID int,

OUT o\_DepartmentID int,

OUT o\_isSameDepartmentCodeExists int

)

BEGIN

IF EXISTS(Select DepartmentID from tblDepartmentDetails where DepartmentCode=i\_DepartmentCode) THEN

set o\_isSameDepartmentCodeExists=1;

ELSE

BEGIN

set o\_isSameDepartmentCodeExists=0;

Insert into tblDepartmentDetails

(

DepartmentName,

DepartmentCode,

IsActive,

UpdatedUserID,

UpdatedDate

)

values

( i\_DepartmentName,

i\_DepartmentCode,

1,

i\_AdminID,

current\_timestamp()

);

set o\_DepartmentID=(select DepartmentID from tblDepartmentDetails order by 1 desc LIMIT 1);

END;

END IF;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspDepartmentUpdate`(IN i\_DepartmentCode varchar(100) ,

IN i\_DepartmentName varchar(1000),

IN i\_AdminID int

)

BEGIN

declare i\_DepartmentID int;

set i\_DepartmentID=(select departmentID from tblDepartmentDetails where DepartmentCode=i\_DepartmentCode);

Update tblDepartmentDetails

set DepartmentCode=i\_DepartmentCode,

DepartmentName=i\_DepartmentName,

UpdatedUserID=i\_AdminID,

UpdatedDate=current\_timestamp()

where DepartmentID=i\_DepartmentID and isactive=1;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspFacultyAssignInsert`(

IN i\_AssignFacultyCode varchar(100) ,

IN in\_Filter int,

OUT o\_isSameSubjectCodeAssigned int

)

BEGIN

declare i\_FacultyId int;

set i\_FacultyId=(Select FacultyNumber from tblFacultyDetails where FacultyCode=i\_AssignFacultyCode);

IF EXISTS(Select SubjectID,FacultyID from tblFacultyXSubjects where FacultyID=i\_FacultyID and SubjectID=in\_Filter) THEN

set o\_isSameSubjectCodeAssigned=1;

ELSE

BEGIN

set o\_isSameSubjectCodeAssigned=0;

Insert into tblFacultyXSubjects

(

FacultyID,

SubjectID,

UpdatedDate,

IsActive

)

values

(

i\_FacultyID,

Cast(in\_Filter as unsigned),

current\_timestamp(),

1

);

END;

END IF;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspFacultyDelete`(

i\_FacultyCode varchar(45),

i\_FacultyFirstName varchar(1000),

i\_FacultyLastName varchar(1000),

i\_Qualification varchar(45),

i\_Designation varchar(45)

)

BEGIN

Declare i\_FacultyID int;

Set i\_FacultyID=(select FacultyNumber from tblFacultyDetails where FacultyCode=i\_FacultyCode);

Delete from tblFacultyDetails where FacultyNumber=i\_FacultyID;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspFacultyDetailsGet`()

BEGIN

Select FacultyCode,FacultyFirstName,FacultyLastName,Qualification,Designation from tblfacultydetails;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspFacultyInsert`(

IN i\_FacultyCode varchar(100) ,

IN i\_FacultyFirstName varchar(1000),

IN i\_FacultyLastName varchar(1000) ,

i\_Qualification varchar(100),

IN i\_Designation Varchar(100),

OUT o\_FacultyID int,

OUT o\_isSameFacultyCodeExists int

)

BEGIN

IF EXISTS(Select FacultyNumber from tblFacultyDetails where FacultyCode=i\_FacultyCode) THEN

set o\_isSameFacultyCodeExists=1;

ELSE

BEGIN

set o\_isSameFacultyCodeExists=0;

Insert into tblFacultydetails

(

FacultyFirstName,

FacultyLastName,

Qualification,

Designation,

FacultyCode

)

values

( i\_FacultyFirstName,

i\_FacultyLastName,

i\_Qualification,

i\_Designation,

i\_FacultyCode

);

set o\_FacultyID=(select FacultyNumber from tblfacultydetails order by 1 desc LIMIT 1);

END;

END IF;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspFacultyUpdate`(

IN i\_FacultyCode varchar(100) ,

IN i\_FacultyFirstName varchar(100) ,

IN i\_FacultyLastName varchar(1000),

IN i\_Qualification varchar(100),

IN i\_Designation varchar(100)

)

BEGIN

declare i\_FacultyID int;

set i\_FacultyID=(select FacultyID from tblfacultydetails where FacultyCode=i\_FacultyCode);

Update tblfacultydetails

set FacultyCode=i\_FacultyCode,

FirstName=i\_FacultyFirstName,

LastName=i\_FacultyLastName,

Qualification=i\_Qualification,

Designation=i\_Designation

where FacultyNumber=i\_FacultyID;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspStudentDetailsGet`(IN i\_StudentID int,

OUT o\_FirstName varchar(100),

OUT o\_LastName varchar(100),

OUT o\_EmailAddress varchar(100),

OUT o\_ContactNumber bigint

)

BEGIN

set o\_FirstName=(Select FirstName from tblStudentRegistration where StudentID=i\_StudentID);

set o\_LastName=(Select LastName from tblStudentRegistration where StudentID=i\_StudentID);

set o\_EmailAddress=(Select EmailAddress from tblStudentRegistration where StudentID=i\_StudentID);

set o\_ContactNumber=(Select ContactNumber from tblStudentRegistration where StudentID=i\_StudentID);

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspStudentDetailsInsert`(

IN i\_FirstName varchar(100) ,

IN i\_LastName varchar(100) ,

IN i\_EmailAddress varchar(100) ,

IN i\_contactNumber bigint ,

IN i\_UserName varchar(100) ,

IN i\_PasswordforUserName varchar(100),

OUT o\_StudentID int

)

Begin

Insert into tblstudentregistration

(

FirstName,

LastName,

EmailAddress,

contactNumber,

UserName,

PasswordforUserName

)

values

(

i\_FirstName,

i\_LastName,

i\_EmailAddress,

i\_contactNumber,

i\_UserName,

i\_PasswordforUserName);

set o\_StudentID=(select studentid from tblstudentregistration order by 1 desc LIMIT 1);

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspStudentLoginGet`(

IN i\_UserName varchar(100),

IN i\_StudentID int,

IN i\_Password varchar(100)

)

BEGIN

Select \* from tblstudentregistration where

username=i\_UserName and studentID=i\_StudentID and PasswordforUserName=i\_Password;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspSubjectdeEnroll`(

IN i\_SubjectCode varchar(12),

IN i\_StudentID int,

out o\_SubjectAlreadyEnrolled int

)

BEGIN

declare i\_SubjectID int;

set i\_SubjectID=(Select SubjectID from tblSubjectDetails where SubjectCode=i\_SubjectCode);

IF EXISTS(Select StudentXSubjectID from tblStudentXSubjectsEnroll where StudentID=i\_StudentID and SubjectID=i\_SubjectID and Isactive=1) THen

set o\_SubjectAlreadyEnrolled=1;

Update tblstudentxsubjectsenroll set IsActive=0 where StudentID=i\_StudentID and SubjectID=i\_SubjectID;

else

set o\_SubjectAlreadyEnrolled=0;

END IF;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspSubjectDelete`(

IN i\_SubjectCode varchar(100) ,

IN i\_DepartmentCode varchar(1000),

IN i\_CourseCode varchar(1000),

IN i\_SubjectName varchar(1000),

IN i\_AdminID int,

OUT o\_issubjectcodeexists int

)

BEGIN

Declare i\_SubjectID int;

Declare i\_DepartmentID int;

Declare i\_CourseID int;

set i\_DepartmentID=(Select DepartmentID from tbldepartmentdetails where Departmentcode=i\_DepartmentCode);

set i\_CourseID=(Select CourseID from tblcoursedetails where Coursecode=i\_CourseCode);

set i\_SubjectID=(Select SubjectID from tblSubjectDetails where SubjectCode=i\_SubjectCode and DepartmentID=i\_DepartmentID and CourseID=i\_CourseID );

IF EXISTS(Select SubjectID from tblSubjectDetails where SubjectCode=i\_SubjectCode and DepartmentID=i\_DepartmentID and CourseID=i\_CourseID) THen

set o\_issubjectcodeexists=1;

Update tblsubjectdetails set IsActive=0 where SubjectID=i\_SubjectID;

else

set o\_issubjectcodeexists=0;

END IF;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspSubjectDetailsGet`()

BEGIN

Select distinct sd.SubjectCode as SubjectCode,dd.DepartmentCode as DepartmentCode,cd.CourseCode as CourseCode,sd.SubjectName as SubjectName

from tblSubjectdetails sd

inner join tblDepartmentDetails dd on dd.departmentID=sd.departmentid

inner join tblCourseDetails cd on cd.courseid=sd.courseid

where sd.isactive=1;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspSubjectEnroll`(

IN i\_SubjectCode varchar(12),

IN i\_StudentID int,

out o\_SubjectAlreadyEnrolled int

)

BEGIN

declare i\_SubjectID int;

set i\_SubjectID=(Select SubjectID from tblSubjectDetails where SubjectCode=i\_SubjectCode);

IF EXISTS(Select StudentXSubjectID from tblStudentXSubjectsEnroll where StudentID=i\_StudentID and SubjectID=i\_SubjectID and IsActive=1) THen

set o\_SubjectAlreadyEnrolled=1;

else

set o\_SubjectAlreadyEnrolled=0;

INSERT INTO tblstudentxsubjectsenroll

(

`StudentID`,

`SubjectID`,

`IsActive`)

VALUES

(

i\_StudentID,

i\_SubjectID,

1);

END IF;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspSubjectDetailsGetforStudent`(IN i\_StudentID int)

BEGIN

Select SubjectCode,SubjectName from tblSubjectDetails sd

inner join tblstudentXsubjectsenroll sse on sse.SubjectID=sd.SubjectID where sse.StudentID=i\_StudentID and sse.isActive=1 and sd.isactive=1;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspSubjectInsert`(

IN i\_SubjectCode varchar(100) ,

IN i\_Departmentcode varchar(100),

IN i\_CourseCode varchar(100) ,

IN i\_SubjectName varchar(1000),

IN i\_AdminID int,

OUT o\_SubjectID int,

OUT o\_isSameSubjectCodeExists int

)

BEGIN

declare i\_DepartmentID int;

declare i\_CourseID int;

set i\_DepartmentID=(Select DepartmentID from tbldepartmentdetails where DepartmentCode=i\_DepartmentCode);

set i\_CourseID=(Select CourseID from tblcoursedetails where CourseCode=i\_CourseCode);

IF EXISTS(Select SubjectCode from tblsubjectdetails where SubjectCode=i\_SubjectCode) THEN

set o\_isSameSubjectCodeExists=1;

ELSE

BEGIN

set o\_isSameSubjectCodeExists=0;

Insert into tblSubjectdetails

(

SubjectCode,

DepartmentID,

CourseID,

IsActive,

UpdatedUserID,

UpdatedDate,

SubjectName

)

values

( i\_SubjectCode,

i\_DepartmentID,

i\_CourseID,

1,

i\_AdminID,

current\_timestamp(),

i\_SubjectName

);

set o\_SubjectID=(select SubjectID from tblSubjectdetails order by 1 desc LIMIT 1);

END;

END IF;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspSubjectUpdate`(

IN i\_SubjectCode varchar(100) ,

IN i\_DepartmentCode varchar(100) ,

IN i\_CourseCode varchar(1000),

IN i\_SubjectName varchar(1000),

IN i\_AdminID int

)

BEGIN

declare i\_SubjectID int;

declare i\_CourseID int;

declare i\_DepartmentID int;

set i\_SubjectID=(select SubjectId from tblsubjectdetails where SubjectCode=i\_SubjectCode);

set i\_CourseID=(select CourseID from tblCourseDetails where CourseCode=i\_CourseCode);

set i\_DepartmentID=(select DepartmentId from tbldepartmentdetails where DepartmentCode=i\_DepartmentCode);

Update tblsubjectdetails

set SubjectCode=i\_SubjectCode,

DepartmentID=i\_DepartmentID,

CourseID=i\_CourseID,

SubjectName=i\_SubjectName,

UpdatedUserID=i\_AdminID,

UpdatedDate=current\_timestamp()

where SubjectID=i\_SubjectID and isactive=1;

END$$

DELIMITER ;

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `uspSubjectDetailsGetforCourses`(IN in\_Filter int)

BEGIN

Select SubjectCode,SubjectName from tblSubjectDetails where CourseID=in\_Filter and isactive=1;

END$$

DELIMITER ;