Class: INFX 563
Project: Final Project - Demonstration/walk-through
Names:

Robert Marshall
Ram Ganesan
Anuj Joshi
Pierre Augustamar

Date: December 15, 2017

Report showing what classes a specific student is currently attending and the room/building that the classes are taken place

Artifact Name: CAMP.uspGetReportingStudentAndAssignedClasses
Object Type : Stored Procedure
Description : Reporting a specific student's schedule
Demo : USE SUMMER_CAMP DECLARE @UserId int SELECT @UserId = 1 EXEC CAMP.uspGetReportingStudentAndAssignedClasses @UserId   _(see table below)_

	FIRST_NAME	LAST_NAME	CLASS_NAME	ROOM_ID	BUILDING_ID	START_TIME	END_TIME	CLASS_DATE
1	Sam	Mae	Animals	1	100	09:00:00.0000000	12:00:00.0000000	2018-06-11
2	Sam	Mae	Plants	2	100	09:00:00.0000000	12:00:00.0000000	2018-06-12
3	Sam	Mae	Survival	3	200	09:00:00.0000000	12:00:00.0000000	2018-06-13
4	Sam	Mae	Canoe	4	200	09:00:00.0000000	12:00:00.0000000	2018-06-14
5	Sam	Mae	Shelter	5	300	09:00:00.0000000	12:00:00.0000000	2018-06-15
6	Sam	Mae	Basket Weaving	10	500	13:00:00.0000000	16:00:00.0000000	2018-06-15
7	Sam	Mae	First Aid	9	500	13:00:00.0000000	16:00:00.0000000	2018-06-14
8	Sam	Mae	Reptiles	8	400	13:00:00.0000000	16:00:00.0000000	2018-06-13
9	Sam	Mae	Trees	7	400	13:00:00.0000000	16:00:00.0000000	2018-06-12
10	Sam	Mae	Insects	6	300	13:00:00.0000000	16:00:00.0000000	2018-06-11

This report is one that most parents will execute to find out what classes/activities that their children will be having for each day during the camp session. Note that administrators and

instructors may also implement this report when trying to check on a specific student. However, it is expected that this report will be executed at a lesser rate by the camp's staff.

Report showing what classes each students in the camp are currently attending and the room/building that the classes are taken place

Artifact Name: CAMP.vwStudentsSchedules
Object Type : Views
Description : Reporting the schedule of all the students that are attending the Summer camp
Demo : USE SUMMER_CAMP SELECT * FROM CAMP.vwStudentsSchedules

FIRST_NAME	LAST_NAME	CLASS_NAME	ROOM_ID	BUILDING_ID	START_TIME	END_TIME	CLASS_DATE
Sam	Mae	Animals	1	100	09:00:00.0000000	12:00:00.0000000	2018-06-11
Sam	Mae	Plants	2	100	09:00:00.0000000	12:00:00.0000000	2018-06-12
Sam	Mae	Survival	3	200	09:00:00.0000000	12:00:00.0000000	2018-06-13
Sam	Mae	Canoe	4	200	09:00:00.0000000	12:00:00.0000000	2018-06-14
Sam	Mae	Shelter	5	300	09:00:00.0000000	12:00:00.0000000	2018-06-15
Jack	Paul	Insects	2	100	09:00:00.0000000	12:00:00.0000000	2018-06-11
Jack	Paul	Trees	3	200	09:00:00.0000000	12:00:00.0000000	2018-06-12
Jack	Paul	Animals	4	200	09:00:00.0000000	12:00:00.0000000	2018-06-13
Jack	Paul	First Aid	5	300	09:00:00.0000000	12:00:00.0000000	2018-06-14
Jack	Paul	Basket Weaving	6	300	09:00:00.0000000	12:00:00.0000000	2018-06-15
Matt	Joe	Plants	3	200	09:00:00.0000000	12:00:00.0000000	2018-06-11
Matt	Joe	Survival	4	200	09:00:00.0000000	12:00:00.0000000	2018-06-12
Matt	Joe	Insects	5	300	09:00:00.0000000	12:00:00.0000000	2018-06-13
Matt	Joe	Shelter	6	300	09:00:00.0000000	12:00:00.0000000	2018-06-14
Matt	Joe	Reptiles	7	400	09:00:00.0000000	12:00:00.0000000	2018-06-15
Anne	Anderson	Trees	4	200	09:00:00.0000000	12:00:00.0000000	2018-06-11
Anne	Anderson	Animals	5	300	09:00:00.0000000	12:00:00.0000000	2018-06-12
Anne	Anderson	Plants	6	300	09:00:00.0000000	12:00:00.0000000	2018-06-13
Anne	Anderson	Basket Weaving	7	400	09:00:00.0000000	12:00:00.0000000	2018-06-14

Administrators and instructors will only execute this report. Parents and other users in the system will not have the authorization to run such report. A critical scenario for this kind of report would be to ensure students are listed in the right camp activities or classes. Also, at the start of each camp, every member of the staff will be eligible to execute this report so that they are aware of the students that are expected to report to a specific camp. Note that this report will only show students that are stated to be active users meaning that their permissions forms, releases forms, and payment have been verified.

Report showing what classes each instructors in the camp are currently teaching during camp and the room/building that the classes are taken place

Artifact Name: CAMP.vwTeachersSchedules
Object Type : Views
Description : Reporting the schedule activities/classes for each instructors
Demo : SELECT * FROM CAMP.vwTeachersSchedules

FIRST_NAME	LAST_NAME	CLASS_NAME	ROOM_ID	BUILDING_ID	START_TIME	END_TIME	CLASS_DATE
Anuj	Joshi	Animals	1	100	09:00:00.0000000	12:00:00.0000000	2018-06-11
Betty	Pawn	Insects	2	100	09:00:00.0000000	12:00:00.0000000	2018-06-11
Chance	Checker	Plants	3	200	09:00:00.0000000	12:00:00.0000000	2018-06-11
Nate	Chess	Trees	4	200	09:00:00.0000000	12:00:00.0000000	2018-06-11
Maple	Ladder	Survival	5	300	09:00:00.0000000	12:00:00.0000000	2018-06-11
Maple	Ladder	Insects	6	300	13:00:00.0000000	16:00:00.0000000	2018-06-11
Nate	Chess	Plants	7	400	13:00:00.0000000	16:00:00.0000000	2018-06-11
Chance	Checker	Trees	8	400	13:00:00.0000000	16:00:00.0000000	2018-06-11
Betty	Pawn	Survival	9	500	13:00:00.0000000	16:00:00.0000000	2018-06-11
Anuj	Joshi	Animals	10	500	13:00:00.0000000	16:00:00.0000000	2018-06-11
Anuj	Joshi	Plants	2	100	09:00:00.0000000	12:00:00.0000000	2018-06-12
Betty	Pawn	Trees	3	200	09:00:00.0000000	12:00:00.0000000	2018-06-12
Chance	Checker	Survival	4	200	09:00:00.0000000	12:00:00.0000000	2018-06-12
Nate	Chess	Animals	5	300	09:00:00.0000000	12:00:00.0000000	2018-06-12
Maple	Ladder	Insects	6	300	09:00:00.0000000	12:00:00.0000000	2018-06-12
Maple	Ladder	Trees	7	400	13:00:00.0000000	16:00:00.0000000	2018-06-12
Nate	Chess	Survival	8	400	13:00:00.0000000	16:00:00.0000000	2018-06-12
Chance	Checker	Animals	9	500	13:00:00.0000000	16:00:00.0000000	2018-06-12
Betty	Pawn	Insects	10	500	13:00:00.0000000	16:00:00.0000000	2018-06-12
Anuj	Joshi	Plants	1	100	13:00:00.0000000	16:00:00.0000000	2018-06-12
Anuj	Joshi	Survival	3	200	09:00:00.0000000	12:00:00.0000000	2018-06-13
Betty	Pawn	Animals	4	200	09:00:00.0000000	12:00:00.0000000	2018-06-13

This report is available to anyone who is part of the staff. However, administrators will be the primary users to generate this report. It allows staff members to have a historical list of classes or activities taught during a specific camp. Thus, administrators can use that information to adjust or update as needed for the upcoming Summer camp or to make changes due to current staff skills and or because of an unscheduled time off.

Report showing what classes that a particular instructor is/has been teaching during camp.

Artifact Name: CAMP..uspGetReportingTeacherAndAssignedClasses
Object Type : Stored Procedure
Description : Reporting the schedule activities/classes for a specific instructor
Demo :   USE SUMMER_CAMP

```
DECLARE @StaffId int
SELECT @StaffId = 512
EXEC CAMP.uspGetReportingTeacherAndAssignedClasses @StaffId
```

FIRST_NAME	LAST_NAME	CLASS_NAME	ROOM_ID	BUILDING_ID	START_TIME	END_TIME	CLASS_DATE
Anuj	Joshi	Animals	1	100	09:00:00.0000000	12:00:00.0000000	2018-06-11
Anuj	Joshi	Animals	10	500	13:00:00.0000000	16:00:00.0000000	2018-06-11
Anuj	Joshi	Plants	2	100	09:00:00.0000000	12:00:00.0000000	2018-06-12
Anuj	Joshi	Plants	1	100	13:00:00.0000000	16:00:00.0000000	2018-06-12
Anuj	Joshi	Survival	3	200	09:00:00.0000000	12:00:00.0000000	2018-06-13
Anuj	Joshi	Basket Weaving	2	100	13:00:00.0000000	16:00:00.0000000	2018-06-13
Anuj	Joshi	Canoe	4	200	09:00:00.0000000	12:00:00.0000000	2018-06-14
Anuj	Joshi	Canoe	3	200	13:00:00.0000000	16:00:00.0000000	2018-06-14
Anuj	Joshi	Shelter	5	300	09:00:00.0000000	12:00:00.0000000	2018-06-15
Anuj	Joshi	Shelter	4	200	13:00:00.0000000	16:00:00.0000000	2018-06-15

This report allows an instructor to check his/her schedule during the time this person has been teaching as part of a staff member at the camp.

Trigger on update to protect from updating a camp's payment due date

Artifact Name: CAMP.STUDENT_PAYMENT_UPDATE
Object Type : Trigger
Description : A warning message is thrown if anyone is trying to change a camp's payment due date
Demo :

```
USE SUMMER_CAMP
UPDATE [SUMMER_CAMP].[CAMP].[MAP_STUDENT_PAYMENT]
  SET PAYMENT_DUE_DATE = CAST('2017-12-01' AS DATE)
WHERE CAMP_YEAR = '2018'


Updating the payment due date for a specific camp is not allowed
Msg 3609, Level 16, State 1, Line 120
The transaction ended in the trigger. The batch has been aborted.
```

This trigger protects the system from being updated for whatever reasons when it comes to the camp's payment due date. Administrators will set the original payment due date, and this date is final. This trigger also ensures that this payment due date is not comprised of anyone.

Trigger on delete to protect from removing any users from the camp database

Artifact Name: CAMP.USER_DELETE
Object Type : Trigger
Description : A warning message is thrown if anyone is trying to remove any users from the user table
Demo :

```
USE SUMMER_CAMP
DELETE FROM [SUMMER_CAMP].[CAMP].[MAP_USER_ROLE]
WHERE USER_ID = 45
GO
DELETE FROM [CAMP].[STAFF]
WHERE STAFF_ID = 45
GO
DELETE FROM [SUMMER_CAMP].[CAMP].[USER]
WHERE USER_ID = 45

Msg 50000, Level 16, State 1, Procedure USER_DELETE, Line 6 [Batch Start Line 149]
You cannot delete any users instead set the Is_ACTIVE flag to false for the related user
Msg 3609, Level 16, State 1, Line 150
The transaction ended in the trigger. The batch has been aborted.
```

The user table is meant to retain users info for life. Thus, this trigger is intended to ensure that no users will ever be deleted. In case that a user is no longer an active participant in the camp, then the best option is to set the is_active flag to false. This report will allow to retain users info and also reactive them as needed if they ever come back to work for the camp. Also, this is a security layer if anyone were to hack the system.

Function to check what students have more than one permissions

Artifact Name: CAMP.fx_StudentWithMultiPermissions
Object Type : Function
Description : returns the name of students who have more than one permissions
Demo :

```
select * from CAMP.fx_StudentWithMultiPermissions()
```

StudentName	Total
Sam Mae	5

Parents have a few permissions that they can check off that would allow their kids to take on some activities that are not being monitored by camp staff. For instance, a parent can let their kids go off the camp. Ideally, most kids would only have one permission because of the logistic. But there are occasions where a parent would be open to allow their kids to more than one permissions. This function will enable staff to one know which students have more than one permissions and also have a list in mind how many kids will need to check off for specific monitoring. Only administrators will be required to run this function.

Report showing all the active members in the summer camp

Artifact Name: CAMP.vwCurrentActiveUsers
Object Type : Views
Description : List out all active participants in the camp
Demo :   SELECT * FROM CAMP.vwCurrentActiveUsers

User name	ROLE_NAME
Sam Mae	student
Jack Paul	student
Matt Joe	student
Anne Anderson	student
Paul Smith	student
Frank Lynn	student
Hillary Clinton	student
Joe Coffee	student
Gladis Night	student
Clay Walker	student
Sal Mae	student
Jeff Mae	parent
Rand Paul	parent
Cool Joe	parent
Mike Anderson	parent
Will Smith	parent
Sue Lynn	parent
Bernie Trump	parent
Chuck Coffee	parent

This report allows administrators to view all the participants in the camp that are considered active participants.

Report showing all the active members in the summer camp

Artifact Name: CAMP.vwCurrentInactiveUsers
Object Type : Views
Description : List out all inactive participants in the camp
Demo : SELECT * FROM **CAMP.vwCurrentInactiveUsers**

User name	ROLE_NAME
Mort Knight	administrator
Walter Rook	work crews
Susan Castle	instructor
Noah Kelly	student
Emma Mason	student
Olivia Brock	student
Jacob Kent	student
Ava Lester	student
Ethan Jones	student
Mia Tea	student
Alex Light	student
Sophia Mills	student
Jackson Kelly	parent
Matt Mason	parent
Dan Brock	parent
Logan Kent	parent
Harper Lester	parent
Lucas Jones	parent
Avery Tea	parent
Ella Light	parent
Eva Mills	parent

This report allows administrators to view all the inactive participants that may at some point have been active but are now inactive.

Verify student's permissions - creating a stored proc to check student's permission and also check the duration for which the student is exempted

Artifact Name: CAMP.usp_getstudentpermissions
Object Type : Stored Procedure
Description : Provides student's permissions and it's start and end date
Demo : DECLARE @userId int SET @userId = 3 EXECUTE CAMP.[usp_getstudentpermissions] @userId   <table><tr><td></td><td>FIRST_NAME</td><td>LAST_NAME</td><td>PERMISSION_START_DATE</td><td>PERMISSION_END_DATE</td><td>PERMISSION_TYPE</td><td>PERMISSION_TYPE_DESC</td></tr><tr><td>1</td><td>Matt</td><td>Joe</td><td>2018-06-11</td><td>2018-06-15</td><td>Off Camp</td><td>permission to go off camp areas</td></tr></table>

This stored procedure will verify student's permissions, for example if a student has permission to go off camp areas, then in that case these permissions can be viewed using this stored procedure by respective students or the administration.

Report to show permission - creating a view for everyone to check student's permission

Artifact Name: CAMP.vw_getstudentpermissions
Object Type : View
Description : Provides student's permissions and it's start and end date
Demo : SELECT * FROM CAMP.[vw_getstudentpermissions]

	FIRST_NAME	LAST_NAME	PERMISSION_START_DATE	PERMISSION_END_DATE	PERMISSION_TYPE	PERMISSION_TYPE_DESC
1	Sam	Mae	2018-06-11	2018-06-13	Leave Early	permission to leave camp early
2	Sam	Mae	2018-06-11	2018-06-13	Drive	permission to drive to and from camp
3	Sam	Mae	2018-06-11	2018-06-15	Media Share	permission to share photos on social media
4	Sam	Mae	2018-06-11	2018-06-15	ALT Parents	permission to leave camp with other parents
5	Sam	Mae	2018-06-11	2018-06-15	Off Camp	permission to go off camp areas
6	Jack	Paul	2018-06-11	2018-06-15	Off Camp	permission to go off camp areas
7	Matt	Joe	2018-06-11	2018-06-15	Off Camp	permission to go off camp areas
8	Anne	Anderson	2018-06-11	2018-06-15	Off Camp	permission to go off camp areas
9	Paul	Smith	2018-06-11	2018-06-15	Off Camp	permission to go off camp areas

This view will display the student's permissions, for example if a student has permission to go off camp areas, then in that case these permissions can be viewed using this view.

Report to show release - creating a view and stored procedure to display release dates and exception categories

Artifact Name: CAMP.vw_getreleaseinformation
Object Type : View
Description : Provides release information of a student

Demo :

SELECT * FROM CAMP.[vw_getreleaseinformation]

	FIRST_NAME	LAST_NAME	RELEASE_START_DATE	RELEASE_END_DATE	RELEASE_TYPE	RELEASE_TYPE_DESC
1	Sam	Mae	2018-06-11	2018-06-15	Exception	exception to release
2	Jack	Paul	2018-06-11	2018-06-15	Consent	consent to release
3	Matt	Joe	2018-06-11	2018-06-15	Consent	consent to release
4	Anne	Anderson	2018-06-11	2018-06-15	Non-consent	non-consent to release
5	Paul	Smith	2018-06-11	2018-06-15	Consent	consent to release
6	Frank	Lynn	2018-06-11	2018-06-15	Consent	consent to release
7	Hillary	Clinton	2018-06-11	2018-06-15	Consent	consent to release
8	Joe	Coffee	2018-06-11	2018-06-15	Consent	consent to release
9	Gladis	Night	2018-06-11	2018-06-15	Consent	consent to release
10	Clay	Walker	2018-06-11	2018-06-15	Consent	consent to release
11	Sal	Mae	2018-06-11	2018-06-15	Exception	exception to release

This report will give us release information of students giving us details about release type and its description with release start and end date.

Artifact Name: CAMP.usp_getreleaseinformation
Object Type : Stored Procedure
Description : Provides release information of a student

Demo :

```
DECLARE @userId int
SET @userId = 4
EXECUTE CAMP.[usp_getreleaseinformation] @userId
```

	FIRST_NAME	LAST_NAME	RELEASE_START_DATE	RELEASE_END_DATE	RELEASE_TYPE	RELEASE_TYPE_DESC
1	Anne	Anderson	2018-06-11	2018-06-15	Non-consent	non-consent to release

The stored procedures is for students if they want to check their release information.

Report to show payment - create a view and stored procedure to track payment status of students

---

Artifact Name: CAMP.vw_getpaymentinformation

Object Type : View

Description : Provides payment information of students

Demo :

```
SELECT * FROM CAMP.[vw_getpaymentinformation]
```

	FIRST_NAME	LAST_NAME	PAYMENT_DATE	PAYMENT_DUE_DATE	PAYMENT_STATUS	CAMP_YEAR
1	Sam	Mae	2017-01-11	2017-12-01	paid	2018
2	Jack	Paul	2017-03-21	2017-12-01	paid	2018
3	Matt	Joe	2017-09-13	2017-12-01	paid	2018
4	Anne	Anderson	2017-07-18	2017-12-01	paid	2018
5	Paul	Smith	2017-02-19	2017-12-01	paid	2018
6	Frank	Lynn	2017-08-22	2017-12-01	paid	2018
7	Hillary	Clinton	2017-11-19	2017-12-01	paid	2018
8	Joe	Coffee	2017-06-09	2017-12-01	paid	2018
9	Gladis	Night	NULL	2017-12-01	not paid	2018
10	Clay	Walker	NULL	2017-12-01	not paid	2018
11	Sal	Mae	2017-07-01	2017-12-01	paid	2018

This view will give the payment information of students, giving details about payment due date, payment status, and payment date.

Artifact Name: CAMP.usp_getpaymentinformation

Object Type : Stored Procedure
Description : Provides payment information of a student
Demo : DECLARE @userId int SET @userId = 4 EXECUTE CAMP.[usp_getpaymentinformation] @userId 	FIRST_NAME	LAST_NAME	PAYMENT_DATE	PAYMENT_DUE_DATE	PAYMENT_STATUS	CAMP_YEAR
---	---	---	---	---	---
1 Anne	Anderson	2017-07-18	2017-12-01	paid	2018	

Report that parents can generate - creating a stored procedure to display all the required information a parent can access of their child

Artifact Name: CAMP.usp_GetChildDetails
Object Type : Stored Procedure
Description : Provides required information about students to their parents
Demo : DECLARE @userId int SET @userId = 22 EXECUTE CAMP.[usp_GetChildDetails] @userId 	FIRST_NAME	LAST_NAME	DATE_OF_BIRTH	PAYMENT_DATE	PAYMENT_DUE_DATE	PAYMENT_STATUS	CAMP_YEAR
---	---	---	---	---	---	---
1 Jack	Paul	2007-02-02	2017-11-21	2017-12-01	paid	2018	

With this stored procedure, parents can access their child's record using their parent id and will get information regarding payment due date and payment

Display weekly scheduled activities - This will create a report giving information about class schedule and associated rooms and building

Artifact Name: CAMP.vw_getweeklyschedule

	Object Type : View
	Description : Provides weekly scheduled of all the activities in the camp
	Demo : SELECT * FROM CAMP.[vw_getweeklyschedule]

	CLASS_NAME	ROOM_NAME	BUILDING_ID	SCHEDULE_ID	CLASS_DATE	START_TIME	END_TIME
1	Animals	Red A	100	51111	2017-06-12	09:00:00.0000000	12:00:00.0000000
2	Insects	Red B	100	51112	2017-06-12	09:00:00.0000000	12:00:00.0000000
3	Plants	White A	200	51113	2017-06-12	09:00:00.0000000	12:00:00.0000000
4	Trees	White B	200	51114	2017-06-12	09:00:00.0000000	12:00:00.0000000
5	Survival	Green A	300	51115	2017-06-12	09:00:00.0000000	12:00:00.0000000
6	Insects	Green B	300	51121	2017-06-12	13:00:00.0000000	16:00:00.0000000
7	Plants	Blue A	400	51122	2017-06-12	13:00:00.0000000	16:00:00.0000000
8	Trees	Blue B	400	51123	2017-06-12	13:00:00.0000000	16:00:00.0000000
9	Survival	Pink A	500	51124	2017-06-12	13:00:00.0000000	16:00:00.0000000
10	Animals	Pink B	500	51125	2017-06-12	13:00:00.0000000	16:00:00.0000000
11	Plants	Red B	100	51211	2017-06-13	09:00:00.0000000	12:00:00.0000000
12	Trees	White A	200	51212	2017-06-13	09:00:00.0000000	12:00:00.0000000
13	Survival	White B	200	51213	2017-06-13	09:00:00.0000000	12:00:00.0000000
14	Animals	Green A	300	51214	2017-06-13	09:00:00.0000000	12:00:00.0000000
15	Insects	Green B	300	51215	2017-06-13	09:00:00.0000000	12:00:00.0000000
16	Trees	Blue A	400	51221	2017-06-13	13:00:00.0000000	16:00:00.0000000
17	Survival	Blue B	400	51222	2017-06-13	13:00:00.0000000	16:00:00.0000000
18	Animals	Pink A	500	51223	2017-06-13	13:00:00.0000000	16:00:00.0000000
19	Insects	Pink B	500	51224	2017-06-13	13:00:00.0000000	16:00:00.0000000

The weekly schedule view gives a report about all the activities that are happening on camp with their building locations and timings as well.

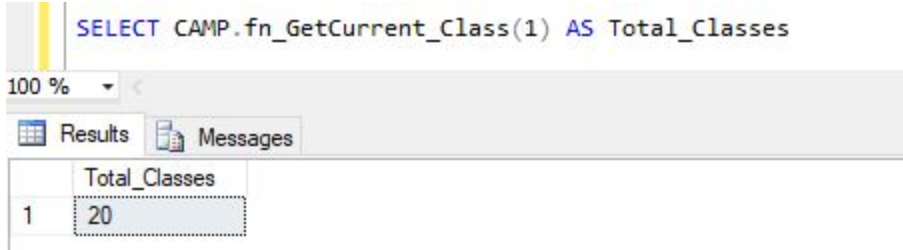| |
|---|
| **Artifact Name: CAMP.fn_GetCurrent_Class** |
| **Object Type :** Function |
| **Description :** |

This Function will list the total classes the student is enrolled/scheduled for.
| **Demo :**<br><br>SELECT CAMP.fn_GetCurrent_Class(1) AS Total_Classes<br><br>```
SELECT CAMP.fn_GetCurrent_Class(1) AS Total_Classes
```<br>100 %  ▾<br><br>▦ Results  📄 Messages<br><br>Total_Classes<br>1   20 |

Artifact Name: CAMP.fn_Check_Permission
Object Type : Function
Description : **This Function will enable users to verify a permission for a student.**
| **Demo :**<br><br>SELECT CAMP.fn_Check_Permission(1, 'drive') AS IsPermitted<br>SELECT CAMP.fn_Check_Permission(1, 'Leave Early') AS IsPermitted<br>SELECT CAMP.fn_Check_Permission(2, 'Leave Early') AS IsPermitted<br><br>```
SELECT CAMP.fn_Check_Permission(1, 'drive') AS IsPermitted
SELECT CAMP.fn_Check_Permission(1, 'Leave Early') AS IsPermitted
SELECT CAMP.fn_Check_Permission(2, 'Leave Early') AS IsPermitted
```<br>0 %  ▾<br><br>▤ Results  📄 Messages<br><br>IsPermitted<br>Yes<br><br>IsPermitted<br>No<br><br>IsPermitted<br>No |
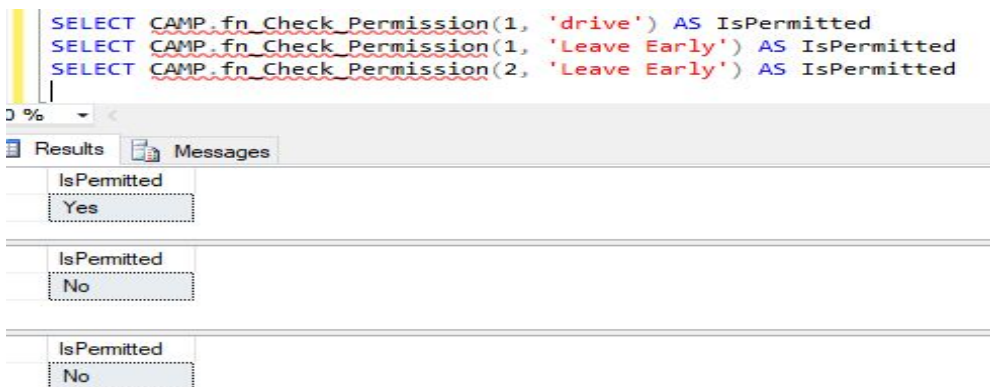
Artifact Name: CAMP.vwUserRoles
Object Type : View

Description : This is an all up View to view all users and the roles they are assigned to.

Demo :

SELECT * FROM CAMP.[vwUserRoles]

```
SELECT * FROM CAMP.[vwUserRoles]
```
100 %

Results | Messages

	USER_ID	FULL_NAME	IS_ADMIN	IS_INSTRUCTOR	IS_STUDENT	IS_PARENT	IS_WORKCREW	USER_STATUS
1	1	Sam Mae	No	No	Yes	No	No	Active
2	2	Jack Paul	No	No	Yes	No	No	Active
3	3	Matt Joe	No	No	Yes	No	No	Active
4	4	Anne Anderson	No	No	Yes	No	No	Active
5	5	Paul Smith	No	No	Yes	No	No	Active
6	6	Frank Lynn	No	No	Yes	No	No	Active
7	7	Hillary Clinton	No	No	Yes	No	No	Active
8	8	Joe Coffee	No	No	Yes	No	No	Active
9	9	Gladis Night	No	No	Yes	No	No	Active
10	10	Clay Walker	No	No	Yes	No	No	Active
11	11	Sal Mae	No	No	Yes	No	No	Active
12	21	Jeff Mae	No	No	No	Yes	No	Active
13	22	Rand Paul	No	No	No	Yes	No	Active
14	23	Cool Joe	No	No	No	Yes	No	Active
15	24	Mike Anderson	No	No	No	Yes	No	Active
16	25	Will Smith	No	No	No	Yes	No	Active
17	26	Sue Lynn	No	No	No	Yes	No	Active
18	27	Bernie Trump	No	No	No	Yes	No	Active
19	28	Chuck Coffee	No	No	No	Yes	No	Active
20	29	Fred Night	No	No	No	Yes	No	Active
21	30	Mary Walker	No	No	No	Yes	No	Active
22	41	Jefferson King	Yes	No	No	No	No	Active
23	42	Ray Queen	Yes	No	No	No	No	Active
24	43	Cory Bishop	Yes	No	No	No	No	Active
25	44	Mort Knight	Yes	No	No	No	No	InActive
26	45	Walter Rook	No	No	No	No	Yes	InActive
27	51	Susan Castle	No	Yes	No	No	No	InActive
28	52	Betty Pawn	No	Yes	No	No	No	Active
29	53	Chance Chec...	No	Yes	No	No	No	Active

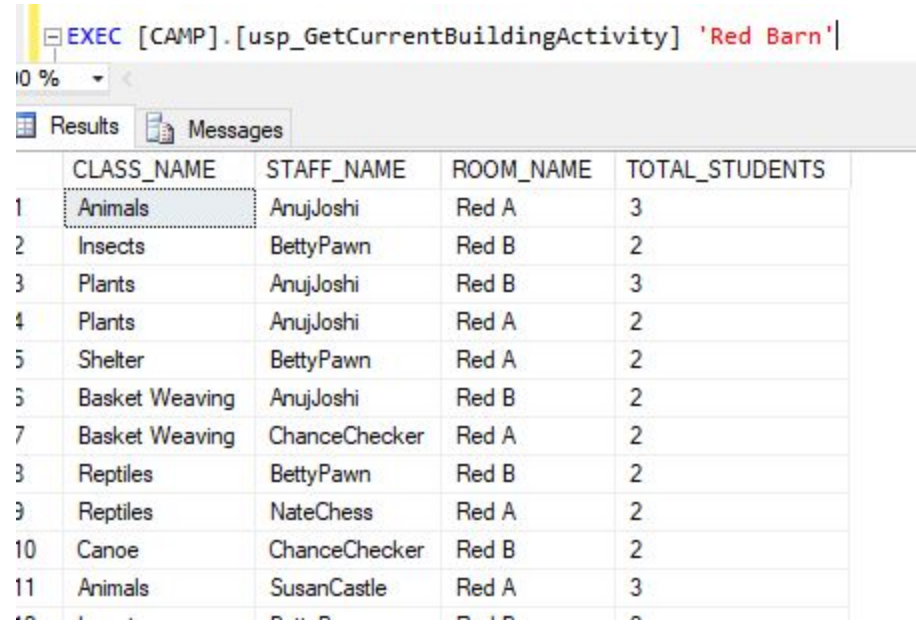**Artifact Name: CAMP.usp_GetCurrentBuildingActivity**

Object Type : Stored Procedure

Description : This procedure will fetch current list of activities fior a building, including

rooms, classes and total an estimated #students in the class.

Demo :

EXEC [CAMP].[usp_GetCurrentBuildingActivity] 'Red Barn'

```
EXEC [CAMP].[usp_GetCurrentBuildingActivity] 'Red Barn'
```
10 %

Results | Messages

	CLASS_NAME	STAFF_NAME	ROOM_NAME	TOTAL_STUDENTS
1	Animals	AnujJoshi	Red A	3
2	Insects	BettyPawn	Red B	2
3	Plants	AnujJoshi	Red B	3
4	Plants	AnujJoshi	Red A	2
5	Shelter	BettyPawn	Red A	2
6	Basket Weaving	AnujJoshi	Red B	2
7	Basket Weaving	ChanceChecker	Red A	2
8	Reptiles	BettyPawn	Red B	2
9	Reptiles	NateChess	Red A	2
10	Canoe	ChanceChecker	Red B	2
11	Animals	SusanCastle	Red A	3