Project Phase 2 (WithFriends)

Team Members:
Daniel Sha (danielsha94@gmail.com)
Kaile Ye (kaile.ye@stonybrook.edu)
Yida Yuan (yida.yuan@stonybrook.edu)

1. Users should be able to perform the following general transactions

- Register

INSERT INTO WithFriends_Users ($userdata) VALUES (…) ;
INSERT INTO WithFriends_Pages ($pagedata) VALUES (...);

```
1  INSERT INTO WithFriends_Users (UserId, FirstName, LastName, sex, EmailID, PWD, DOB, City, State,
2  ZipCode, Telephone, AccountNumber, AccountCreationDate, Preference)
3  VALUES (100031, "NewUser#1", "Phase2", "M", "newuser#1@cse305.stonybrook.edu",
4      "admin11", '1993-10-28', "New York", "NY",  10001,  111111111,  900031, '2016-11-18',
5      "Register a new user for project phase 2");
6  -- Create associate personal page
7  INSERT INTO WithFriends_Pages (PageId, OwnerId, GroupId, PostCount, PageType)
8  VALUES (300012, 100031, NULL, 0, 'User');
9  SELECT * FROM WithFriends_Users;
```

100% 1:9

Result Grid Filter Rows: Search Edit: Export/Import:

UserId	FirstName	LastName	sex	EmailID	PWD	DOB	Address	City	State	ZipCode	Telephone	Acc
100027	Empoyee...	Sys#7	M	sys7@cse305.stonybrook.edu	admin7	1993-10-28	NULL	New York	NY	10001	111111111	900
100028	Empoyee...	Sys#8	M	sys8@cse305.stonybrook.edu	admin8	1993-10-28	NULL	New York	NY	10001	111111111	900
100029	Empoyee...	Sys#9	M	sys9@cse305.stonybrook.edu	admin9	1993-10-28	NULL	New York	NY	10001	111111111	900
100030	Empoyee...	Sys#10	M	sys10@cse305.stonybrook.edu	admin10	1993-10-28	NULL	New York	NY	10001	111111111	900
100031	NewUser#1	Phase2	M	newuser#1@cse305.stonybrook.edu	admin11	1993-10-28	NULL	New York	NY	10001	111111111	900
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NUL

- Sign in and sign out

INSERT INTO WithFriends_Logins (LoginInId) VALUES ($LoginInId);
DELETE FROM WithFriends_Logins WHERE LoginInId = $LoginInId;

```
1  -- Sign in a user
2  INSERT INTO WithFriends_Logins (LoginInId)
3  VALUES (100031);
4  SELECT * FROM WithFriends_Logins;
5  -- Sign out a user
6  DELETE FROM WithFriends_Logins
7  WHERE LoginInId = 100031;
```

100% 1:2

Result Grid Filter Rows: Search Edit:

LoginInId
100031
NULL

- Post messages in their personal pages

INSERT INTO Posts (PosterId, PageId, Date, Content, CommentCount) VALUES (...)

```
INSERT INTO WithFriends_Posts (PostId, PosterId, PageId, PostDate, Content, CommentCount)
VALUES (400011, 100031, 300012, '2016-11-17', "Post a msg for project phase 2", 0);

SELECT * FROM WithFriends_Posts;
```

100% 33:4

Result Grid | Filter Rows: Search | Edit: | Export/Import:

PostId	PosterId	PageId	PostDate	Content	CommentCount
400007	100010	300009	2016-11-15	I'm Kylie, Does anyone know the requirements t...	0
400008	100001	300009	2016-11-15	Aced Google's interview today, really easy	0
400009	100013	300009	2016-11-15	What's the workload of CSE381 like? I'm really i...	0
400010	100004	300009	2016-11-15	My schedule for the next semester is : CSE 219...	0
400011	100031	300012	2016-11-17	Post a msg for project phase 2	0
NULL	NULL	NULL	NULL	NULL	NULL

WithFriends_Posts 2

- Send a message

INSERT INTO WithFriends_Messages (MessageId, SentDate, Subject, Content) VALUES (...);

-- Send this msg from user1 to user2

INSERT INTO WithFriends_MessagesSent (MessageId, SenderId, ReceiverId)

VALUES ($MsgId, $User1, $User2);

```
INSERT INTO WithFriends_Messages (MessageId, SentDate, Subject, Content)
VALUES (600012, '2016-11-17', 'Project phase2', 'new msg');
-- Send this msg from newly created user to user1 (Michael)
INSERT INTO WithFriends_MessagesSent (MessageId, SenderId, ReceiverId)
VALUES (600012, 100031, 100001);

SELECT * FROM WithFriends_Messages;
```

100% 36:7

Result Grid | Filter Rows: Search | Edit: | Export/Import:

MessageId	SentDate	Subject	Content
600008	2016-11-17	System Msg	Content#8
600009	2016-11-17	System Msg	Content#9
600010	2016-11-17	System Msg	Content#10
600011	2016-11-17	System Msg	Content#11
600012	2016-11-17	Project phase2	new msg
NULL	NULL	NULL	NULL

WithFriends_Messages 3

- Receive a message

SELECT * FROM Messages
INNER JOIN MessagesSend ON Messages.MessageId = MessagesSent.MessageId AND MessagesSent.ReceiverId = $userid AND MessagesSent.Deleted = 0

- Delete a message

DELETE FROM WithFriends_Messages
WHERE MessageId = $MessageId;

```
1  DELETE FROM WithFriends_Messages
2  WHERE MessageId = 600012;
3
4  SELECT * FROM WithFriends_Messages;
```

100% 1:3

Result Grid | Filter Rows: Search | Edit: | Exp

MessageId	SentDate	Subject	Content
600007	2016-11-17	System Msg	Content#7
600008	2016-11-17	System Msg	Content#8
600009	2016-11-17	System Msg	Content#9
600010	2016-11-17	System Msg	Content#10
600011	2016-11-17	System Msg	Content#11
NULL	NULL	NULL	NULL

Users should be able to perform the following transactions with regard to their own groups

- Create a group

INSERT INTO Groups (GroupName, Type, OwnerId) VALUES (...)
INSERT INTO GroupsMembers (GroupId, UserId) VALUES (...)
INSERT INTO Pages (GroupId, PageType) VALUES (...)

```
1  INSERT INTO WithFriends_Groups (GroupId, GroupName, Type, OwnerId)
2  VALUES (200011, "Sys group", "SYS", 100021);
3  -- The owner should be inserted as a group member.
4  INSERT INTO WithFriends_GroupsMembers (GroupId, UserId)
5  VALUES(200011 ,100021);
6  -- Create associated group page
7  INSERT INTO WithFriends_Pages (PageId, OwnerId, GroupId, PostCount, PageType)
8  VALUES (300011, NULL, 200011, 0, "Group");
9
10 SELECT * FROM WithFriends_Groups;
```

100% 43:8

Result Grid | Filter Rows: Search | Edit: | Export/Import:

GroupId	GroupName	Type	OwnerId
200003	StonyBrookGang	Friends	100003
200004	CS Folks	Friends	100004
200005	My Family	Friends	100005
200006	Microsoft Groupies	Friends	100006
200007	WOWers	Game	100012
200008	Apartment Union	Neighbors	100013
200009	Agents	Work	100014
200010	Google Interns	Interns	100015
200011	Sys group	SYS	100021

WithFriends_Groups 4

- Search for a user and add him to a group

SELECT UserId FROM Users WHERE (<different search options, ex: name, email, etc>)
INSERT INTO GroupsMembers (GroupId, UserId) VALUES (...)

- Make a post

INSERT INTO WithFriends_Posts (...)
VALUES (...);
UPDATE WithFriends_Pages
SET PostCount = $PostCount + 1
WHERE PageId = $PageId;

```
1   -- User 100005 (Mackenzie Wells) make a new post
2   -- (PostId=400011) on Group page (300009) CS Folks
3   INSERT INTO WithFriends_Posts (PostId, PosterId, PageId, PostDate, Content, CommentCount)
4   VALUES (400011, 100005, 300009, '2016-11-18', "Hi CS folks, I'm taking 305", 0);
5
6   UPDATE WithFriends_Pages
7   SET PostCount = PostCount + 1
8   WHERE PageId = 300009;
9
10  SELECT * FROM WithFriends_Posts;
```

100% 32:10

Result Grid | Filter Rows: Search | Edit: | Export/Import:

PostId	PosterId	PageId	PostDate	Content	CommentCount
400001	100001	300006	2016-11-15	Hey guys, I'm Michael Collins. The number one…	0
400002	100001	300006	2016-11-15	This is the second post, just want to make sure i…	0
400003	100002	300006	2016-11-15	This is Aria, I think CSE 320 is hard. Can someb…	0
400004	100003	300006	2016-11-15	Does anyboby selling a laptop?	0
400005	100004	300004	2016-11-15	Hey, I'm Ellie, just registered for WithFriends an…	0
400006	100004	300004	2016-11-15	I only got 50 on the midterm, OMG, I'm failing	0
400007	100010	300009	2016-11-15	I'm Kylie, Does anyone know the requirements t…	0
400008	100001	300009	2016-11-15	Aced Google's interview today, really easy	0
400009	100013	300009	2016-11-15	What's the workload of CSE381 like? I'm really i…	0
400010	100004	300009	2016-11-15	My schedule for the next semester is : CSE 219…	0
400011	100005	300009	2016-11-18	Hi CS folks, I'm taking 305	0
NULL	NULL	NULL	NULL	NULL	NULL

WithFriends_Posts 5

Note: When you make a new post, the post count of the page has to increase 1.

- Comment on a post

INSERT INTO WithFriends_Comments (...) VALUES (...);
UPDATE WithFriends_Posts
SET CommentCount = $CommentCount + 1
WHERE PostId = $PostId;

```
1  -- User 100001 make a new comment(500011) on post 400011
2  INSERT INTO WithFriends_Comments (CommentId, PosterId, PostId, PostDate, Content)
3  VALUES (500011, 100001, 400011, '2016-11-18', "I'm also taking 305");
4
5  UPDATE WithFriends_Posts
6  SET CommentCount = CommentCount + 1
7  WHERE PostId = 400011;
8
9  SELECT * FROM WithFriends_Comments;
```

100% 35:9

Result Grid | Filter Rows: Search | Edit: | Export/Import:

CommentId	PosterId	PostId	PostDate	Content
500001	100002	400001	2016-11-16	The rule is ridiculous Michael!!
500002	100001	400003	2016-11-16	Switch to ISE and you will feel better.
500003	100002	400004	2016-11-16	I'm selling my 2014 Macbook Pro for $1000. Me…
500004	100002	400005	2016-11-16	Hi, I'm Aria, It's so nice to meet you on WithFrie…
500005	100001	400006	2016-11-16	Don't worry too much, the average is like 38. Yo…
500006	100001	400007	2016-11-16	You will have to pass both 114 and 215 with C o…
500007	100002	400008	2016-11-16	You are really good. What kind of questions you…
500008	100001	400008	2016-11-16	Well, it's basically just coding stuff with data stru…
500009	100001	400009	2016-11-16	CSE 381 is hard and heavliy use C++.
500010	100002	400010	2016-11-16	You will have no time to sleep if you do this.
500011	100001	400011	2016-11-18	I'm also taking 305
NULL	NULL	NULL	NULL	NULL

Note: when you add a new comment on a post, the comment count has to increase 1.

- Like a post

INSERT INTO WithFriends_LikePost (UserId, PostId)
VALUES ($UserId, $PostId);

```
1  -- User (Levi West 100008) likes Post (400001)
2  INSERT INTO  WithFriends_LikePost (UserId, PostId)
3  VALUES (100008, 400001);
4
5  SELECT * FROM WithFriends_LikePost;
```

100% 1:5

Result Grid | Filter Rows: Search | Edit: | Export/In

UserId	PostId
100001	400001
100002	400001
100004	400001
100008	400001
100009	400001
100010	400001
100003	400002
100008	400002
100005	400003
100007	400003
100006	400004
NULL	NULL

WithFriends_LikePost 1

- Like a comment

INSERT INTO WithFriends_LikeComment (UserId, CommentId)
VALUES ($UserId, $CommentId);

```
1  -- User (Victoria Powell 100015) likes Comment (500002)
2  INSERT INTO WithFriends_LikeComment (UserId, CommentId)
3  VALUES (100015, 500002);
4
5  SELECT * FROM WithFriends_LikeComment;
```

100% 38:5

Result Grid | Filter Rows: Search | Edit: | Export/Im

UserId	CommentId
100002	500001
100003	500001
100004	500001
100006	500001
100008	500001
100005	500002
100007	500002
100015	500002
100001	500004
100009	500005
100010	500007
NULL	NULL

WithFriends_LikeComment 3

- Remove a user from a group

DELETE FROM WithFriends_GroupsMembers
WHERE GroupId = $group AND UserId = $removedUserId

```
1  INSERT INTO WithFriends_GroupsMembers (GroupId, UserId)
2  VALUES(200005 ,100018);
3  -- DELETE
4  DELETE FROM WithFriends_GroupsMembers
5  WHERE GroupId = 200005 AND UserId = 100018;
6
7  SELECT * FROM WithFriends_GroupsMembers;
```

100% 35:5

Result Grid | Filter Rows: Search | Edit: | Export/Impo

GroupId	UserId
200001	100001
200004	100001
200001	100002
200001	100003
200003	100003
200004	100004
200003	100005
200005	100005
200001	100006
200004	100010
200004	100013
200005	100017
200005	100020

WithFriends_GroupsMembers 2

- Remove a post

DELETE FROM WithFriends_Posts
WHERE PostId=400011;
UPDATE WithFriends_Pages
SET PostCount = PostCount - 1
WHERE PageId = 300004;

```
INSERT INTO WithFriends_Posts (PostId, PosterId, PageId, PostDate, Content, CommentCount)
VALUES (400011, 100004, 300004, '2016-11-15', "new post for delete", 0);
UPDATE WithFriends_Pages
SET PostCount = PostCount + 1
WHERE PageId = 300004;
-- Delete
DELETE FROM WithFriends_Posts
WHERE PostId=400011;
UPDATE WithFriends_Pages
SET PostCount = PostCount - 1
WHERE PageId = 300004;
SELECT * FROM WithFriends_Posts;
```

100% 10:6

Result Grid | Filter Rows: Search | Edit: | Export/Import:

PostId	PosterId	PageId	PostDate	Content	CommentCount
400001	100001	300006	2016-11-15	Hey guys, I'm Michael Collins. The number one…	0
400002	100001	300006	2016-11-15	This is the second post, just want to make sure i…	0
400003	100002	300006	2016-11-15	This is Aria, I think CSE 320 is hard. Can someb…	0
400004	100003	300006	2016-11-15	Does anyboby selling a laptop?	0
400005	100004	300004	2016-11-15	Hey, I'm Ellie, just registered for WithFriends an…	0
400006	100004	300004	2016-11-15	I only got 50 on the midterm, OMG, I'm failing	0
400007	100010	300009	2016-11-15	I'm Kylie, Does anyone know the requirements t…	0
400008	100001	300009	2016-11-15	Aced Google's interview today, really easy	0
400009	100013	300009	2016-11-15	What's the workload of CSE381 like? I'm really i…	0
400010	100004	300009	2016-11-15	My schedule for the next semester is : CSE 219…	0
NULL	NULL	NULL	NULL	NULL	NULL

Note: Delete a post will automatically delete all the comments under this post since "ON DELETE CASCADE" is enforced.

- Remove a comment

DELETE FROM WithFriends_Comments
WHERE CommentId = $CommentId;
UPDATE WithFriends_Posts
SET CommentCount = $CommentCount - 1
WHERE PostId = $PostId;

```
-- Remove comment 500011 on Post 400011
DELETE FROM WithFriends_Comments
WHERE CommentId = 500011;

UPDATE WithFriends_Posts
SET CommentCount = CommentCount - 1
WHERE PostId = 400011;

SELECT * FROM WithFriends_Comments;
```

100% 33:2

Result Grid | Filter Rows: Search | Edit: | Expo

CommentId	PosterId	PostId	PostDate	Content
500001	100002	400001	2016-11-16	The rule is ridiculous Michael!!
500002	100001	400003	2016-11-16	Switch to ISE and you will feel better.
500003	100002	400004	2016-11-16	I'm selling my 2014 Macbook Pro for $1000. Me…
500004	100002	400005	2016-11-16	Hi, I'm Aria, It's so nice to meet you on WithFrie…
500005	100001	400006	2016-11-16	Don't worry too much, the average is like 38. Yo…
500006	100001	400007	2016-11-16	You will have to pass both 114 and 215 with C o…
500007	100002	400008	2016-11-16	You are really good. What kind of questions you…
500008	100001	400008	2016-11-16	Well, it's basically just coding stuff with data stru…
500009	100001	400009	2016-11-16	CSE 381 is hard and heavliy use C++.
500010	100002	400010	2016-11-16	You will have no time to sleep if you do this.

Note: When remove a comment from a post, then the comment count has to decrease by 1.

- Unlike a post

DELETE FROM WithFriends_LikePost
WHERE UserId=$UserId AND PostId=$PostId;

```
1  -- User (Levi West 100008) unlikes Post (400001)
2  DELETE FROM WithFriends_LikePost
3  WHERE UserId=100008 AND PostId=400001;
4
5  SELECT * FROM WithFriends_LikePost;
```

100% 36:5

Result Grid | Filter Rows: Search | Edit:

UserId	PostId
100001	400001
100002	400001
100004	400001
100009	400001
100010	400001
100003	400002
100008	400002
100005	400003
100007	400003
100006	400004
NULL	NULL

WithFriends_LikePost 2

- Unlike a comment

DELETE FROM WithFriends_LikeComment
WHERE UserId=$UserId AND CommentId=$CommentId;

```
1  -- User (Victoria Powell 100015) unlikes Comment (500002)
2  DELETE FROM WithFriends_LikeComment
3  WHERE UserId=100015 AND CommentId=500002;
4
5  SELECT * FROM WithFriends_LikeComment;
```

100% 34:3

Result Grid | Filter Rows: Search | Edit: | Export/Impo

UserId	CommentId
100002	500001
100003	500001
100004	500001
100006	500001
100008	500001
100005	500002
100007	500002
100001	500004
100009	500005
100010	500007
NULL	NULL

WithFriends_LikeComment 4

- Modify a post

UPDATE WithFriends_Posts SET Content = $newcontent WHERE Posts = $PostId

- Modify a comment

UPDATE WithFriends_Comments SET Content = $newcontent WHERE Comment = $comment

- Delete a group

DELETE FROM WithFriends_Groups
WHERE GroupId = $GroupId;

```
1  INSERT INTO WithFriends_Groups (GroupId, GroupName, Type, OwnerId)
2  VALUES (200011, "Sys group", "SYS", 100021);
3  INSERT INTO WithFriends_GroupsMembers (GroupId, UserId)
4  VALUES(200011 ,100021);
5  INSERT INTO WithFriends_GroupsMembers (GroupId, UserId)
6  VALUES(200011 ,100022);
7  INSERT INTO WithFriends_GroupsMembers (GroupId, UserId)
8  VALUES(200011 ,100023);
9
10 DELETE FROM WithFriends_Groups
11 WHERE GroupId = 200011;
12 SELECT * FROM WithFriends_Groups;
```

100% 1:12

Result Grid | Filter Rows: Search | Edit: | Export/Import:

GroupId	GroupName	Type	OwnerId
200001	My Friends	Friends	100001
200002	Best Friends	Friends	100002
200003	StonyBrookGang	Friends	100003
200004	CS Folks	Friends	100004
200005	My Family	Friends	100005
200006	Microsoft Groupies	Friends	100006
200007	WOWers	Game	100012
200008	Apartment Union	Neighbors	100013
200009	Agents	Work	100014
200010	Google Interns	Interns	100015

WithFriends_Groups 3

Note: If we delete a group, all group members from that group will also be deleted from GroupsMembers table.

- Rename a group

UPDATE WithFriends_Groups
SET GroupName = $newname
WHERE GroupId = $GroupId

Users should also be able to perform the following transactions with regard to other users' groups (Some of the transactions are duplicate from the above User-level actions, so new screenshots may not be included)

- Join a group

```
INSERT INTO WithFriends_GroupsMembers (GroupId, UserId)
Values ($GroupId, $UserId);
```

- Unjoin a group

```
DELETE FROM WithFriends_GroupsMembers
WHERE GroupId=$GroupId AND UserId=$UserId;
```

- Make a post on a group page

```
INSERT INTO WithFriends_Posts (...) VALUES (...);
UPDATE WithFriends_Pages
SET PostCount = $PostCount + 1
WHERE PageId = $PageId;
```

- Comment on a post on a group page

```
INSERT INTO WithFriends_Comments (...) VALUES (...);
UPDATE WithFriends_Posts
SET CommentCount = $CommentCount + 1
WHERE PostId = $PostId;
```

- Like a post on a group page

```
INSERT INTO LikePost (UserId, PostId) Values ($UserId, $PostId);
```

- Like a comment on a group page

```
INSERT INTO LikeComment (UserId, CommentId) Values ($UserId, $CommentId);
```

- Remove one of their posts on a group page

```
DELETE FROM Posts WHERE PostId=$PostId
```

- Remove a comment

```
DELETE FROM Comments WHERE CommentId=$CommentId
```

- Unlike a post

```
DELETE FROM LikePost WHERE UserId=$UserId, PostId=$PostId;
```

- Unlike a comment
DELETE FROM LikeComment WHERE UserId=$UserId, CommentId=$CommentId;

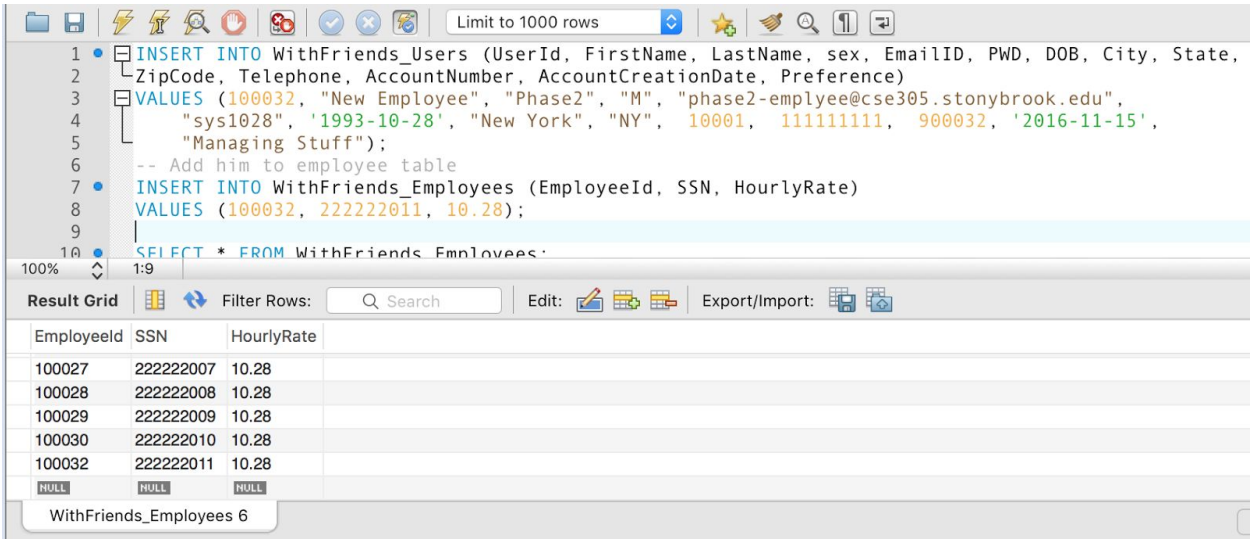
- Modify a post
UPDATE Posts SET Content=$NewContent WHERE PostId=$PostId;

- Modify a comment
UPDATE Posts SET Content=$NewContent WHERE CommentId=$CommentId;

Manager-Level Transactions

- Add, Edit and Delete information for an employee
Note: Since employee is a special kind of user in our database (EmployeeID is a foreign key of UserId), If we need to add an employee, we need to first add him as a user, then add him to the Employee table.

Add an emplyee:

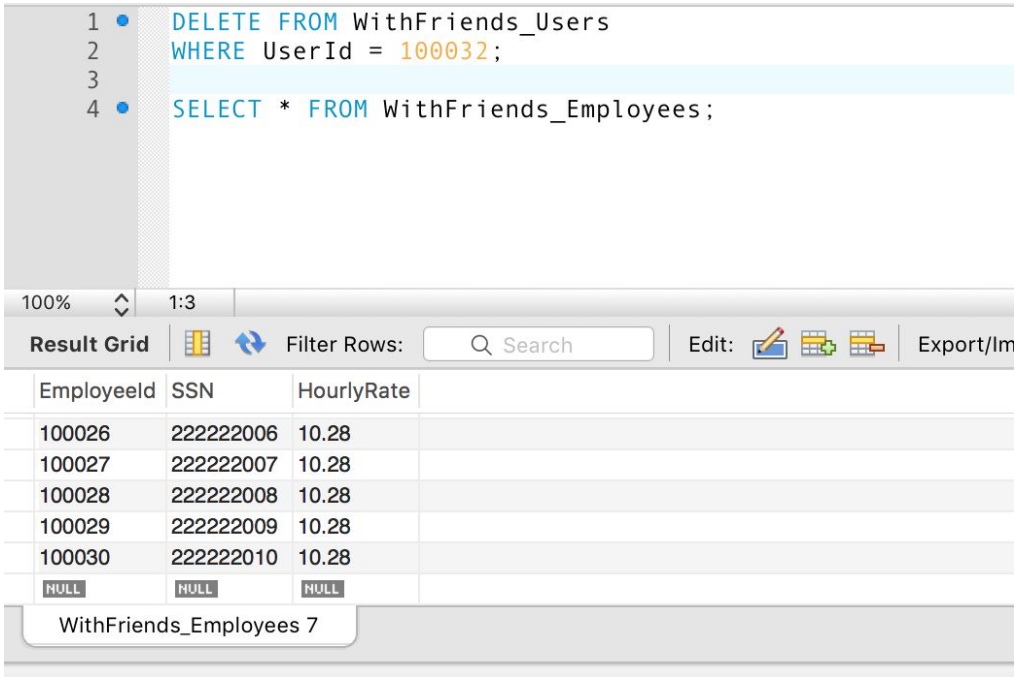


Edit an employee info:

1. For info in User table columns:
   UPDATE WithFriends_Users SET info=$newInfo WHERE UserId=$EmployeeId
2. For info in Employee table columns:
   UPDATE WithFriends_Employees SET info=$newInfo WHERE EmployeeId=$EmployeeId

Delete an employee:

DELETE FROM WithFriends_Users
WHERE UserId = $EmployeeId



Note: Since "ON DELETE CASCADE" and "ON UPDATE CASCADE" are enforced on EmployeeId. Any changes in the WithFriends_Users will automatically affect WithFriends_Employees table.

- Obtain a sales report for a particular month

CREATE VIEW SalesReport (TransactionId, ItemName, UnitPrice, units) AS
SELECT DISTINCT S.TransactionId, A.ItemName, A.UnitPrice, S.Units
FROM WithFriends_Sales S, WithFriends_Advertisements A,
(SELECT EXTRACT(month FROM SalesDate) AS SaleMonth FROM WithFriends_Sales) M
WHERE S.ADId = A.ADId
AND M.SaleMonth = $month;

```
1  CREATE VIEW SalesReport (TransactionId, ItemName, UnitPrice, units) AS
2  SELECT DISTINCT S.TransactionId, A.ItemName, A.UnitPrice, S.Units
3  FROM WithFriends_Sales S, WithFriends_Advertisements A,
4      (SELECT EXTRACT(month FROM SalesDate) AS SaleMonth FROM WithFriends_Sales) M
5  WHERE S.ADId = A.ADId
6      AND M.SaleMonth = 11;
7
8  SELECT * FROM WithFriends.salesreport;
```

100% 1:8

Result Grid | Filter Rows: Search | Export:

TransactionId	ItemName	UnitPrice	units
110001	Coat#1	100	10
110002	Coat#2	150	12
110003	Blackboard	300	15
110004	Coat#3	900	40
110005	Sneaker#1	150	30
110006	MacBook Pro	1999	60
110007	Surface Pro	1899	110
110008	Large Notebook	3	13
110009	Handbag#1	3000	53
110010	Handbag#2	500	12

- Produce a comprehensive listing of all items being advertised on the site

CREATE VIEW ALL_Items (Company, ItemName) AS
SELECT DISTINCT A.Company, A.ItemName
FROM WithFriends_Advertisements A;

```
1  CREATE VIEW ALL_Items (Company, ItemName) AS
2  SELECT DISTINCT A.Company, A.ItemName
3  FROM WithFriends_Advertisements A;
4
5  SELECT * FROM ALL_Items;
```

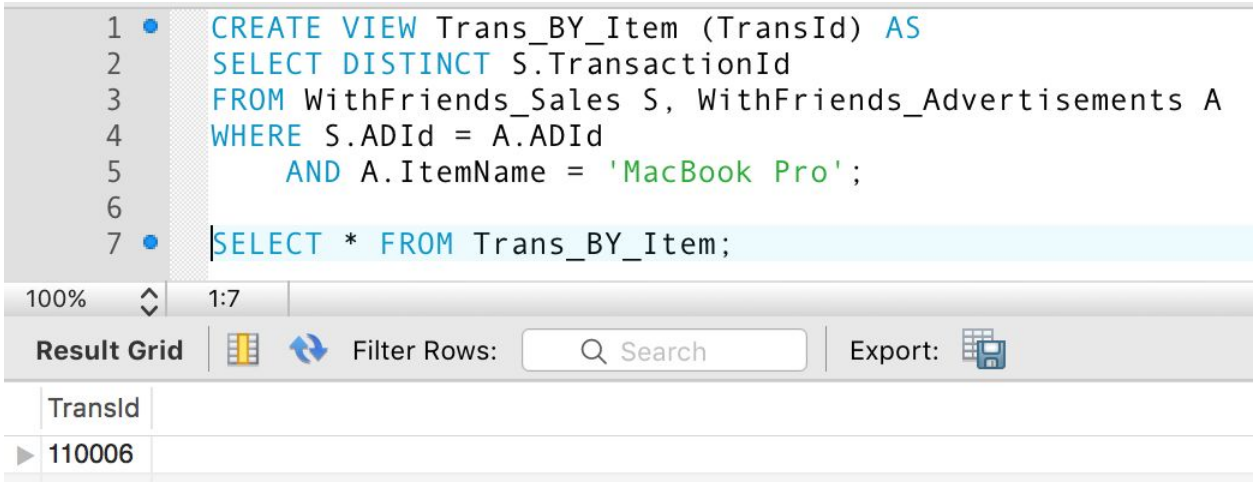
100% 25:5

Result Grid | Filter Rows: Search | Export:

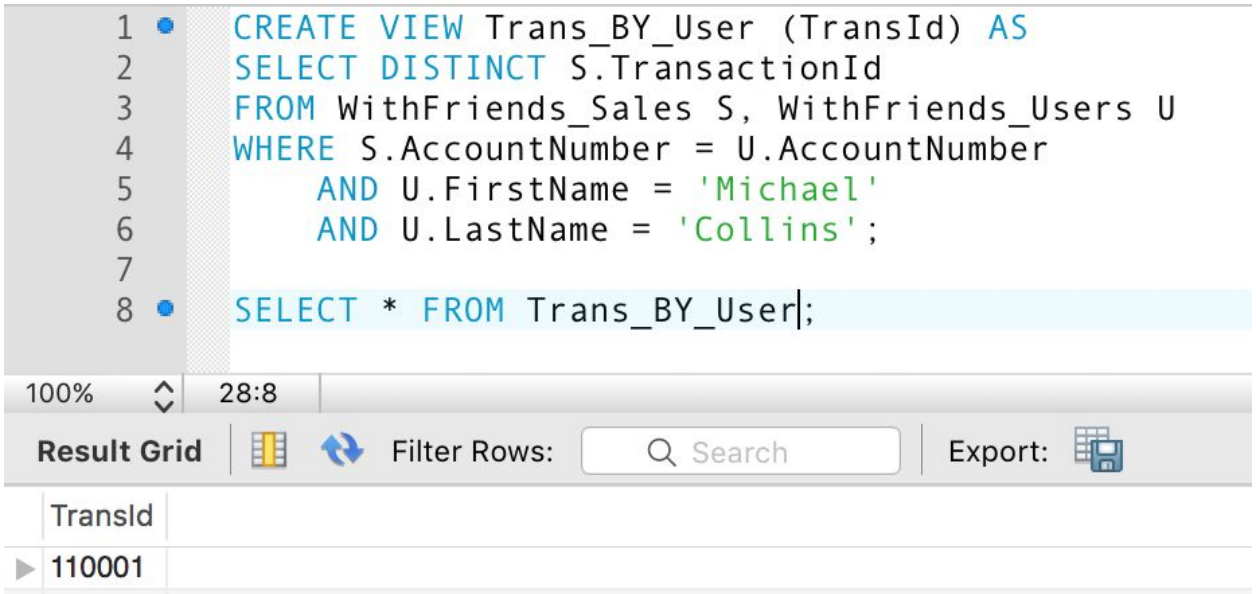
Company	ItemName
H&M	Coat#1
A&F	Coat#2
Blackboard Inc.	Blackboard
Canada Goose	Coat#3
Nike	Sneaker#1
Apple	MacBook Pro
Microsoft	Surface Pro
Staple	Large Notebook
Dior	Handbag#1
Michael Kors	Handbag#2

- Produce a list of transactions by item name or by user name

```
CREATE VIEW Trans_BY_Item (TransId) AS
SELECT DISTINCT S.TransactionId
FROM WithFriends_Sales S, WithFriends_Advertisements A
WHERE S.ADId = A.ADId
        AND A.ItemName = $ItemName;
```




```
CREATE VIEW Trans_BY_User (TransId) AS
SELECT DISTINCT S.TransactionId
FROM WithFriends_Sales S, WithFriends_Users U
WHERE S.AccountNumber = U.AccountNumber
        AND U.FirstName = $FirstName
   AND U.LastName = $LastName;
```



- Produce a summary listing of revenue generated by a particular item, item type, or customer

CREATE VIEW Revenue_BY_Item (ItemName, Revenue) AS
SELECT DISTINCT A.ItemName, S.Units*A.UnitPrice
FROM WithFriends_Sales S, WithFriends_Advertisements A
WHERE S.ADId = A.ADId
    AND A.ItemName = $ItemName;

```
1  CREATE VIEW Revenue_BY_Item (ItemName, Revenue) AS
2  SELECT DISTINCT A.ItemName, S.Units*A.UnitPrice
3  FROM WithFriends_Sales S, WithFriends_Advertisements A
4  WHERE S.ADId = A.ADId
5      AND A.ItemName = 'Surface Pro';
6
7  SELECT * FROM Revenue_BY_Item;
```

100% 1:7

Result Grid | Filter Rows: Search | Export:

ItemName	Revenue
Surface Pro	208890

CREATE VIEW Revenue_BY_ItemType (ItemName, Revenue) AS
SELECT A.Type, S.Units*A.UnitPrice
FROM WithFriends_Sales S, WithFriends_Advertisements A
WHERE S.ADId = A.ADId
    AND A.Type = $Type;
SELECT ItemName, SUM(Revenue) AS Total FROM Revenue_BY_ItemType;

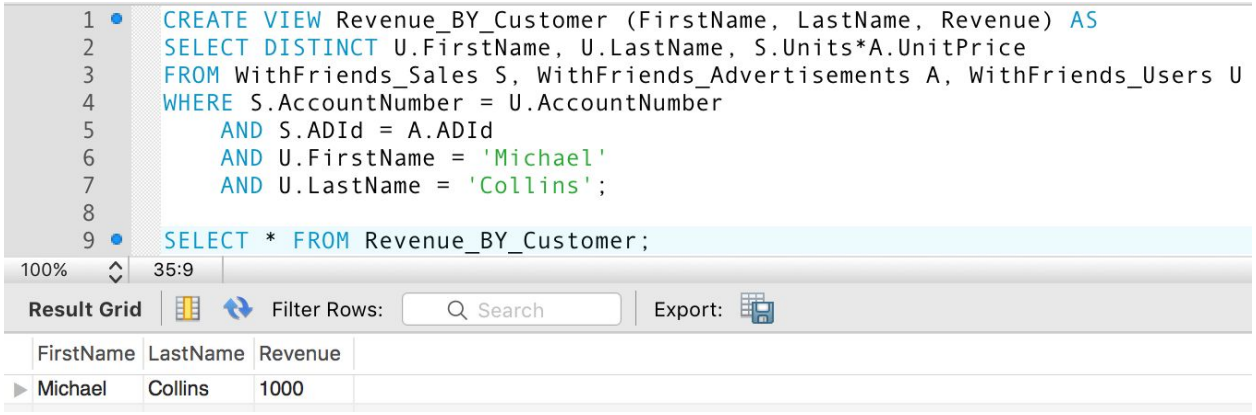
```
1  CREATE VIEW Revenue_BY_ItemType (ItemName, Revenue) AS
2  SELECT A.Type, S.Units*A.UnitPrice
3  FROM WithFriends_Sales S, WithFriends_Advertisements A
4  WHERE S.ADId = A.ADId
5      AND A.Type = 'Commercial';
6
7  SELECT ItemName, SUM(Revenue) AS Total FROM Revenue_BY_ItemType;
```

100% 1:7

Result Grid | Filter Rows: Search | Export:

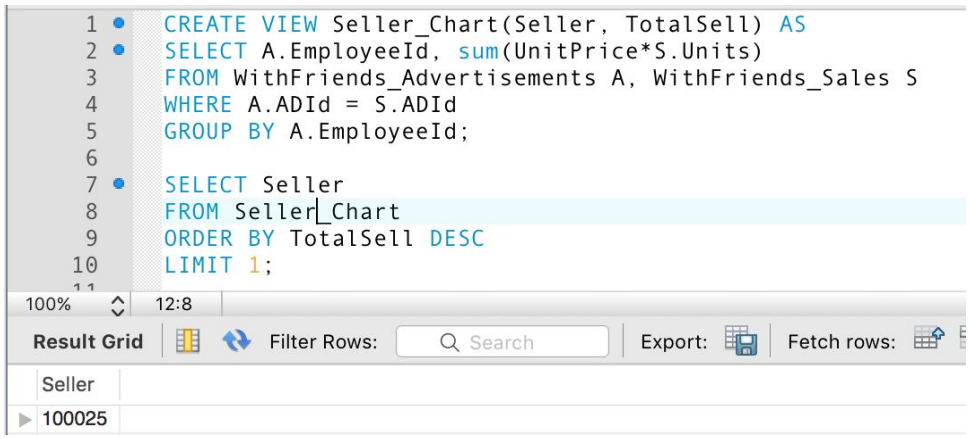
ItemName	Total
Commercial	532630

```
CREATE VIEW Revenue_BY_Customer (FirstName, LastName, Revenue) AS
SELECT DISTINCT U.FirstName, U.LastName, S.Units*A.UnitPrice
FROM WithFriends_Sales S, WithFriends_Advertisements A, WithFriends_Users U
WHERE S.AccountNumber = U.AccountNumber
        AND S.ADId = A.ADId
   AND U.FirstName = '$FirstName'
   AND U.LastName = '$LastName';
```



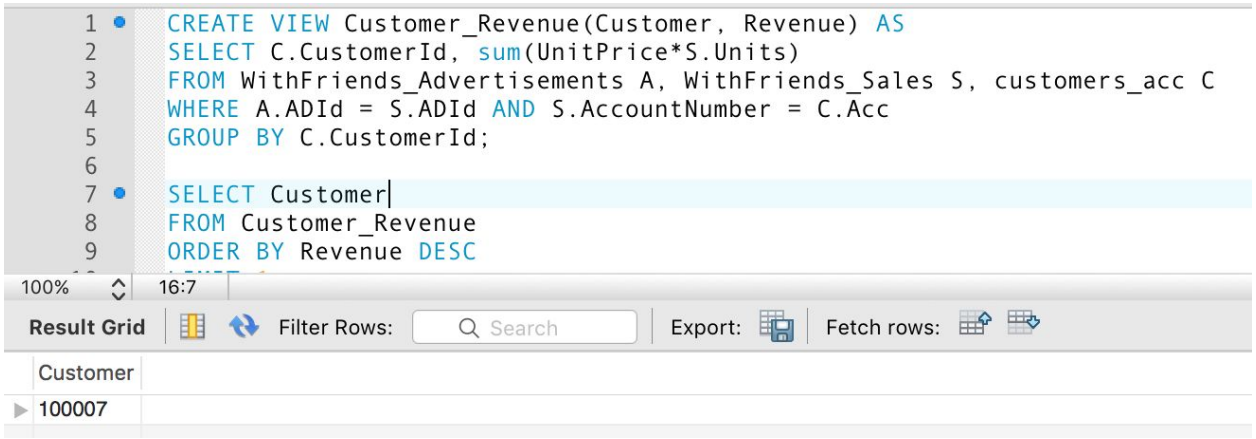
- Determine which customer representative generated most total revenue

```
CREATE VIEW Seller_Chart(Seller, TotalSell) AS
SELECT A.EmployeeId, sum(UnitPrice*S.Units)
FROM WithFriends_Advertisements A, WithFriends_Sales S
WHERE A.ADId = S.ADId
GROUP BY A.EmployeeId;

SELECT Seller
FROM Seller_Chart
ORDER BY TotalSell DESC
LIMIT 1;
```



- Determine which customer generated most total revenue

```
CREATE VIEW Customer_Revenue(Customer, Revenue) AS
SELECT C.CustomerId, sum(UnitPrice*S.Units)
FROM WithFriends_Advertisements A, WithFriends_Sales S, customers_acc C
WHERE A.ADId = S.ADId AND S.AccountNumber = C.Acc
GROUP BY C.CustomerId;

SELECT Customer
FROM Customer_Revenue
ORDER BY Revenue DESC
LIMIT 1;
```



- Produce a list of most active items

```
SELECT A.ItemName, sum(S.Units) AS SalesVolume
FROM WithFriends_Sales S, WithFriends_Advertisements A
WHERE S.ADId = A.ADId
GROUP BY A.ItemName
ORDER BY SalesVolume DESC;
```

```
1  SELECT A.ItemName, sum(S.Units) AS SalesVolume
2  FROM WithFriends_Sales S, WithFriends_Advertisements A
3  WHERE S.ADId = A.ADId
4  GROUP BY A.ItemName
5  ORDER BY SalesVolume DESC;
```

100% 15:5

Result Grid | Filter Rows: Search | Export:

ItemName	SalesVolume
Software	112
Surface Pro	111
MacBook Pro	61
Handbag#1	53
Coat#3	40
Sneaker#1	30
Large Notebook	18
Blackboard	15
Coat#2	12
Handbag#2	12
Coat#1	10
Office	3

- Produce a list of all customers who have purchased a particular item

```
SELECT C.CustomerId
FROM WithFriends_Sales S, WithFriends_Advertisements A, customers_acc C
WHERE S.ADId = A.ADId
        AND C.Acc = S.AccountNumber
        AND ItemName = $ItemName
```

```
1  SELECT C.CustomerId
2  FROM WithFriends_Sales S, WithFriends_Advertisements A, customers_acc C
3  WHERE S.ADId = A.ADId
4      AND C.Acc = S.AccountNumber
5      AND ItemName = "Yida's Notes"
```

100% 31:5

Result Grid | Filter Rows: Search | Export:

CustomerId
100004
100007
100013

- Produce a list of all items for a given company

```
SELECT ItemName
FROM WithFriends_Advertisements
WHERE Company = $CompanyName;
```

```
1  SELECT ItemName
2  FROM WithFriends_Advertisements
3  WHERE Company = 'Yida Corp';
4
```

100% 29:3

Result Grid | Filter Rows: Search | Expo

ItemName
Software
Yida's Notes

Customer-Representative-Level Transactions

Customer Representatives should be thought of as sales agents and should be able to

- Create an advertisement

INSERT INTO WithFriends_Advertisements (ADId, EmployeeId, Type, Date, Company, ItemName, Content, UnitPrice, Units) VALUES (...)

```
1  INSERT INTO WithFriends_Advertisements (ADId, EmployeeId, Type, PostDate,
2  Company, ItemName, Content, UnitPrice)
3  VALUES (800011, 100029, 'Job', '2016-11-18', 'Stony Brook',
4  'Student#1', 'Great Programmer', '100000');
5  SELECT * FROM WithFriends_Advertisements;
```

100% 1:4

Result Grid | Filter Rows: Search | Edit: | Export/Import:

ADId	EmployeeId	Type	PostDate	Company	ItemName	Content	UnitPrice
800001	100021	Commercial	2016-11-17	H&M	Coat#1	Worm and water-proof	100
800002	100021	Commercial	2016-11-17	A&F	Coat#2	Worm	150
800003	100022	Educational	2016-11-17	Blackboard Inc.	Blackboard	Easy to use	300
800004	100023	Commercial	2016-11-17	Canada Goose	Coat#3	Great	900
800005	100024	Educational	2016-11-17	Nike	Sneaker#1	Confortable for running	150
800006	100024	Commercial	2016-11-17	Apple	MacBook Pro	Powerful	1999
800007	100025	Commercial	2016-11-17	Microsoft	Surface Pro	Ppwerful too	1899
800008	100024	Educational	2016-11-17	Staple	Large Notebook	Cheap	3
800009	100026	Commercial	2016-11-17	Dior	Handbag#1	Luxury	3000
800010	100027	Commercial	2016-11-17	Michael Kors	Handbag#2	Great deal	500
800011	100029	Job	2016-11-18	Stony Brook	Student#1	Great Programmer	100000
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

- Delete an advertisement

DELETE FROM WithFriends_Advertisements WHERE ADId = $ADId

```
1  DELETE FROM WithFriends_Advertisements
2  WHERE ADId = 800011;
3  SELECT * FROM WithFriends_Advertisements;
```

100% 21:2

Result Grid | Filter Rows: Search | Edit: | Export/Import:

ADId	EmployeeId	Type	PostDate	Company	ItemName	Content	UnitPrice
800001	100021	Commercial	2016-11-17	H&M	Coat#1	Worm and water-proof	100
800002	100021	Commercial	2016-11-17	A&F	Coat#2	Worm	150
800003	100022	Educational	2016-11-17	Blackboard Inc.	Blackboard	Easy to use	300
800004	100023	Commercial	2016-11-17	Canada Goose	Coat#3	Great	900
800005	100024	Educational	2016-11-17	Nike	Sneaker#1	Confortable for running	150
800006	100024	Commercial	2016-11-17	Apple	MacBook Pro	Powerful	1999
800007	100025	Commercial	2016-11-17	Microsoft	Surface Pro	Ppwerful too	1899
800008	100024	Educational	2016-11-17	Staple	Large Notebook	Cheap	3
800009	100026	Commercial	2016-11-17	Dior	Handbag#1	Luxury	3000
800010	100027	Commercial	2016-11-17	Michael Kors	Handbag#2	Great deal	500
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

- Produce customer mailing lists

CREATE VIEW Customers (CustomerId) As
SELECT DISTINCT U.UserId
FROM WithFriends_Sales S, WithFriends_Users U
WHERE S.AccountNumber = U.AccountNumber;

SELECT U.FirstName, U. LastName, U.EmailID
FROM WithFriends_Users U, Customers C
WHERE U.UserId = C.CustomerId;

```
1  CREATE VIEW Customers (CustomerId) As
2  SELECT DISTINCT U.UserId
3  FROM WithFriends_Sales S, WithFriends_Users U
4  WHERE S.AccountNumber = U.AccountNumber;
5
6  SELECT U.FirstName, U. LastName, U.EmailID
7  FROM WithFriends_Users U, Customers C
8  WHERE U.UserId = C.CustomerId;
```

100% 1:6

Result Grid | Filter Rows: Search | Export:

FirstName	LastName	EmailID
Michael	Collins	Michael.Collins@cse305.stonybrook.edu
Aria	Rose	Aria.Rose@cse305.stonybrook.edu
Jase	Black	Jase.Black@cse305.stonybrook.edu
Ellie	Franklin	Ellie.Franklin@cse305.stonybrook.edu
Mackenzie	Wells	Mackenzie.Wells@cse305.stonybrook.edu
Cameron	Armstrong	Cameron.Armstrong@cse305.stonybrook.edu
Levi	Ross	Levi.Ross@cse305.stonybrook.edu
Levi	West	Levi.West@cse305.stonybrook.edu
Christian	Davies	Christian.Davies@cse305.stonybrook.edu
Kylie	Davies	Kylie.Davies@cse305.stonybrook.edu

- Produce a list of item suggestions for a given customer (based on that customer's past transactions)

```
CREATE VIEW Customers_Acc (CustomerId, Acc) As
SELECT DISTINCT U.UserId, U.AccountNumber
FROM WithFriends_Sales S, WithFriends_Users U
WHERE S.AccountNumber = U.AccountNumber;

CREATE VIEW Suggest_Item (ItemName) As
SELECT DISTINCT A.ItemName
FROM WithFriends_Advertisements A, Customers_Acc C, WithFriends_Sales S
WHERE C.CustomerId = $GrivenCustomerId
        AND C.Acc = S.AccountNumber
   AND S.ADId = A.ADId;
```

```
1  CREATE VIEW Customers_Acc (CustomerId, Acc) As
2  SELECT DISTINCT U.UserId, U.AccountNumber
3  FROM WithFriends_Sales S, WithFriends_Users U
4  WHERE S.AccountNumber = U.AccountNumber;
5
6  CREATE VIEW Suggest_Item (ItemName) As
7  SELECT DISTINCT A.ItemName
8  FROM WithFriends_Advertisements A, Customers_Acc C, WithFriends_Sales S
9  WHERE C.CustomerId = 100002
10     AND C.Acc = S.AccountNumber
11     AND S.ADId = A.ADId;
12
13 SELECT * FROM WithFriends.suggest_item;
```

100% 29:10

Result Grid | Filter Rows: Search | Export:

ItemName
Coat#2
MacBook Pro
Surface Pro
Large Notebook

Customers should also be able to perform the following transactions with regard to advertisements

- Purchase one or more copies of an advertised item

```
INSERT INTO WithFriends_Sales (TransactionId, SalesDate, ADId, Units, AccountNumber)
VALUES ($NewTransId, $Time, $AD, $NumOfCopies, $Acc);
```

```
1  INSERT INTO WithFriends_Sales (TransactionId, SalesDate, ADId, Units, AccountNumber)
2  VALUES (110016, '2016-11-18 23:23:23', 800012, 12, 900012);
3
4  SELECT * FROM WithFriends_Sales;
```

100% 33:4

Result Grid | Filter Rows: Search | Edit: | Export/Import:

TransactionId	SalesDate	ADId	Units	AccountNumber
110001	2016-11-17 23:23:23	800001	10	900001
110002	2016-11-17 23:23:23	800002	12	900002
110003	2016-11-17 23:23:23	800003	15	900003
110004	2016-11-17 23:23:23	800004	40	900004
110005	2016-11-17 23:23:23	800005	30	900005
110006	2016-11-17 23:23:23	800006	60	900006
110007	2016-11-17 23:23:23	800007	110	900007
110008	2016-11-17 23:23:23	800008	13	900008
110009	2016-11-17 23:23:23	800009	53	900009
110010	2016-11-17 23:23:23	800010	12	900010
110011	2016-11-17 23:23:23	800006	1	900002
110012	2016-11-17 23:23:23	800007	1	900002
110013	2016-11-17 23:23:23	800008	5	900002
110014	2016-11-17 23:23:23	800011	3	900010
110015	2016-11-17 23:23:23	800012	100	900010
110016	2016-11-18 23:23:23	800012	12	900012
NULL	NULL	NULL	NULL	NULL

While customers (users) will not be permitted to access the database directly, they should be able to retrieve the following information:

- A customer's current groups

```
CREATE VIEW Cus_GroupId (GroupId) AS
SELECT GroupId
FROM WithFriends_GroupsMembers
WHERE UserId = $RequestCustomerId;

SELECT G.GroupName
FROM WithFriends_Groups G, Cus_GroupId D
WHERE G.GroupId = D.GroupId;
```

```
1  CREATE VIEW Cus_GroupId (GroupId) AS
2  SELECT GroupId
3  FROM WithFriends_GroupsMembers
4  WHERE UserId = 100005;
5
6  SELECT G.GroupName
7  FROM WithFriends_Groups G, Cus_GroupId D
8  WHERE G.GroupId = D.GroupId;
```

100% 23:7

Result Grid | Filter Rows: Search | Export:

GroupName
StonyBrookGang
My Family

- For each of a customer's accounts, the account history

SELECT * FROM WithFriends_Sales
WHERE AccountNumber = $Acc;

```
1  SELECT * FROM WithFriends_Sales
2  WHERE AccountNumber = 900002;
```

100% 30:2

Result Grid | Filter Rows: Search | Edit:

TransactionId	SalesDate	ADId	Units	AccountNumber
110002	2016-11-17 23:23:23	800002	12	900002
110011	2016-11-17 23:23:23	800006	1	900002
110012	2016-11-17 23:23:23	800007	1	900002
110013	2016-11-17 23:23:23	800008	5	900002
NULL	NULL	NULL	NULL	NULL

- Best-Seller list of items

CREATE VIEW Best_Seller (SellerId) AS
SELECT Seller
FROM Seller_Chart
ORDER BY TotalSell DESC
LIMIT 1;

SELECT DISTINCT A.ItemName
FROM WithFriends_Advertisements A, WithFriends_Sales S, Best_Seller B
WHERE B.SellerId = A.EmployeeId
    AND A.ADId = S.ADId

```sql
CREATE VIEW Best_Seller (SellerId) AS
SELECT Seller
FROM Seller_Chart
ORDER BY TotalSell DESC
LIMIT 1;

SELECT DISTINCT A.ItemName
FROM WithFriends_Advertisements A, WithFriends_Sales S, Best_Seller B
WHERE B.SellerId = A.EmployeeId
    AND A.ADId = S.ADId
```

100% 21:16

Result Grid | Filter Rows: Search | Export:

ItemName
Surface Pro
Office
Software

- Personalized item suggestion list

CREATE VIEW Suggest_Item (ItemName) As
SELECT DISTINCT A.ItemName
FROM WithFriends_Advertisements A, Customers_Acc C, WithFriends_Sales S
WHERE C.CustomerId = $GrivenCustomerId
AND C.Acc = S.AccountNumber
AND S.ADId = A.ADId;

```sql
CREATE VIEW Customers_Acc (CustomerId, Acc) As
SELECT DISTINCT U.UserId, U.AccountNumber
FROM WithFriends_Sales S, WithFriends_Users U
WHERE S.AccountNumber = U.AccountNumber;

CREATE VIEW Suggest_Item (ItemName) As
SELECT DISTINCT A.ItemName
FROM WithFriends_Advertisements A, Customers_Acc C, WithFriends_Sales S
WHERE C.CustomerId = 100002
    AND C.Acc = S.AccountNumber
    AND S.ADId = A.ADId;

SELECT * FROM WithFriends.suggest_item;
```

100% 29:10

Result Grid | Filter Rows: Search | Export:

ItemName
Coat#2
MacBook Pro
Surface Pro
Large Notebook