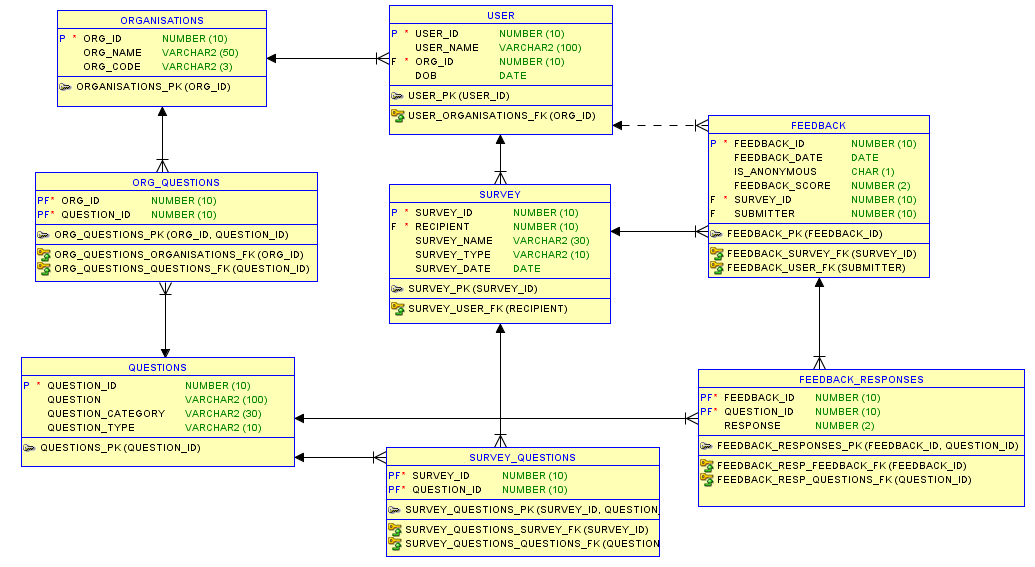
**Data Model Design and Schema for 360-Degree Feedback Application**

1. **Data Model Schema and Design**
2. **ERD diagram showing all entities and relationships**

****

Please click on the below link, if the above image is not proper/resizable

[360\_Degree\_Feedback\_DataModel\_IMG](https://drive.google.com/file/d/0B3_a7_090OxOVUZxMmM1aExIOFE/view?usp=sharing)

1. **Description of Data Model**

**Entity: ORGANISATIONS**

Organizations table contains different departments of a company, with ORG\_ID as primary key

**Entity: QUESTIONS**

Questions contain survey questions across all departments with QUESTION\_ID as primary key

**Entity: ORG\_QUESTIONS**

ORG\_QUESTIONS is an association table between **ORGANISATIONS** and **QUESTIONS,** to map many-many relationships between orgs and questions since each org can have multiple questions and each question can be associated with one or more orgs. (ORG\_ID, QUESTION\_ID) acts as primary key and both are foreign keys with respect to their associative tables.

**Entity: USER**

User is employee of any org and belongs to only one org. USER\_ID is primary key and ORG\_ID is foreign key.

**Entity: SURVEY**

Survey has all surveys identified by SURVEY\_ID as primary key and has attributes like Survey name, date, type and recipient, conducting the survey.

**Entity: SURVEY\_QUESTIONS**

Each survey has many questions and each question can be associated with one or more surveys. To map this association, SURVEY\_QUESTIONS association table is used.

**Entity: FEEDBACK**

Feedback stores the feedback information for a particular survey including feedback score, submitter, whether the response is anonymous, feedback date etc. FEEDBACK\_ID is the primary key and SURVEY\_ID is the foreign key. Each feedback has many responses and hence there is one-many mapping between feedback and feedback responses

**Entity: FEEDBACK\_RESPONSES**

Feedback responses is the association table between feedback questions and their responses where the primary key is (FEEDBACK\_ID, QUESTION\_ID). Questions table has one-may relationship with this table

1. **Physical Schema.**

create table feedback

(

feedback\_id number (10) not null ,

feedback\_date date ,

is\_anonymous char (1) ,

feedback\_score number (2) ,

survey\_id number (10) not null ,

submitter number (10)

) ;

alter table feedback add constraint feedback\_pk primary key ( feedback\_id ) ;

create table feedback\_responses

(

feedback\_id number (10) not null ,

question\_id number (10) not null ,

response number (2)

) ;

alter table feedback\_responses add constraint feedback\_responses\_pk primary key ( feedback\_id, question\_id ) ;

create table organisations

(

org\_id number (10) not null ,

org\_name varchar2 (50) ,

org\_code varchar2 (3)

) ;

alter table organisations add constraint organisations\_pk primary key ( org\_id ) ;

create table org\_questions

(

org\_id number (10) not null ,

question\_id number (10) not null

) ;

alter table org\_questions add constraint org\_questions\_pk primary key ( org\_id, question\_id ) ;

create table questions

(

question\_id number (10) not null ,

question varchar2 (100) ,

question\_category varchar2 (30) ,

question\_type varchar2 (10)

) ;

alter table questions add constraint questions\_pk primary key ( question\_id ) ;

create table survey

(

survey\_id number (10) not null ,

recipient number (10) not null ,

survey\_name varchar2 (30) ,

survey\_type varchar2 (10) ,

survey\_date date

) ;

alter table survey add constraint survey\_pk primary key ( survey\_id ) ;

create table survey\_questions

(

survey\_id number (10) not null ,

question\_id number (10) not null

) ;

alter table survey\_questions add constraint survey\_questions\_pk primary key ( survey\_id, question\_id ) ;

create table "user"

(

user\_id number (10) not null ,

user\_name varchar2 (100) ,

org\_id number (10) not null ,

dob date

) ;

alter table "user" add constraint user\_pk primary key ( user\_id ) ;

alter table feedback\_responses add constraint feedback\_resp\_feedback\_fk foreign key ( feedback\_id ) references feedback ( feedback\_id ) ;

alter table feedback\_responses add constraint feedback\_resp\_questions\_fk foreign key ( question\_id ) references questions ( question\_id ) ;

alter table feedback add constraint feedback\_survey\_fk foreign key ( survey\_id ) references survey ( survey\_id ) ;

alter table feedback add constraint feedback\_user\_fk foreign key ( submitter ) references "user" ( user\_id ) ;

alter table org\_questions add constraint org\_questions\_organisations\_fk foreign key ( org\_id ) references organisations ( org\_id ) ;

alter table org\_questions add constraint org\_questions\_questions\_fk foreign key ( question\_id ) references questions ( question\_id ) ;

alter table survey\_questions add constraint survey\_questions\_questions\_fk foreign key ( question\_id ) references questions ( question\_id ) ;

alter table survey\_questions add constraint survey\_questions\_survey\_fk foreign key ( survey\_id ) references survey ( survey\_id ) ;

alter table survey add constraint survey\_user\_fk foreign key ( recipient ) references "user" ( user\_id ) ;

alter table "user" add constraint user\_organisations\_fk foreign key ( org\_id ) references organisations ( org\_id ) ;

1. **Insert scripts for model data**

/\*insert into ORG\*/

delete from ORGANISATIONS ;

insert into ORGANISATIONS (ORG\_ID, ORG\_CODE, ORG\_NAME) values (1, 'HR', 'Human Resources');

insert into ORGANISATIONS (ORG\_ID, ORG\_CODE, ORG\_NAME) values (2, 'OR', 'Operations');

insert into ORGANISATIONS (ORG\_ID, ORG\_CODE, ORG\_NAME) values (3, 'IT', 'Information Technology');

insert into ORGANISATIONS (ORG\_ID, ORG\_CODE, ORG\_NAME) values (4, 'FI', 'Finance');

insert into ORGANISATIONS (ORG\_ID, ORG\_CODE, ORG\_NAME) values (5, 'MR', 'Marketing');

/\*insert into questions\*/

delete from QUESTIONS ;

insert into QUESTIONS (QUESTION\_ID, QUESTION,QUESTION\_CATEGORY, QUESTION\_TYPE) values (1, 'Rate HR Dept', 'FEEDBACK', 'Dept');

insert into QUESTIONS (QUESTION\_ID, QUESTION,QUESTION\_CATEGORY, QUESTION\_TYPE) values (2, 'Rate OR Dept', 'FEEDBACK', 'Dept');

insert into QUESTIONS (QUESTION\_ID, QUESTION,QUESTION\_CATEGORY, QUESTION\_TYPE) values (3, 'Rate FI Dept', 'FEEDBACK', 'Dept');

insert into QUESTIONS (QUESTION\_ID, QUESTION,QUESTION\_CATEGORY, QUESTION\_TYPE) values (4, 'Rate IT Dept', 'FEEDBACK', 'Dept');

insert into QUESTIONS (QUESTION\_ID, QUESTION,QUESTION\_CATEGORY, QUESTION\_TYPE) values (5, 'Rate MR Dept', 'FEEDBACK', 'Dept');

insert into QUESTIONS (QUESTION\_ID, QUESTION,QUESTION\_CATEGORY, QUESTION\_TYPE) values (6, 'Fun activities', 'SUGGESTIONS', 'GENERAL');

insert into QUESTIONS (QUESTION\_ID, QUESTION,QUESTION\_CATEGORY, QUESTION\_TYPE) values (7, 'Employee engagement', 'SUGGESTIONS', 'GENERAL');

insert into QUESTIONS (QUESTION\_ID, QUESTION,QUESTION\_CATEGORY, QUESTION\_TYPE) values (8, 'Annual outings', 'SUGGESTIONS', 'GENERAL');

insert into QUESTIONS (QUESTION\_ID, QUESTION,QUESTION\_CATEGORY, QUESTION\_TYPE) values (9, 'Workplace improvement', 'SUGGESTIONS', 'GENERAL');

/\*insert into ORG\_QUESTIONS \*/

delete from ORG\_QUESTIONS;

insert into ORG\_QUESTIONS (ORG\_ID, QUESTION\_ID) values (1, 1);

insert into ORG\_QUESTIONS (ORG\_ID, QUESTION\_ID) values (1, 8);

insert into ORG\_QUESTIONS (ORG\_ID, QUESTION\_ID) values (2, 2);

insert into ORG\_QUESTIONS (ORG\_ID, QUESTION\_ID) values (3, 3);

insert into ORG\_QUESTIONS (ORG\_ID, QUESTION\_ID) values (3, 7);

insert into ORG\_QUESTIONS (ORG\_ID, QUESTION\_ID) values (3, 8);

insert into ORG\_QUESTIONS (ORG\_ID, QUESTION\_ID) values (5, 6);

insert into ORG\_QUESTIONS (ORG\_ID, QUESTION\_ID) values (5, 9);

/\*insert into users\*/

delete from "USER";

insert into "USER" (USER\_ID, USER\_NAME, ORG\_ID, DOB) values (1, 'James', 1, '');

insert into "USER" (USER\_ID, USER\_NAME, ORG\_ID, DOB) values (2, 'Wilson', 2, '');

insert into "USER" (USER\_ID, USER\_NAME, ORG\_ID, DOB) values (3, 'Ryan', 3, to\_date('01-19-2017', 'MM-DD-YYYY'));

insert into "USER" (USER\_ID, USER\_NAME, ORG\_ID, DOB) values (4, 'Michael', 3, '');

insert into "USER" (USER\_ID, USER\_NAME, ORG\_ID, DOB) values (5, 'Jared', 3, '');

insert into "USER" (USER\_ID, USER\_NAME, ORG\_ID, DOB) values (6, 'Ramesh', 3, '');

/\*

/\* insert into survey\_questions\*/

insert into SURVEY (SURVEY\_ID,SURVEY\_NAME,SURVEY\_TYPE,RECIPIENT, SURVEY\_DATE) values (1,'Early Year Survey','GENERAL', 3, to\_date('01-05-2017', 'MM-DD-YYYY'));

insert into SURVEY (SURVEY\_ID,SURVEY\_NAME,SURVEY\_TYPE,RECIPIENT, SURVEY\_DATE) values (2,'Quarterly Year Survey','QUARTERLY', 1, to\_date('03-01-2017', 'MM-DD-YYYY'));

insert into SURVEY (SURVEY\_ID,SURVEY\_NAME,SURVEY\_TYPE,RECIPIENT, SURVEY\_DATE) values (3,'Mid Year Survey','GENERAL', 3, to\_date('06-01-2017', 'MM-DD-YYYY'));

insert into SURVEY (SURVEY\_ID,SURVEY\_NAME,SURVEY\_TYPE,RECIPIENT, SURVEY\_DATE) values (4,'End Year Survey','GENERAL', 2, to\_date('09-15-2017', 'MM-DD-YYYY'));

/\*insert into survey\_questions\*/

delete from SURVEY\_QUESTIONS;

insert into SURVEY\_QUESTIONS (SURVEY\_ID, QUESTION\_ID) values (1, 6);

insert into SURVEY\_QUESTIONS (SURVEY\_ID, QUESTION\_ID) values (1, 7);

insert into SURVEY\_QUESTIONS (SURVEY\_ID, QUESTION\_ID) values (1, 8);

insert into SURVEY\_QUESTIONS (SURVEY\_ID, QUESTION\_ID) values (1, 9);

insert into SURVEY\_QUESTIONS (SURVEY\_ID, QUESTION\_ID) values (1, 1);

insert into SURVEY\_QUESTIONS (SURVEY\_ID, QUESTION\_ID) values (2, 1);

insert into SURVEY\_QUESTIONS (SURVEY\_ID, QUESTION\_ID) values (2, 2);

insert into SURVEY\_QUESTIONS (SURVEY\_ID, QUESTION\_ID) values (2, 3);

insert into SURVEY\_QUESTIONS (SURVEY\_ID, QUESTION\_ID) values (2, 5);

/\*insert into user\_feedback\*/

delete from FEEDBACK;

insert into FEEDBACK (FEEDBACK\_ID, FEEDBACK\_DATE,IS\_ANONYMOUS, FEEDBACK\_SCORE, SURVEY\_ID, SUBMITTER) values (1, to\_date('01-15-2017', 'MM-DD-YYYY'), 0, 8,1,1);

insert into FEEDBACK (FEEDBACK\_ID, FEEDBACK\_DATE,IS\_ANONYMOUS, FEEDBACK\_SCORE, SURVEY\_ID, SUBMITTER) values (2, to\_date('01-17-2017', 'MM-DD-YYYY'), 0, 7, 1, 2);

insert into FEEDBACK (FEEDBACK\_ID, FEEDBACK\_DATE,IS\_ANONYMOUS, FEEDBACK\_SCORE, SURVEY\_ID, SUBMITTER) values (3, to\_date('01-19-2017', 'MM-DD-YYYY'), 1, 10,1,null);

insert into FEEDBACK (FEEDBACK\_ID, FEEDBACK\_DATE,IS\_ANONYMOUS, FEEDBACK\_SCORE, SURVEY\_ID, SUBMITTER) values (4, to\_date('01-23-2017', 'MM-DD-YYYY'), 0, 4,1,4);

insert into FEEDBACK (FEEDBACK\_ID, FEEDBACK\_DATE,IS\_ANONYMOUS, FEEDBACK\_SCORE, SURVEY\_ID, SUBMITTER) values (5, to\_date('01-25-2017', 'MM-DD-YYYY'), 1, 9,1,null);

insert into FEEDBACK (FEEDBACK\_ID, FEEDBACK\_DATE,IS\_ANONYMOUS, FEEDBACK\_SCORE, SURVEY\_ID, SUBMITTER) values (6, to\_date('03-01-2017', 'MM-DD-YYYY'), 0, 8,2,5);

insert into FEEDBACK (FEEDBACK\_ID, FEEDBACK\_DATE,IS\_ANONYMOUS, FEEDBACK\_SCORE, SURVEY\_ID, SUBMITTER) values (7, to\_date('03-02-2017', 'MM-DD-YYYY'), 1, 5,2,null);

insert into FEEDBACK (FEEDBACK\_ID, FEEDBACK\_DATE,IS\_ANONYMOUS, FEEDBACK\_SCORE, SURVEY\_ID, SUBMITTER) values (8, to\_date('03-03-2017', 'MM-DD-YYYY'), 0, 7,2,2);

insert into FEEDBACK (FEEDBACK\_ID, FEEDBACK\_DATE,IS\_ANONYMOUS, FEEDBACK\_SCORE, SURVEY\_ID, SUBMITTER) values (9, to\_date('03-04-2017', 'MM-DD-YYYY'), 1, 9,2,null);

/\*insert into feedback\_responses\*/

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (1, 6, 6);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (1, 7, 10);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (1, 8, 8);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (1, 9, 9);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (1, 1, 5);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (2, 6, 8);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (2, 7, 7);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (2, 8, 4);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (2, 9, 10);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (2, 1, 8);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (3, 6, 5);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (3, 7, 5);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (3, 8, 8);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (3, 9, 8);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (3, 1, 7);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (4, 6, 6);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (4, 7, 6);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (4, 8, 7);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (4, 1, 9);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (5, 6, 8);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (5, 7, 9);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (5, 8, 10);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (5, 9, 3);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (5, 1, 10);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (6, 1, 3);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (6, 2, 9);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (6, 3, 7);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (6, 4, 6);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (7, 1, 8);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (7, 2, 6);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (7, 3, 8);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (7, 4, 8);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (8, 2, 9);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (8, 1, 10);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (8, 3, 6);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (8, 4, 8);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (9, 1, 9);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (9, 2, 9);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (9, 3, 10);

insert into FEEDBACK\_RESPONSES (FEEDBACK\_ID, QUESTION\_ID, RESPONSE) values (9, 4, 7);

1. **QUERIES**
2. Query to return feedback responses from a submitter

select s.survey\_name, u1.user\_name as recipient, u2.user\_name as submitter, q.question, fr.response, f.is\_anonymous from survey s

join feedback f on s.survey\_id = f.survey\_id

join "user" u1 on u1.user\_id = s.recipient

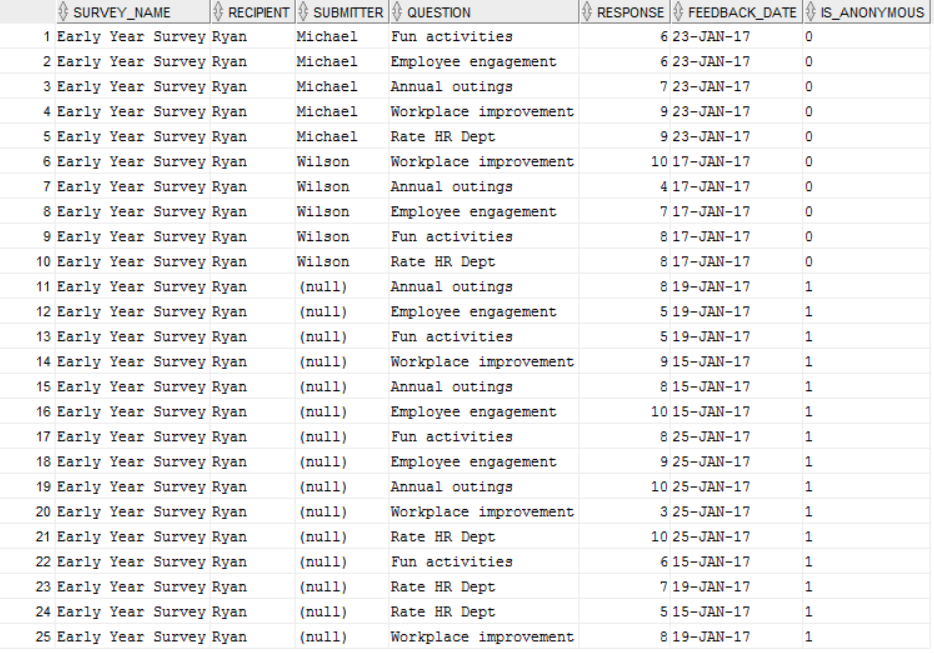
join feedback\_responses fr on f.feedback\_id = fr.feedback\_id

join questions q on q.question\_id = fr.question\_id

left outer join "user" u2 on u2.user\_id = f.submitter

where s.survey\_id = 1 order by submitter;

**Query Result**



1. Query to return avg feedback score across a time period

select s.survey\_name, avg(f.feedback\_score) as avg\_feedback\_score from survey s

join feedback f on s.survey\_id = f.survey\_id

join feedback\_responses fr on f.feedback\_id = fr.feedback\_id

join questions q on q.question\_id = fr.question\_id

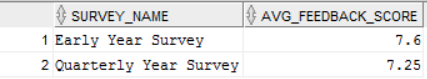
left outer join "user" u on u.user\_id = f.submitter

where s.survey\_date <= to\_date('01-31-2017', 'mm-dd-yyyy') group by s.survey\_name;

Query result



Query result across all surveys



Assumptions:

1. Feedback table contains avg score across all responses for a particular submitter
2. It is normalized to be between 1 and 10
3. Algorithm to find average scores across all submitters and all feedback per survey request

**double** findAverageScore (List <FeedBack> feedBackList) {  
 **double** totalScore = 0, avgScore = 0;  
 **for** (FeedBack feedback : feedBackList) {  
 **double** submitterScore = 0;  
 **for** (Response response : feedback.getAllResponses()) {  
 submitterScore += response.getScore();  
 }  
 totalScore += submitterScore;  
 }  
 **int** surveySize = feedBackList.getSize();  
 avgScore = totalScore / surveySize;  
 **return** avgScore;  
}

Assumptions:

1. The datastructure is constructed as above where a list of all feedbacks for a particular survey are stored in the list
2. Each entry in the list has feedback responses for a single submitter.
3. Feedback is a wrapper where it contains list of submitter responses along with the score for each response
4. The score for every response is normalized to be between 1 and 10, since that is the range provided for feedback.

The above algorithm can be iterated for all surveys where each survey score can be normalized to be between 1 and 10, if each survey allows a different feedback scoring scale and avg normalized score can be calculated across all surveys for a particular year or years together.

Pseudocode

avgAcrossSurveys = 0;

totalNormalizedScore = 0;

For(survey in all Surveys):

surveyScore = 0;

Calculate score for each survey using previous survey score calculation algo;

surveyScore = findAverageScore(Survey);

Normalize the survey score to be between 1 and 10;

surveyScore = normalized score;

totalNormalizedScore = totalNormalizedScore+ surveyScore

avgAcrossAllSurveys = totalNormalizedScore/no of surveys