

Software Engineer Screening Test –Minuli Bambarandage

3. SQL

3.1. Create tables and insert data

Create user table and insert data:

```
CREATE TABLE user (  
    [id] INT,  
    [firstName] VARCHAR(255),  
    [lastName] VARCHAR(255),  
    [email] VARCHAR(255),  
    [cultureID] INT,  
    [deleted] BIT,  
    [country] VARCHAR(255),  
    [isRevokeAccess] BIT,  
    [created] DATETIME  
);  
  
INSERT INTO "user" ([id], [firstName], [lastName], [email], [cultureID], [deleted], [country],  
[isRevokeAccess], [created])  
  
VALUES  
  
(1, 'Victor', 'Shevchenko', 'vs@ gmail.com', 1033, 1, 'US', 0, '2011-04-05'),  
(2, 'Oleksandr', 'Petrenko', 'op@ gmail.com', 1034, 0, 'UA', 0, '2014-05-01'),  
(3, 'Victor', 'Tarasenko', 'vt@gmail.com', 1033, 1, 'US', 1, '2015-07-03'),  
(4, 'Sergiy', 'Ivanenko', 'sergiy@gmail.com', 1046, 0, 'UA', 1, '2010-02-02'),  
(5, 'Vitalii', 'Danilchenko', 'shumko@ gmail.com', 1031, 0, 'UA', 1, '2014-05-01'),  
(6, 'Joe', 'Dou', 'joe@ gmail.com', 1032, 0, 'US', 1, '2009-01-01'),  
(7, 'Marko', 'Polo', 'marko@gmail.com', 1033, 1, 'UA', 1, '2015-07-03');
```

Create group table and insert data:

```
CREATE TABLE "group" (  
    "id" INT,  
    "name" VARCHAR(255),  
    "created" DATETIME  
);
```

```
INSERT INTO "group" ("id", "name", "created")  
VALUES  
(10, 'Support', '2010-02-02'),  
(12, 'Dev team', '2010-02-03'),  
(13, 'Apps team', '2011-05-06'),  
(14, 'TEST - dev team', '2013-05-06'),  
(15, 'Guest', '2014-02-02'),  
(16, 'TEST-QA-team', '2014-02-02'),  
(17, 'TEST-team', '2011-01-07');
```

Create groupMembership table and insert data:

```
CREATE TABLE "groupMembership" (  
    "id" INT,  
    "userID" INT,  
    "groupID" INT,  
    "created" DATETIME  
);
```

```
INSERT INTO "groupMembership" ("id", "userID", "groupID", "created")
VALUES
(110, 2, 10, '2010-02-02'),
(112, 3, 15, '2010-02-03'),
(114, 1, 10, '2014-02-02'),
(115, 1, 17, '2011-05-02'),
(117, 4, 12, '2014-07-13'),
(120, 5, 15, '2014-06-15');
```

3.2. Select names of all empty test groups (group name starts with “TEST-”).

```
SELECT g."name"
FROM "group" g
LEFT JOIN "groupMembership" gm ON g."id" = gm."groupID"
WHERE g."name" LIKE 'TEST-%'
GROUP BY g."id", g."name"
HAVING COUNT(gm."id") = 0;
```

3.3. Select user first names and last names for the users that have Victor as a first name and are not members of any test groups (they may be members of other groups or have no membership in any groups at all).

```
SELECT u."firstName", u."lastName"
FROM "user" u
LEFT JOIN "groupMembership" gm ON u."id" = gm."userID"
```

```
LEFT JOIN "group" g ON gm."groupID" = g."id"  
WHERE u."firstName" = 'Victor'  
AND (g."name" NOT LIKE 'TEST-%' OR g."name" IS NULL);
```

3.4. Select users and groups for which user was created before the group for which he(he) is member of.

```
SELECT u."firstName", u."lastName", g."name" AS "groupName"  
FROM "user" u  
JOIN "groupMembership" gm ON u."id" = gm."userID"  
JOIN "group" g ON gm."groupID" = g."id"  
WHERE u."created" < g."created";
```