

## Software Engineer – Screening Test

**NOTE:** Please be very accurate and pay attention to all details while working with test tasks. These tasks are requirements. If anything in requirements looks ambiguous or unclear to you then please always make efforts to clarify. Your requirements elicitation and analysis skills will be appraised too. Please remember that failing to collect qualitative requirements can spoil the result of all subsequent work.

### 1. Java Script.

- 1.1. Extend JS Date object with a method `daysTo()` which returns number of complete days between any pair of JS date objects: `d1.daysTo(d2)` should return quantity of complete days from `d1` to `d2`.
- 1.2. Please order by Total  
Develop a program which produces ordered array of sales. Input: array of objects with the following structure `{amount: 10000, quantity: 10}`. Output: new array of ordered sales. Array element structure should be: `{amount: 10000, quantity: 10, Total: 100000}`, where `Total = amount * quantity`. Please order by Total and note that input array shall remain intact.
- 1.3. Develop a program “Object Projection”. Input: any JSON object; prototype object. Output: projected object. Projected object structure shall be intersection of source object and prototype object structures. Values of properties in projected object shall be the same as values of respective properties in source object.  
For example:

```
const src = {
  prop11: {
    prop21: 21,
    prop22: {
      prop31: 31,
      prop32: 32
    }
  },
  prop12: 12
};

const proto = {
  prop11: {
    prop22: null
  }
};

const res = {
  prop11: {
    prop22: {
      prop31: 31,
      prop32: 32
    }
  }
};
```

## 2. REST API

- 2.1. Develop a program in JS which returns array of free/busy intervals in a given time period for any shared Google calendar. Input: shared Google calendar ID; time period (starting and ending moments). Output: array of busy intervals.

Alternatively (if 3.1 is too difficult to develop) provide sequence of REST API calls that can be executed in REST API client (Postman) in order to achieve the same result.

### 3. SQL

Please prepare scripts executable on this Try-SQL Editor

[https://www.w3schools.com/sql/trysql.asp?filename=trysql\\_select\\_columns](https://www.w3schools.com/sql/trysql.asp?filename=trysql_select_columns)

#### 3.1. Create tables and insert data

Create table "user" with the following structure:

```
[id] int,  
[firstName] varchar(255),  
[lastName] varchar(255),  
[email] varchar(255),  
[cultureID] int,  
[deleted] bit,  
[country] varchar(255),  
[isRevokeAccess] bit,  
[created] datetime
```

Insert the data into the table:

```
(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 table "group" with the following structure:

```
[id] int,  
[name] varchar(255),  
[created] datetime
```

Insert the data into the table:

```
(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 table “groupMembership” with the following structure:

```
[id] int,  
[userID] int,  
[groupID] int,  
[created] datetime
```

Insert the data into the table:

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