MySQL

Consider the following MySQL tables :

```
CREATE TABLE `clients` (
    `id` INT UNSIGNED NOT NULL AUTO_INCREMENT PRIMARY KEY ,
    `name` VARCHAR( 100 ) CHARACTER SET utf8 COLLATE utf8_general_ci NOT NULL
) ENGINE = MYISAM
```

```
CREATE TABLE `calls` (
 `client_from` int(10) unsigned NOT NULL,
 `client_to` int(10) unsigned NOT NULL,
 `duration` int(10) unsigned NOT NULL,
 `cost` decimal(6,1) unsigned NOT NULL,
 `date` timestamp NOT NULL default '0000-00-00 00:00:00'
) ENGINE=MyISAM
```

The « clients » table stores the different clients of a company providing mobile phone services.

Each client has a unique identifier (the « id » field) and other information (only a name to keep it simple).

The « calls » table stores each call between clients of the company. The « client_from » is the id of the calling client and client_to is the id of the callee. These ids exist in the clients table. Duration is stored in seconds and cost in euros. The data field store the starting date of the call.

- All the SQL statements must be written as single query and for a MySQL 5 database.
- Answers are not unique. There exist several manners to get same results. Please give the most optimized choice and/or the most simple way if you can. **Please privilegiate solutions using EXPLICIT JOIN synthax (INNER JOIN, LEFT JOIN …)**
- If you don't know how to answer or have any doubts, please answer by explaining your misunderstanding or issues faced.

Please answer the following questions:

1. Write a SQL statement to only get the « name » of all the company's client.
2. Write a SQL statement adding the « Facebook » client to the « clients »'s table
3. Write a SQL statement to get the name and id of the clients who called at least one time. There must be unique results. Please give two possible answers if possible.
4. Write a SQL statement to get the name and id of the clients who received at least one call. There must be unique results. Please give two possible answers if possible.
5. Write a SQL statement to get the name and id of the clients who have not received a call yet. There must be unique results.
6. Write a SQL statement to get the name and id of the clients who have not called someone yet. There must be unique results.
7. Write the most simple SQL statement to get all the calls starting from the 24/01/2011 and the 03/03/2011 (start date and end date included).
8. Write a SQL statement to get the total cost of all calls made by the client « Adams » with the id 42. Only this total cost must be returned and it must be named as « total_cost »
9. Write a SQL statement to get the total cost associated to each caller.
10. Write a SQL statement to insert a new call in the « calls » table. This call was made by the client id 42 to the client id 56 and lasted 40 minutes. The communication cost is 0,1 euros/minute. The call began at 10pm on the 21/05/2009.
11. What is not optimized in the structure of the « calls » table regarding the previous statments you wrote (Q. 3, 4, 5, 6) ? Are there something else not optimized for this table ?

PHP, HTML and JS

1. Write a PHP code displaying the current date. The script must output the date in the following format: **14/04/1912 03:40**
2. Write a **simple (no formatting, no design)** HTML « login » form displaying an email texbox, a password textbox and a submit button. This form must send the data to the current PHP file using the appropriate HTTP method. You are required to write a proper HTML structure of your choice for the document displaying the form (but no CSS!).
3. Add to the previous form a Javascript Validation check (email should contain an @ and a dot). Password must not be empty. In case of errors, do not use the alert() function but display an error message above the form using JavaScript.
4. Javascript can be disabled on the browser. So add to the previous form a PHP validation once the user submits the form. This validation is the same than the JavaScript one. In case of error, display an error message above the HTML form.

Data and RegEx

1. Data extracting

Input:

An example text:

On February 13, 2009, at exactly 23:31:30 (UTC) the decimal representation of Unix time was equal to 1234567890. Parties and other celebrations were held around the world, among various technical subcultures, to celebrate this day.

Output:

Output to the browser all numbers greater than 4 digits contained in the given input text using PHP.

Guidelines

Use regular expressions

Of course, the output must work whatever is the text given as input

2. Data replacing

Input:

The following text:

*During the day, **Damien** is working.*

Output:

From the given input, output the following text to the browser:

***Damien** is happy to work.*

Guidelines

Use regular expressions functions only

No more than 2 lines of code expected. Bigger code will be rejected

3. Parsing XML (20min)

Input specifications:

The following XML file from the Youtube GDATA API:
http://gdata.youtube.com/feeds/api/playlists/4308E545B34F885D?v=2

Expected output:

Print to the browser the ***title***, ***yt:videoId*** and ***media:description*** of each video in the most simple and efficient way you can using PHP.