Preface

Complete the test in IDE or simple text editor of your choice. It's important to represent knowledge on volume and writing speed. The time to perform the test is 1 h. If you get ready sooner please let me know. Internet and other assisting tools (except help from a friend) are allowed. You are required to complete both “PHP coding” and “MySQL/MariaDB” parts. The order in which the tasks are performed is optional.

PHP coding

This part is mandatory. Solve as many as you can. Please note that solving more tasks gives you better overall score.

Task 1

Compose a script that shows a list of numbers from 1 to 30, each number on a separate line. If the number is divisible by three, replace the number with the text „divisible by three“. If the number is divisible by five, replace the number with the text „divisible by five“. If the number is divisible by both three and five, display “divisible by three and five”.

Task 2

Write a function that sums up all the members of an array and its sub-arrays. The members are integers or subarrays and don’t need further verification. Sub-arrays could also have arrays as members, the given array is multi-dimensional and has unlimited depth. Use only native PHP, non-array related functions.

Task 3

Implement a “groupByAuthor” method that returns an associative array containing an array of book names for each author. Input array is:

[
“Learning PHP” => “John Smith”,
“Understanding relational databases” => “Mary Little”,
“Freelancers” => “Robin Round”
“I love LISP” => “Mary Little”,
“Python for dummies” => John Smith”,
];

Expected result is (sorting order is not important):

[
“John Smith” => [“Learning PHP”, “Python for dummies”],

“Mary Little” => [“Understanding relational databases”, “I love LISP”],
“Robin Round” => [“Freelancers”]
];

Task 4

The user interface contains two types of user input controls: TextInput, which accepts all texts and NumericInput, which accepts only digits.
Implement the class TextInput that contains:

- Public function add($value) – adds the given text to the current value.
- Public function getValue() – returns the current value (string).

Implement the class NumericInput that:

- Inherits from TextInput.
- Overrides the add method so that each non-numeric text is ignored.

For example, the following code should output ‘10’:

```
$input = new NumericInput();
$input->add(‘1’);
$input->add(‘a’);
$input->add(‘0’);
echo $input->getValue();
```

Task 5

Consider the following code:

```
$str1 = 'yabadabadoo';
$str2 = 'yaba';
if (strpos($str1,$str2)) {
    echo "\"" . $str1 . "\" contains \"" . $str2 . "\"";
} else {
    echo "\"" . $str1 . "\" does not contain \"" . $str2 . "\"";
}
```

The output will be:
"yabadabadoo" does not contain "yaba"
Why? How can this code be fixed to work correctly?

Task 6

Write a program for this pattern:

```
*
* *
* * *
* * * *
* * * * *
* * * * * *
* * * * *
```

```
* * * *
* * *
* *
*
```

Task 7

Write a program for finding the biggest number in an array without using any array functions.

Task 8

Write a program that outputs multiplication table based on input argument. For example if input is 5, the result would be:

	1	2	3	4	5
1	1	2	3	4	5
2	2	4	6	8	10
3	3	6	9	12	15
4	4	8	12	16	20
5	5	10	15	20	25

Task 9

Below is program for calculating results for dart game. Results are calculated by average of three throws. Unfortunately program does not return correct answer. Fix the code.

```
$scores = [
"John" => [7, 8, 7],
"Sue" => [10, 8, 4],
"Tommy" => [8, 8, 7],
"Mary" => [7, 6, 6]
];

$averages = [];
$places = [];

foreach($scores as $player => $values) {
    $averages[$player] = $values[0] + $values[1] + $values[2] / 3;
}

arsort($averages);
$place = 0;
$lastscore = null;
foreach($averages as $player => $average) {
    if ($lastscore and $lastscore = $average) {
        echo sprintf("Place %s (tie) - %s\n", $place, $player);
    } else {
        $place++;
        echo sprintf("Place %s - %s\n", $place, $player);
    }
    $lastscore = $average;
}
```

MySQL/MariaDB

This part is mandatory. Solve task 1. Solve at least one (but the more the better) subtask from task 2. You may choose any of them. Solving more subtasks gives you better overall score.

Task 1

Consider simple book rental service. At the moment everything is stored in Excel worksheet. Example of worksheet is represented by following table.

Customer	Phone	Books rented	Borrow date	Return date
Mary Sue	5012345	"Mysteries of Java"	14.02.2020	
Alan Smith	5123456	"The Big Rewrite", "Design patterns"	17.02.2020	25.02.2020
Mary Sue	5598765	"Design patterns"	02.03.2020	04.03.2020
Joe Goodspeed	5234567	"Inversion of control"	02.03.2020	
Joe Goodspeed	5234567	"Design patterns"	04.03.2020	
Nicky Jones	5345678	"Why my code smells?"	07.03.2020	

Employees at shop are having more and more problems using this Excel worksheet. There are more borrowers every day. Our company received order for creating small modern program to solve those problems. You are given the task to design database schema. Based on information above design the best database schema you possibly can.

Task 2

Along with current trial tests file come two additional files, both related to current task:

- Small MySQL database dump (trial.sql). It contains schema and data for one table "Employees".
- Sample data file (data.csv). The data in the file represents single table "Employees" in database.

You may use either alternative. Solve following tests based on provided data:

1) Write a query to list the number of jobs available in the employees table.
2) Write a query to get the maximum salary of an employee working as a Programmer ("IT_PROG").
3) Write a query to get the average salary and number of employees working the department 90.
4) Write a query to get the number of employees with the same job.
5) Write a query to get the difference between the highest and lowest salaries.
6) Write a query to find the manager ID and the salary of the lowest-paid employee for that manager.
7) Write a query to get the department ID and the total salary payable in each department.
8) Write a query to get the average salary for each job ID excluding programmer.,

9) Write a query to get the total salary, maximum, minimum, average salary of employees (job ID wise), for department ID 90 only.
10) Write a query to get the job ID and maximum salary of the employees where maximum salary is greater than or equal to $4000.
11) Write a query to get the average salary for all departments employing more than 10 employees.