

1 Remarks

- the tasks don't have a certain order for processing; the numbering is meant only for easier distinction; choose the order the way you want
- in case you have questions, feel free to ask the other developers
- if you encounter parts in the code, which you consider debatable, add respective comments to the concerning sections

2 Coding Guidelines

The following guidelines are recommendations. It is allowed to deviate from them slightly.

- process the tasks using plain PHP
- use Object-oriented programming
- use K&R indentation style
- use PSR-12
- abide to the principles of clean code

3 Tasks

3.1 Refactoring

Refactor the class `customer` according to the rest of the code and the coding guidelines from above!

3.2 SQL

Write an SQL query for the given database in `database/testbike.sqlite3`, that lists the customer names together with their domain and the number of rentals they had! The result should look as follows:

```
--          --  --
Max Mustermann  de  11
Igor Walasmy    pl  24
John Doe        us  20
--          --  --
```

3.3 Tariff calculator

Implement the following tariffs in the respective prepared classes:

- Tariff "Alpha"

- 1 EUR/30m, at maximum 9 EUR/day
- Tariff “Beta”
 - using PLN (Zloty)
 - until 20m: 0 PLN
 - until 60m: 2 PLN
 - until 120m: 6 PLN
 - above 120m: 4 PLN/h
 - above 12h: calculated price so far + 200 PLN
- Tariff “Gamma”
 - 9am - 18pm: 2 EUR/h, at maximum 16 EUR/d
 - 18pm - 9am: additional 2 EUR to 2 EUR/h

Replace the body of the method stub `Rental::calculatePrice` with calls to your implementations and write tests for them in the file `test.php`!

3.4 Bikes

The implementation at hand does not involve bikes in any way. Describe and/or implement extensions for the system (database and source code) to rectify this!