**Тестовые задания**

**Задания, описанные ниже, в первую очередь, направлены на определения уровня знаний и способности размышлять и думать логически. Так же помогут определить какие пробелы в знаниях нужно закрыть и какие навыки подтянуть**

**Задание 1** «Погода и курс валют»

**Краткое описание:** HTML страница отображающая курс валют и погоду в Москве, на текущий момент.

**Пример внешнего вида:** [Макет](https://www.figma.com/file/pdnZndaO3ZlKn3rvBhHFdy/Untitled?node-id=0%3A1)

**Список валют:** Доллар США, Евро, Шведских крон, Японских иен, Канадский доллар

**Для получение курса использовать** - <http://www.cbr.ru/development/sxml/>

**Для получения погоды использовать можно использовать любое API** (Например: <https://openweathermap.org/api> или <https://www.weatherapi.com/> )

Так же на странице должна присутствовать кнопка, для обновления информации по AJAX

**Задание 2** «Новости»

**Описание:** Написать простое приложение MVC «Новости». Результатом задания должны быть три файла: Model – News.php, Controller – NewsController.php и View – NewsTemplate.php

**News.php**

У модели должны быть следующие свойства:

- Id (В конструкторе заполняется автоматически)

- Заголовок

- Анонс

- Основной текст

- Теги

- Дата публикации (В конструкторе заполняется сегодняшним днем в формате dd.mm.yyy hh:mm:ss)

Методы:

- Геттеры и сеттеры на каждое поле(кроме «Дата» и «Id» для них должен быть только геттер)

- Добавление новости в бд

- Извлечение новости из бд по части заголовка или id

- Удаление новости по id

**NewsController.php**

Имеет 3 метода:

Создание новости (/create)

Все значения для новой новости передаются через POST

Удаление новости(/delete?id=)

В get параметре передаётся id удаляемой новости

Рендер новости (/news?id= или /news?q=)

Поиск и отображение новости по id или заголовку

**NewsTemplate.php**

Шаблон отображения новости, используется только при показе

Важное замечание! Приложение не обязательно должно исполняться, главное в задании ,то как написан код, его структура и логика. Для реализации этого задания можно взять один из популярных MVC Framework`ов (Laraverl, Yii2 и т.д.)

**Задание 3** «Жизнь» - задание дополнительное и не обязательное, выполняется по желанию кандидата.

**Описание:** Имеется некое поле m\*n клеток. Каждая клетка имеет 2 состояния: «Активная» и «Мертвая». У клетки может быть до 8 соседей

|  |  |  |
| --- | --- | --- |
| Х | Х | Х |
| Х | О | Х |
| Х | Х | Х |

Начальное состояние поле определяется случайным образом. Далее запускается цикл, который останавливается в том случае, если все клетки не активны. На каждый тик цикла выполняются 4 операции «Активация», «Перегрузка», «Изоляция», и «Вымирание»

«Активация» - мертвая клетка, у которой ровно три активных соседа, становится активной в следующем состоянии поля.

«Перегрузка» - активная клетка, у которой активных соседей четыре или больше, «умрет» в следующем состоянии поля.

«Изоляция» - Активная клетка, у которой активных соседей один или меньше, «умрет» в следующем состоянии поля.

«Вымирание» - Активная клетка останется такой, только если у неё ровно 2 или 3 активных соседа, иначе «умрет» в следующем состоянии поля.

Пример результата реализации отправляю вместе с документом.