

Формирование главной

Задача: из массива (URL на изображение + текст + ссылка) сформировать HTML заданного формата и CSS-спрайт.

Скрипт будет запускаться по cron, редко. Результат должен быть сохранен в определенную папку.

Скрипт может быть выполнен в PHP, но может и в другом языке. Для формирования CSS-спрайтов можно использовать node.js (например, <https://www.npmjs.com/package/css-sprite>).

Получение изображений с удаленных узлов - php curl. Масштабирование можно делать средствами node.js, а можно php image magick (выбирайте сами - что проще и быстрее).

Если опыта достаточно - весь скрипт можно сделать на node.js.

Образец входа:

```
$data = array(
  'Название одной группы ссылок' => array(
    array( 'text' => 'Ссылка 1', 'image' => 'localfile.jpg', 'href' => 'http://www.ru' ),
    array( 'text' => 'Ссылка 2', 'image' => 'http://www.ru/remotefile.jpg', 'href' => 'http://www.ru' ),
  ),
  'Название второй группы ссылок' => array(
    array( 'text' => 'Ссылка 3', 'image' => 'localfile.jpg', 'href' => 'http://www.ru' ),
    array( 'text' => 'Ссылка 4', 'image' => 'http://www.ru/remotefile.jpg', 'href' => 'http://www.ru' ),
  )
);
```

Настройки в скрипте:

1. качество сжатия JPG
2. макс. размер изображения (max_width, max_height)
3. пути для сохранения CSS и HTML-вставки.
4. мин. % удачных ссылок.

Примерный алгоритм:

1. Скрипт запускается, проверяет что задан массив \$data.
2. Для каждой ссылки проверяет что заданы свойства text, image, href. Если не заданы - пропускаем ссылку.
3. Изображение локальное? Проверяем наличие файла, проверяем его ширину и высоту. Если больше max_*, - масштабируем по максимальному значению и обрезаем остальное (если высота больше ширины - масштабируем по высоте, остальное обрезаем). Сохраняем в папку для спрайта.
4. Изображение удаленное? Пробуем сохранить на диск с временным названием, повторяем манипуляцию из пункта 3, сохраняем в папку для спрайта.
5. Еще раз проверяем что получившиеся изображение читаемо и соответствует параметрам. Если проблемы - пропускаем ссылку.
6. Считаем общее количество получившихся удачных ссылок. Если их количество меньше установленного % успешных - выходим с ошибкой без сохранения HTML и спрайтов.
7. Если ссылок достаточно - формируем HTML-вставку по формату (например, название группы ссылок оформляется через <h1>, затем идет секция ul, li, каждая ссылка - через с классом, который соответствует формату спрайтов)
8. Перед пунктом 7 - формируем спрайт. Здесь самая важная задача - сделать правильную синхронизацию классов CSS и классов ссылок .
9. Если спрайт успешно сформирован - копируем получившийся спрайт + CSS + HTML вставку в заданные в скрипте папки.