Разработка диспетчера в своем MVC фреймворке

Вспомним текущее содержимое файла index.php:

```php
<?php
  namespace Core;

  error_reporting(E_ALL);
  ini_set('display_errors', 'on');

  spl_autoload_register(function($class) {
    // ваша реализация автозагрузки
  });

  $routes = require $_SERVER['DOCUMENT_ROOT'] .
     '/project/config/routes.php';

  $track = ( new Router ) -> getTrack($ro
    utes, $_SERVER['REQUEST_URI']);
?>
```

Как вы видите, пока у нас наш роутер возвращает объект класса Track, содержащий имя контроллера, имя действия и параметры из адресной строки. Кроме того, в предыдущем уроке мы с вами сделали родителя всех контроллеров.

Теперь мы можем сделать так, чтобы происходил вызов метода контроллера, данные которого содержатся в нашей переменной $track.

Этим будет заниматься специальный класс Dispatcher. Диспетчер будет получать объект класса Track и по его данным создавать объект указанного класса, вызывать метод этого класса, передавая в этот метод параметры.

Давайте добавим вызов диспетчера в файл index.php:

```php
<?php
  namespace Core;

  error_reporting(E_ALL);
  ini_set('display_errors', 'on');

  spl_autoload_register(function($class) {
    // ваша реализация автозагрузки
  });

  $routes = require $_SERVER['DOCUMENT_ROOT'] .
     '/project/config/routes.php';

  $track = ( new Router ) -> getTrack($ro
    utes, $_SERVER['REQUEST_URI']);

  // Вызов диспетчера:
  $page  = ( new Dispatcher ) -> getPage($tr
    ack);
?>
```

Вызов метода getPage нашего диспетчера будет вызывать метод render контроллера и возвращать то, что вернул вызов этого метода. Как вы уже знаете из предыдущего урока, метод render контроллера возвращает объект класса Page.

Вот заготовка нашего класса Dispatcher:

```php
<?php
  namespace Core;

  class Dispatcher
  {
    public function getPage(Track $track)
    {
      ...код
      return контроллер -> render(параметры);
    }
  }
?>
```

Задача 105.1

Используя мою заготовку реализуйте описанный класс Dispatcher. Проверьте его работу. При затруднениях посмотрите исходный код в файлах сделанного мною учебного фреймворка.