# Обзор тестового задания

**Введение**

Для тестового задания использован MVC фреймворк Yii версии 2. Структурно базовую комплектацию не менял.

# **База данных**

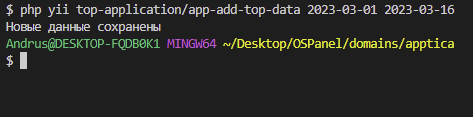
В качестве базы использовалась **MySql**. Создана таблица для топа приложений «**top\_application\_position**». Дамп базы есть в этой же директории под названием «**apptica.sql**».

**Запросы**

1. Создан класс «**app\commands\TopApplicationController**» для заполнения таблицы топа приложений с помощью апи запроса к «https://api.apptica.com/package/top\_history». Запрос не единоразовый, и может использоваться повторно, добавляя записи в таблицу.

Из любопытных свойств стоит отменить что записи с одинаковыми категорией и датой не дублируются. Кроме того, запись уходит в базу одним запросом, что снижает нагрузку если бы, объекты записывались по одному.

Для вызова команды в консоли прописывается команда «**php yii top-application/app-add-top-data 2023-03-01 2023-03-16**» где «2023-03-01» – начало периода и «2023-03-16» – конец периода для выгрузка топа.

Пример использования:  


1. Создан класс «**app\controllers\TopApplicationController**» для запроса по топу приложений.

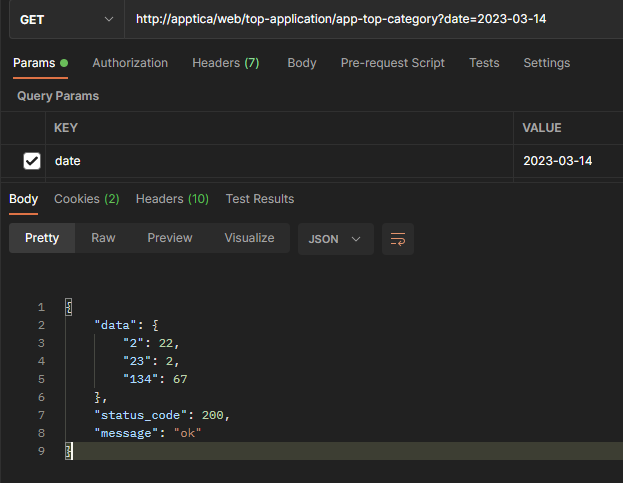
Get запрос «**http://apptica/web/top-application/app-top-category?date=2023-03-14**» отдает информацию по топу за конкретный день для всех имеющихся категорий. Либо же выдает ошибку. Ответ в формате json.  
  
Пример использования:  


Таблица возможных ситуаций:

|  |  |  |
| --- | --- | --- |
| status\_code | message | Info |
| 200 | ok | Если все прошло удачно |
| 404 | Not Found | Если данные не найдены |
| 500 | Set date to format Y-m-d | При отсутствии даты или неверном формате |
| 500 | Too many request | Больше 5 запросов в течении минуты |

**Дополнительно**  
  
Создано несколько вспомогательных классов, для дополнительных функций, таких как ограничение запросов на конкретную точку входа для конкретного ip адреса и логирование нужных данных в файловую систему сервера. Все дополнительные классы находятся в директории «components». Логи пишутся для каждого запроса в нужную директорию (основная «fileLog»).  
  
Пример логов:  
