### **1. Написать регулярное выражение, которое проверяет валидный IP-адрес. Например, 192.168.1.1 подойдет, а 256.300.1.1 — нет.**

**Делала на regex101.com**

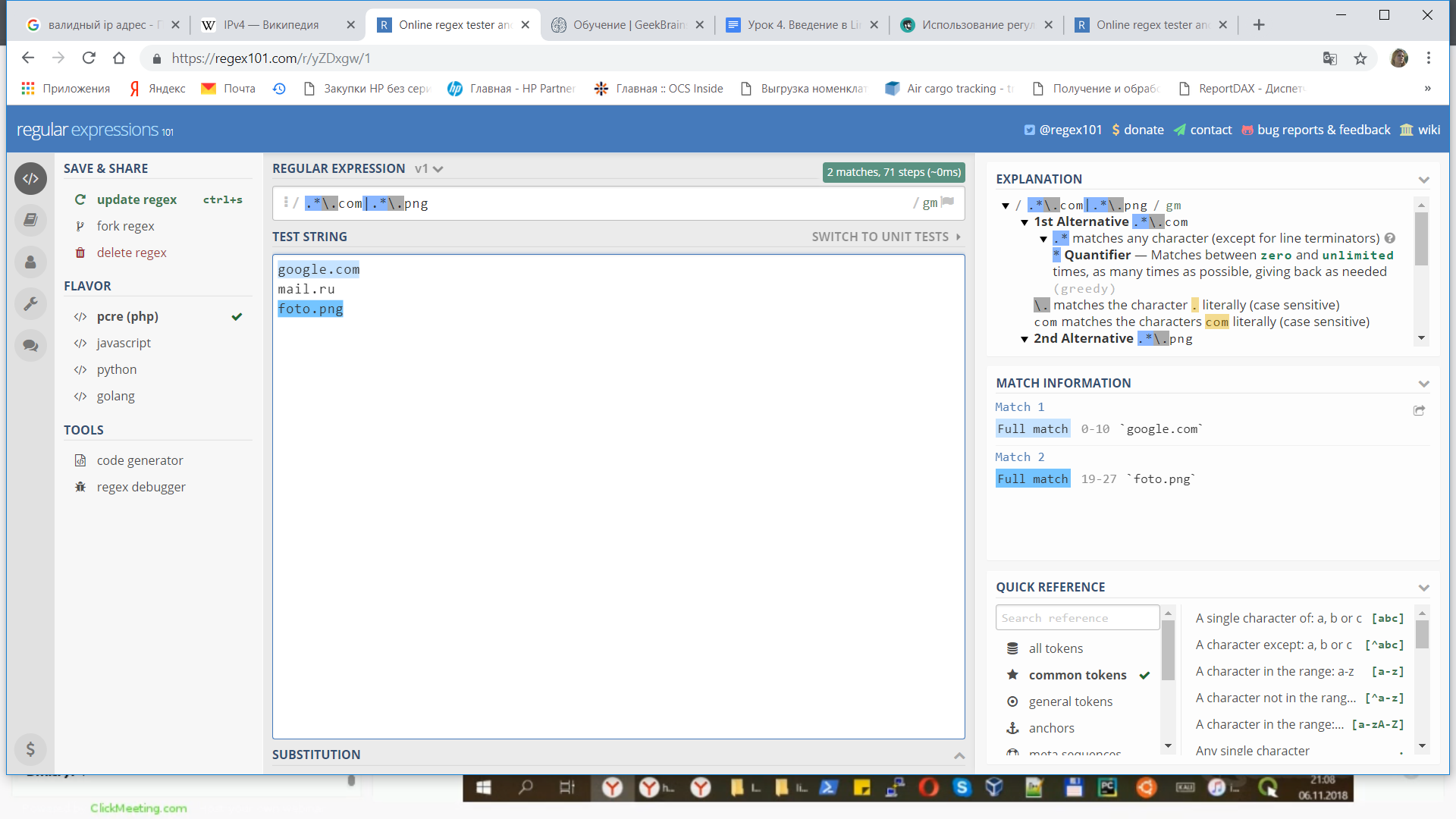
**^((2[0-4][0-5]|{[0-4][0-9]|1[0-9][0-9]|[1-9]?[0-9])\.){3}(25[0-5]|2[0-4][0-9]|1[0-9][0-9]|[1-9]?[0-9]?)$**

### 

### **2. Написать регулярное выражение, которое проверяет, является ли указанный файлом нужного типа (на выбор .com,.exe или .jpg,.png,.gif и т.д.).**

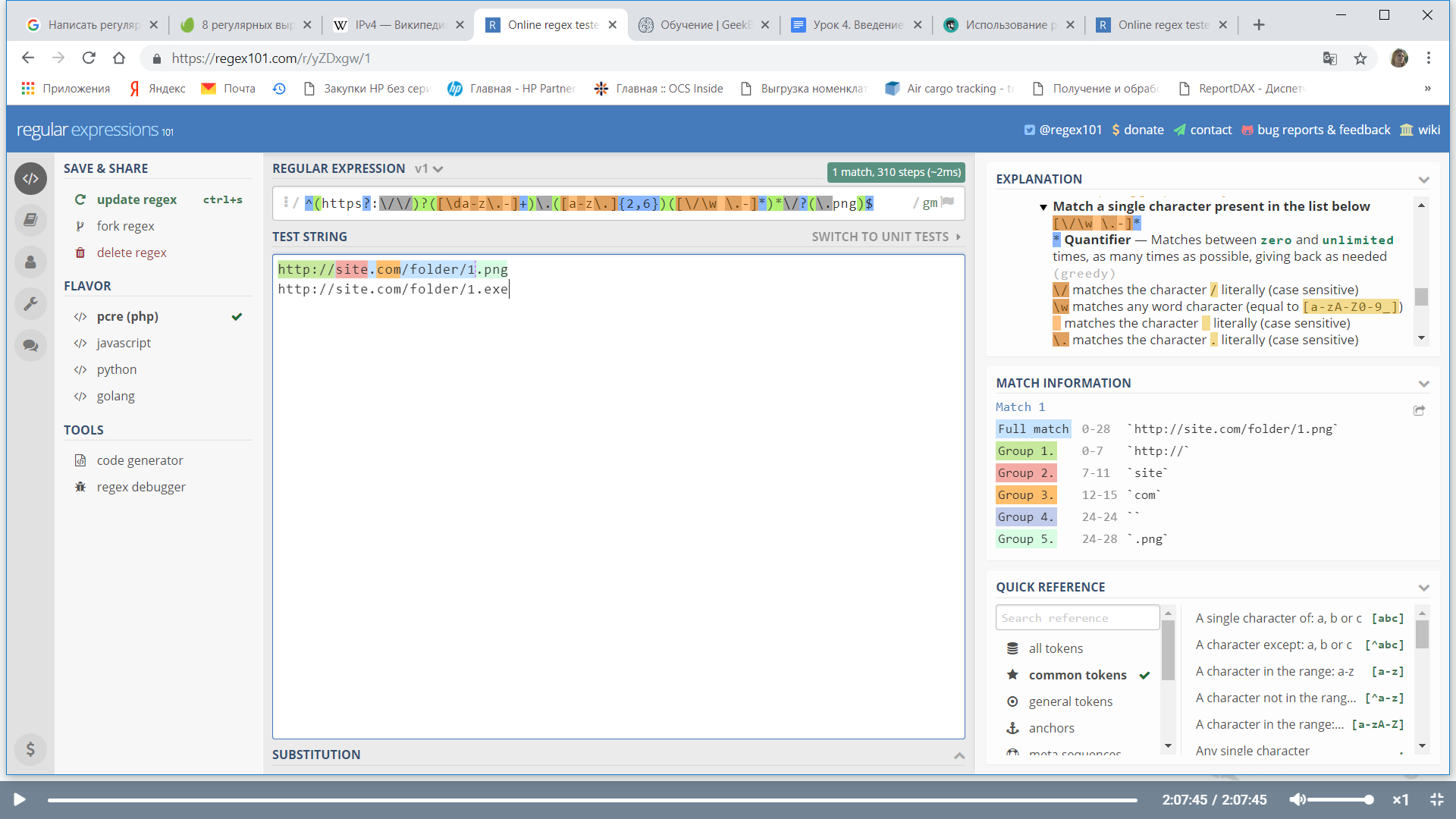
**Проверяла на соответствие .com и .png**

**.\*\.com|.\*\.png**



### **Написать регулярное выражение для проверки, ведет ли ссылка URL на некоторый файл, и это действительно ссылка на картинку (например,**[**http://site.com/folder/1.png**](http://site.com/folder/1.png)**), а не на любой файл.**

**^(https?:\/\/)?([\da-z\.-]+)\.([a-z\.]{2,6})([\/\w \.-]\*)\*\/?(\.png)$**



### **3. \*Написать регулярное выражение, которое проверяет, является выведенное значение «белым» IP-адресом (5.255.255.5 подойдет, а 172.16.0.1 — нет).**

### **4. \*Написать регулярное выражение, которое проверяет, что файл в URL (например,**[**https://site.ru/folder/download/test.docx**](https://site.ru/folder/download/test.docx)**) не обладает неким расширением (например .exe не пройдет, или .sh — не пройдет. Выбор списка исключенных расширений за вами).**