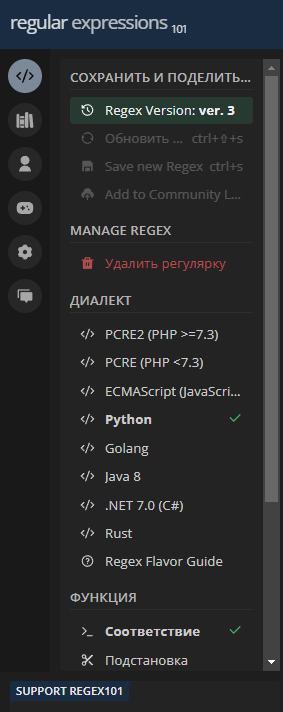
**Написание регулярных выражений**

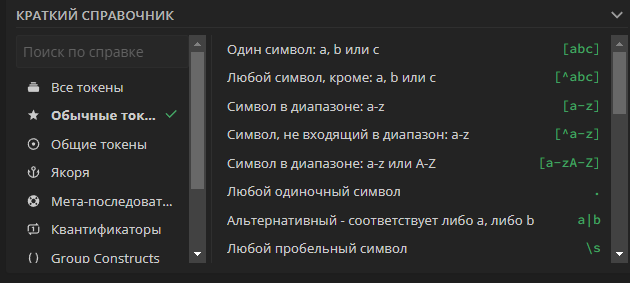


*Regular Expressions*

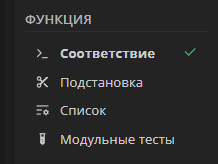
Для выполнения заданий необходимо зайти на сайт [regex101.com](https://regex101.com) (либо на любом другом аналоге написания регулярок), для того чтобы сохранить задания нужно нажать на «Сохранить и поделиться», каждое задание можно выполнять генерируя новую ссылку на него, таким образом и будем сохранять наши регулярки.



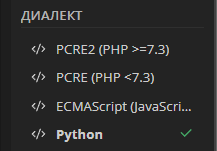
На самом сайте есть справа внизу краткий справочник на случай если вы забыли какие-либо из выражений



В функциях выбираем «Соответствие» (проверяем через «Подстановку». (Язык на русский можно поменять в настройках на левой панели)

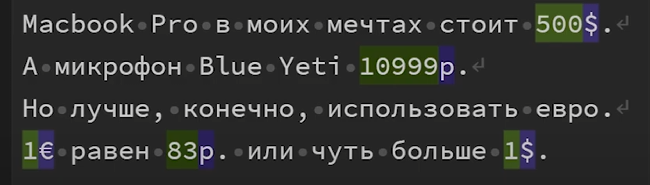


Диалект выбирает обязательно Python (мы для него ведь это пишем)



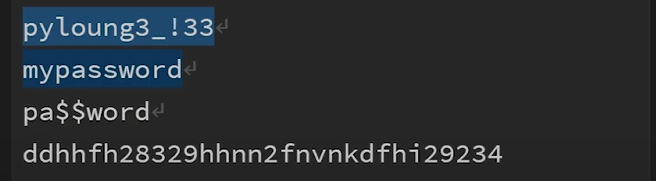
**Задание 1. Вывод цен. (regex)**

Снизу дан текст, вам необходимо написать регулярное выражение которое получит все цены из текста (забирая при этом и валюту (оформляем через |))



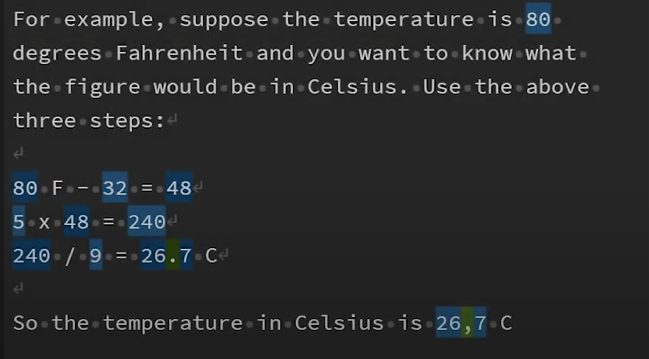
**Задание 2. Проверка надежности пароля. (regex)**

Напишите регулярное выражение проверяющее надежность пароля. Где возможен ввод только букв от a-Z, цифр от 0-9, символов от – до ? в минимальном диапазоне от 8 до 16 символов. Регулярка должна начинаться с якоря и заканчиваться им же. (^$)



**Задание 3. Поиск чисел. (regex)**

Необходимо создать регулярку ищущую все числа в тексте, включая и дробные (регулярка должна принимать дробное число и через запятую и через точку).



**Задание 4. Корректность ввода номера (regex)**

Напишите регулярное выражение которая выполняет примитивную проверку номера телефона на валидность. В начале и в конце используется якорь, так же не забываем что мы проверяем российский номер (!) поэтому первая проверка будет у нас на +7 либо 8.





**Задание 5. Проверка @email (regex)**

Создайте регулярное выражение которое проверяет правильность заполненного поля email (к примеру user@gmail.com)



**Задание 6 Конвертация валют (Python)**

На основе регулярки с задания 1 напишите программу которая выводит цены в долларах и заменяет их на рубли.

**Задание 7 Список чисел (Python)**

На основе регулярки из задания 3 создайте программу, которая принимает все числа и возвращает в виде списка объектов.

**Задание 8. Аутентификация пользователя (Python)**

После того как мы потренировались в создании регулярных выражений давайте напишем программу где будем использовать проверку email и правильность ввода пароля

Нужно реализовать механизм аутентификации пользователя с использования json-файла и регулярных выражений, начало системы будет выглядеть так:

class Menu() - этот класс отвечает за отображение приветствия пользователю после успешной авторизации

В классе присутствует одна функция:

def user\_message() для вывода пользователя сообщения об том что он успешно вошёл в систему

class Loader() – класс реализующий загрузку данных из json-файла и передающий их в класс Reg()

В классе присутствуют 3 функции:

def \_\_init\_\_ содержащий в себе message и filename,

def json\_load() содержащий в себе загрузку данных из json’a (with open)

def loader\_message забирающий из конструткора класса self.message и self.filename

class Reg() – класс, реализующий проверку введенных пользователем данных (email & password) через регулярные выражения.

В классе присутствуют следующие функции:

def \_\_init\_\_ забирающая строку data (чтобы передать данные из loader’a в reg

def reg\_try() являющейся функцией для того чтобы написать нашу аутентификацию проверяющую корректность ввода пользователя. Для этого используем два инпута (почта/пароль) и переберем всех пользователей из users json-файла, внутри него уже реализуем все наши проверки пользователей (по аналогии лабораторной по json’у).

В конце вызываем экземпляры класса внутри блока кода if \_\_name\_\_ == ‘\_\_main\_\_’:

print(‘ваши экземпляры классов)

Json-файл имеет такой вид:



Регулярные выражения имеют следующий вид:

Проверка пароля: Пароль должен быть не менее 8 символов и не более 16, может содержать латинские буквы (верхний и нижний регистр), цифры, символы "\_", "-", "!", "?"

Проверка почты: Email должен соответствовать стандартному формату (например, "user@example.com").

Дополнительные задания:

1. Напишите программу которая реализует проверку корректности ввода номера телефона при попытке указать его при аутентификации пользователя (через функции либо классы) (Python)
2. Напишите программу которая изменяет выделенный текст (через sub) в txt файле через ввод пользователя (Python)
3. Напишите регулярное выражение которое захватывает любую ссылку с большинством популярных доменов (http | https, com | ru), после выведите список объектов (хотя бы 3 разные ссылки) в терминале Python. (regex, Python)

