

Скраббл

В этом проекте вы будете разрабатывать приложение для подсчета очков в игре скраббл. Необходимо использовать словари для систематизации игроков, слов и очков.

1. Мы предоставили вам два списка, буквы и очки. Мы хотели бы объединить эти два в словарь, который бы сопоставил букву с количеством очков за ее использование.

Используя генерирование словарей и `zip`, создайте словарь с именем `letter_to_points`, который содержит элементы `letters` в качестве ключей и элементы `points` в качестве значений.

```
letters = ["A", "B", "C", "D", "E", "F", "G", "H", "I", "J", "K", "L", "M", "N", "O", "P", "Q", "R", "S", "T", "U", "V", "W", "X", "Y", "Z"]
points = [1, 3, 3, 2, 1, 4, 2, 4, 1, 8, 5, 1, 3, 4, 1, 3, 10, 1, 1, 1, 1, 4, 4, 8, 4, 10]
```

2. В нашем списке букв не учитывались пустые плитки. Добавьте в словарь `letter_to_points` элемент с ключом `" "` и значением 0 очков.
3. Мы хотим создать функцию, которая будет принимать слово и возвращать, сколько очков стоит это слово.
4. Определите функцию с именем `score_word`, которая принимает слово в качестве параметра.
5. Реализуйте в рамках функции алгоритм подсчета очков введенного слова.
6. Протестируйте полученную функцию с использованием `print`.
7. Создайте словарь с именем `player_to_words`, который сопоставляет игроков со списком слов, которые они составили в игре. Данные, которые необходимо внести в словарь:

player1	wordNerd	Lexi Con	Prof Reader
BLUE	EARTH	ERASER	ZAP
TENNIS	EYES	BELLY	COMA
EXIT	MACHINE	HUSKY	PERIOD

8. Создайте пустой словарь с именем `player_to_points`.
9. Переберите элементы в `player_to_words`. Назовите каждого игрока `player`, и каждый список слов `words`. В своем цикле создайте переменную с именем `player_points` и установите для нее значение 0.
10. Внутри цикла создайте еще один цикл, который проходит через каждое слово в `words` и добавляет значение `score_word()` со словом в качестве входных данных.
11. После завершения внутреннего цикла установите значение текущего игрока как ключ `player_to_points` со значением `player_points`.
12. `player_to_points` теперь должен содержать сопоставление игроков с количеством набранных ими очков. Выведите в консоль полученный список, чтобы увидеть текущее положение в этой игре!

Дополнительные задания:

Попробуйте реализовать некоторые из этих идей с помощью Python:

- `play_word()` — функция, которая принимает игрока и слово и добавляет это слово в список сыгранных им слов.
- `update_point_totals()` — обернуть ваши вложенные циклы в функцию, которую вы можете вызывать в любое время.
- сделать ваш словарь `letter_to_points` способным обрабатывать ввод в нижнем регистре