



– Snake case — стиль написания составных слов, при котором несколько слов разделяются символом подчеркивания (`_`), и не имеют пробелов в записи, причём каждое слово обычно пишется с маленькой буквы — «`foo_bar`», «`hello_world`» и т. д.

– CamelCase — стиль написания составных слов, при котором несколько слов пишутся слитно без пробелов, при этом каждое слово внутри фразы пишется с прописной буквы. Стиль получил название CamelCase, поскольку прописные буквы внутри слова напоминают горбы верблюда

4. Напишите функцию, которая вычисляет сверхурочную работу и оплату, связанную с сверхурочной работой.

Работа с 9 до 5: обычные часы работы

После 5 вечера это сверхурочная работа

Ваша функция получает массив с 4 значениями:

- Начало рабочего дня, в десятичном формате, (24-часовая дневная нотация)
- Конец рабочего дня. (Тот же формат)
- Почасовая ставка
- Множитель сверхурочных работ

Ваша функция должна возвращать:

\$ + заработанные в тот день (округлены до ближайшей сотой)

Пример:

```
overTime([9, 17, 30, 1.5]) → "$240.00"
```

```
overTime([16, 18, 30, 1.8]) → "$84.00"
```

```
overTime([13.25, 15, 30, 1.5]) → "$52.50"
```

Примечание:

С 16 до 17 регулярно, поэтому  $1 * 30 = 30$

С 17 до 18 сверхурочно, поэтому  $1 * 30 * 1,8 = 54$

$30 + 54 = 84,00$  \$

5. Индекс массы тела (ИМТ) определяется путем измерения вашего веса в килограммах и деления на квадрат вашего роста в метрах. Категории ИМТ таковы:

Недостаточный вес:  $<18,5$

Нормальный вес: 18.5-24.9

Избыточный вес: 25 и более

Создайте функцию, которая будет принимать вес и рост (в килограммах, фунтах, метрах или дюймах) и возвращать ИМТ и связанную с ним категорию. Округлите ИМТ до ближайшей десятой.

Пример:

```
BMI("205 pounds", "73 inches") → "27.0 Overweight"
```

```
BMI("55 kilos", "1.65 meters") → "20.2 Normal weight"
```

```
BMI("154 pounds", "2 meters") → "17.5 Underweight"
```

6. Создайте функцию, которая принимает число и возвращает его мультипликативное постоянство, которое представляет собой количество раз, которое вы должны умножать цифры в num, пока не достигнете одной цифры.

Пример:

```
bugger(39) → 3
// Because 3 * 9 = 27, 2 * 7 = 14, 1 * 4 = 4 and 4 has only one digit.

bugger(999) → 4
// Because 9 * 9 * 9 = 729, 7 * 2 * 9 = 126, 1 * 2 * 6 = 12, and
finally 1 * 2 = 2.

bugger(4) → 0
// Because 4 is already a one-digit number.
```

7. Напишите функцию, которая преобразует строку в звездную стенографию. Если символ повторяется n раз, преобразуйте его в символ\*n.

Пример:

```
toStarShorthand("abbccc") → "ab*2c*3"

toStarShorthand("77777geff") → "7*5gef*2"

toStarShorthand("abc") → "abc"

toStarShorthand("") → ""
```

8. Создайте функцию, которая возвращает true, если две строки рифмуются, и false в противном случае. Для целей этого упражнения две строки рифмуются, если последнее слово из каждого предложения содержит одни и те же гласные.

Пример:

```
doesRhyme("Sam I am!", "Green eggs and ham.") → true

doesRhyme("Sam I am!", "Green eggs and HAM.") → true
// Capitalization and punctuation should not matter.

doesRhyme("You are off to the races", "a splendid day.") → false

doesRhyme("and frequently do?", "you gotta move.") → false
```

Примечание:

- Без учета регистра.
- Здесь мы не обращаем внимания на такие случаи, как "thyme" и "lime".
- Мы также игнорируем такие случаи, как "away" и "today" (которые технически рифмуются, хотя и содержат разные гласные).

9. Создайте функцию, которая принимает два целых числа и возвращает true, если число повторяется три раза подряд в любом месте в num1 и то же самое число повторяется два раза подряд в num2.

Пример:

```
trouble(451999277, 41177722899) → True

trouble(1222345, 12345) → False

trouble(666789, 12345667) → True
```

```
trouble(33789, 12345337) → False
```

- 10.** Предположим, что пара одинаковых символов служит концами книги для всех символов между ними. Напишите функцию, которая возвращает общее количество уникальных символов (книг, так сказать) между всеми парами концов книги.

Эта функция будет выглядеть следующим образом:

```
countUniqueBooks("stringSequence", "bookEnd")
```

Пример:

```
countUniqueBooks("AZYWABBCATTTA", 'A') → 4
```

```
// 1st bookend group: "AZYWA" : 3 unique books: "Z", "Y", "W"
```

```
// 2nd bookend group: "ATTTA": 1 unique book: "T"
```

```
countUniqueBooks("$AA$BBCATT$C$$B$", '$') → 3
```

```
countUniqueBooks("ZZABCDEF", 'Z') → 0
```