

Genel Kurallar:

***Kod pratiğini Google Colab üzerinden yapabilirsiniz. Google arama butonunda yazdığınızda direkt karşınıza çıkacaktır. Gmail hesabınız giriş yaparak ile kendi Colab hesabınızı oluşturabilirsiniz.**

***Aşağıda belirttiğimiz açıklamalarda, # işareti ile başlayan kısımlar yazmanız gereken kodu açıklamaktadır.**

6.Gün: String Yapıları ve Sözlükler

1. Aşağıdaki dizilerin uzunlukları nelerdir?

Aşağıdaki beş diziden her biri için, len() kullanmadan uzunlukları tahmin edin.

a = ""

b = "it's ok"

c = 'it\'s ok'

d = ""hey""

e = '\n'

2. “Veri bilimciler zamanlarının %80’ini veri temizlemekle, %20’sini de veri temizlemekten şikayet etmekle geçiriyorlar.” diye bir deyiş var. US zip kodu verisini temizlemeye yardımcı bir fonksiyon yazabilir misiniz? Bir dizi verildiğinde, bu dizinin geçerli bir zip kodunu temsil edip etmediği görülmelidir. Geçerli bir zip kodu tam olarak 5 digitten oluşan herhangi bir String ile mümkün olmaktadır.

def is_valid_zip(zip_code):

""" 5 digit li bir String kodu döndürün. """

pass

3. Bir araştırmacı binlerce haber makalesi toplamıştır ancak dikkatini belirli bir kelime içeren makaleler üzerinde yoğunlaştırmak istemektedir. Aşağıdaki fonksiyonu tamamlayarak, araştırmacının makale listesini filtrelemesine yardımcı olun.

- Anahtar kelime dizisinin, daha büyük bir kelimenin parçası olarak görüldüğü belgeleri eklemeyin. Örneğin, eğer araştırmacı “closed (kapalı)” anahtar kelimesine bakıyorsa, “enclosed (çevrelenmiş)” dizisini ekleyemezsiniz.

- Araştırmacı büyük harf/ küçük harf ayrımı yapmanızı istememektedir. Yani, "Closed the case." ifadesi, anahtar kelime "closed" olduğunda geçerli sayılacaktır.
- Nokta ve ya virgüllerin eşleşenleri etkilemesine izin vermeyin. Anahtar kelime "closed" olduğunda, "It is closed." ifadesi de aramaya dahil edilmektedir. Ancak başka tür noktalama işaretleri olmadığını varsayabilirsiniz.

```
def word_search(doc_list, keyword):
```

```
    """
```

Belgelerin ve bir anahtar kelimenin listesi alınır (her bir belge bir String dir). Dizin değerlerinin listesini, anahtar kelimeyi içeren tüm belgeler için orijinal listeye dönüştürür.

Example:

```
doc_list = ["The Learn Python Challenge Casino.", "They bought a car", "Casinoville"]
```

```
>>> word_search(doc_list, 'casino')
```

```
>>> [0]
```

```
    """
```

```
    pass
```

4. Şimdi de, araştırmacı, araştırmak için birden fazla anahtar kelime sağlamak istiyor. Yardım için aşağıdaki fonksiyonu tamamlayın.

(Bu fonksiyonu uygularken az önce yazmış olduğunuz word_search (kelime arama) fonksiyonunu kullanmanız önerilir. Kodu bu şekilde tekrar kullanmak, programlarınızı daha sağlam ve okunaklı hale getirmektedir – ve daha az yazdırır!)

```
def multi_word_search(doc_list, keywords):
```

"""Belgelerin ve bir anahtar kelimenin listesi alınır (her bir belge bir String dir). Her anahtarın bir anahtar kelime ve her değerin, o anahtar kelimeyi içeren belgelerin bir indeksler listesinin (doc_list) olduğu bir sözlüğe dönüştürür.

```
>>> doc_list = ["The Learn Python Challenge Casino.", "They bought a car and a casino", "Casinoville"]
```

```
>>> keywords = ['casino', 'they']
```

```
>>> multi_word_search(doc_list, keywords)
```

```
{'casino': [0, 1], 'they': [1]}
```

```
    """
```

```
    pass
```