1. Udemy.com (https://www.udemy.com/)
2. BigData University (https://cognitiveclass.ai/)
3. tuoitre.vn (https://tuoitre.vn/)
4. google.com (https://www.google.com/)
5. vnexpress.net (https://vnexpress.net/)

HV_Chapter2_Exercise1.ipynb

File Edit View Insert Runtime Tools Help Last saved at 9:11 PM

+ Code + Text

Chapter 2 - Exercise 1:

Tạo danh sách 5 website mà bạn thường xuyên truy cập kèm theo link của chúng

cú pháp để tạo liên kết trong Markdown

```
[nội dung mô tả](địa chỉ trang web)
```

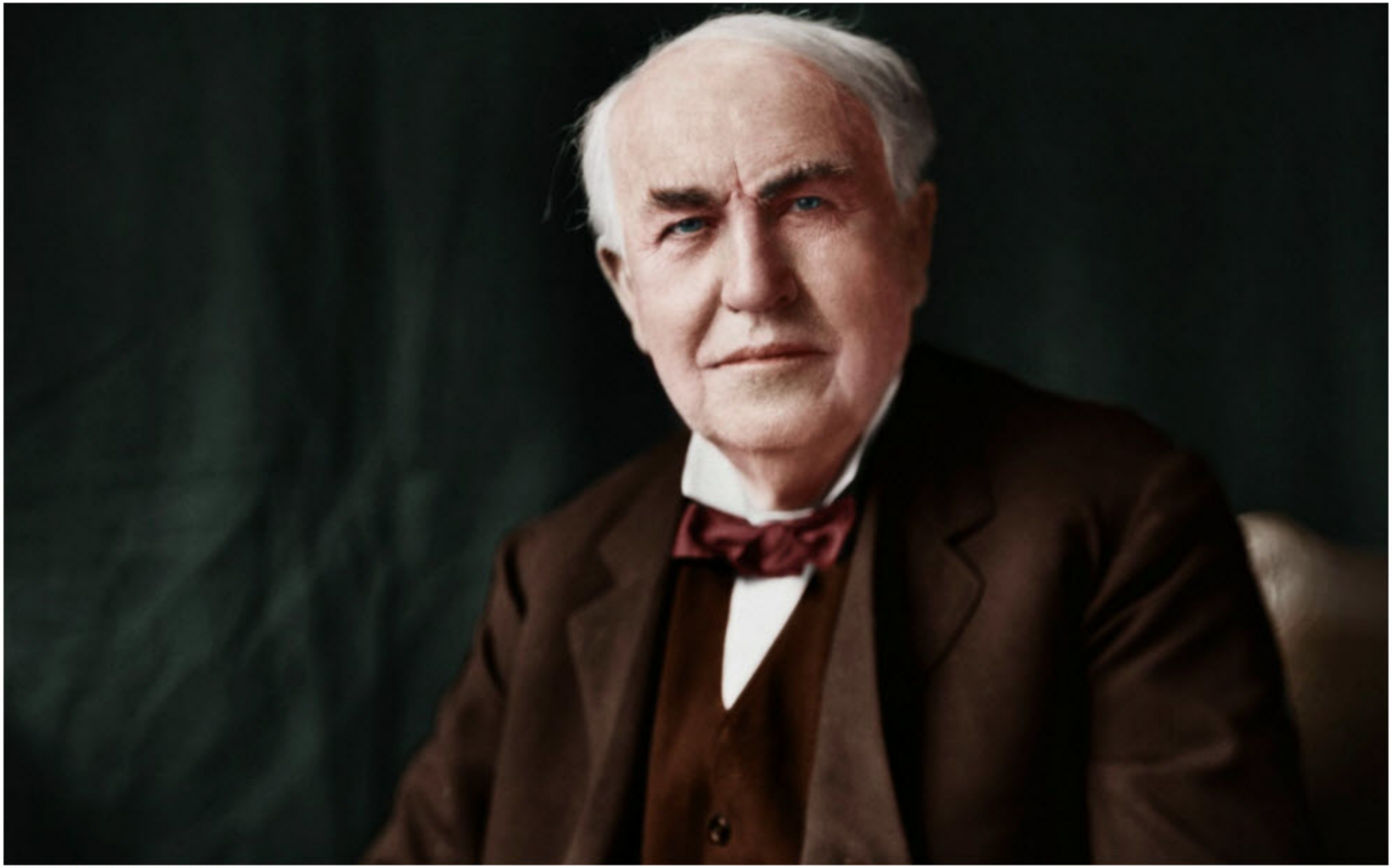
Sử dụng dữ liệu được cung cấp sau đây để tạo liên kết đến 5 website tương ứng:

```
1. [Udemy.com](https://www.udemy.com/)
2. [BigData University](https://cognitiveclass.ai/)
3. [tuoitre.vn](https://tuoitre.vn/)
4. [google.com](https://www.google.com/)
5. [vnexpress.net](https://vnexpress.net/)
```

1. Udemy.com
2. BigData University
3. tuoitre.vn
4. google.com
5. vnexpress.net

▶ Nhấn vào đây để xem kết quả!

Thomas Alva Edison (February 11, 1847 – October 18, 1931)



Visit Thomas Edison Wikipedia

- Born: Thomas Alva Edison, February 11, 1847, Milan, Ohio, U.S.
- Died: October 18, 1931 (aged 84), West Orange, New Jersey, U.S.
- Burial place: Thomas Edison National Historical Park, Nationality American
- Education: Self-educated
- Occupation: Inventor, businessman

HV_Chapter2_Exercise2.ipynb

File Edit View Insert Runtime Tools Help Last saved at 9:11 PM

+ Code + Text

Chapter 2 - Exercise 2: Viết tiểu sử ngắn về một nhà khoa học mà bạn yêu thích, có kèm hình ảnh của nhà khoa học và link đến trang Wikipedia của họ

cú pháp Markdown sử dụng trong bài tập

```
- Tạo liên kết: [nội dung mô tả](link trang web)
- Chèn hình ảnh: ![nội dung mô tả](link hình)
- Tạo danh sách:
   * nội dung 1
   * nội dung 2
   * nội dung 3
```

Sử dụng dữ liệu được cung cấp sau đây để mô tả tiểu sử của nhà khoa học Thomas Alva Edison:

```
## Thomas Alva Edison (February 11, 1847 – October 18, 1931)
![Thomas Edison](https://cdn-images-1.medium.com/max/1000/
1*s2GyMoLeSV3Epc2Gk3qBXA.png)
[Visit Thomas Edison Wikipedia](https://en.wikipedia.org/wiki/Thomas_Edison)
* Born: Thomas Alva Edison, February 11, 1847, Milan, Ohio, U.S.
* Died: October 18, 1931 (aged 84), West Orange, New Jersey, U.S.
* Burial place: Thomas Edison National Historical Park, Nationality American
* Education: Self-educated
* Occupation: Inventor, businessman
```

STT	Tên khóa học	Thời lượng
1	Fundamentals of Python	36
2	Python for Machine Learning, Data Science & Data Visualization	36
3	Maths & Statistic for Data Science	32
4	Databases & SQL for Data Science	32
5	Data Preprocessing & Analysis	40
6	Machine Learning with Python	48
7	R Programming Language for Data Science	48
8	Big Data in Machine Learning	36
9	Capstone Project	50

HV_Chapter2_Exercise3.ipynb

File Edit View Insert Runtime Tools Help Last saved at 9:10 PM

Comment Share

+ Code + Text Connect Editing

Sử dụng dữ liệu được cung cấp sau đây để tạo bảng danh sách các môn học:

```
STT - Tên khóa học - Thời lượng
1 Fundamentals of Python 36
2 Python for Machine Learning, Data Science & Data Visualization 36
3 Maths & Statistic for Data Science 32
4 Databases & SQL for Data Science 32
5 Data Preprocessing & Analysis 40
6 Machine Learning with Python 48
7 R Programming Language for Data Science 48
8 Big Data in Machine Learning 36
9 Capstone Project 50
```

▶ Nhấn vào đây để xem kết quả!

```
|STT|Tên khóa học|Thời lượng|
|:---|:---:|---:|
| 1 | Fundamentals of Python | 36 |
| 2 | Python for Machine Learning, Data Science & Data Visualization| 36 |
| 3 | Maths & Statistic for Data Science | 32 |
| 4 | Databases & SQL for Data Science | 32 |
| 5 | Data Preprocessing & Analysis | 40 |
| 6 | Machine Learning with Python | 48 |
| 7 | R Programming Language for Data Science | 48 |
| 8 | Big Data in Machine Learning | 36 |
| 9 | Capstone Project | 50 |
```

STT	Tên khóa học	Thời lượng
1	Fundamentals of Python	36
2	Python for Machine Learning, Data Science & Data Visualization	36
3	Maths & Statistic for Data Science	32
4	Databases & SQL for Data Science	32
5	Data Preprocessing & Analysis	40
6	Machine Learning with Python	48
7	R Programming Language for Data Science	48
8	Big Data in Machine Learning	36
9	Capstone Project	50

Linenear Regression

$$a = \frac{(\sum y)(\sum x^2)-(\sum x)(\sum xy)}{n(\sum x^2)-(\sum x)^2}$$

$$b = \frac{n(\sum xy)-(\sum x)(\sum y)}{n(\sum x^2)-(\sum x)^2}$$

Decision Tree

- Entropy using the frequency table of one attribute:

$$E(S) = \sum_{i=1}^{c} -p_i log_2 p_i$$

- Entropy using the frequency table of two attributes:

$$E(T, X) = \sum_{c \in X} P(c)E(c)$$

- Gain

$$Gain(T, X) = Entropy(T) - Entropy(E, X)$$

HV_Chapter2_Exercise4.ipynb

File Edit View Insert Runtime Tools Help Last saved at 10:27 PM

+ Code + Text

Chapter 2 - Exercise 4: Sử dụng Markdown để thực hiện các công thức

```
#### Cú pháp Markdown sử dụng trong bài tập:
$\sum$:              
\sum

$\sum_{i=1}^{10} t_i$:       \sum_{i=1}^{10} t_i

$e = \frac{a + b}{c + d}$:    e = \frac{a + b}{c + d}

$\in$:             
\in
```

Cú pháp Markdown sử dụng trong bài tập:

$\sum$: \sum

$\sum_{i=1}^{10} t_i$: \sum_{i=1}^{10} t_i

$e = \frac{a+b}{c+d}$: e = \frac{a + b}{c + d}

$\in$: \in

```
#### Linear Regression

$a=\frac{(\sum y)(\sum x^2)-(\sum x)(\sum xy)}{n(\sum x^2)-(\sum x^2)}$

$b=\frac{n(\sum xy)-(\sum x)(\sum xy)}{n(\sum x^2)-(\sum x)^2}$

#### Decision Tree
- Entropy using frequency table of one attribute:

${E(S)}=\sum_{i=1}^{C}-pilog_{2Pi}$

- Entropy using frequency table of two attribute:

${E(T,X)}=\sum_{c \in _{X}}P(c)E(c)$

- Gain

${Gain(T,X)}=Entropy(T) - Entropy(E,X)$
```

Linear Regression

$$a = \frac{(\sum y)(\sum x^2)-(\sum x)(\sum xy)}{n(\sum x^2)-(\sum x^2)}$$

$$b = \frac{n(\sum xy)-(\sum x)(\sum xy)}{n(\sum x^2)-(\sum x)^2}$$

Decision Tree

- Entropy using frequency table of one attribute:

$$E(S) = \sum_{i=1}^{C} -pilog_{2Pi}$$

- Entropy using frequency table of two attribute:

$$E(T,X) = \sum_{c \in X} P(c)E(c)$$

- Gain

$$Gain(T,X) = Entropy(T) - Entropy(E,X)$$

localhost:8888/notebooks/HV_Chapter3_Exercise1_a.ipynb

jupyter HV_Chapter3_Exercise1_a Last Checkpoint: a minute ago (unsaved changes)

Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

Run Code

Chapter 3 - Exercise 1a: Thực hiện các yêu cầu sau, và đối chiếu với kết quả cho trước

In [2]:
```
import numpy as np
```

In [7]:
```
# Câu 1: Tạo numpy array có giá trị từ 0-9 và lưu vào biến arr
arr = np.array([0,1,2,3,4,5,6,7,8,9])
# Hiển thị các phần tử có trong arr
print("Cac phan tu cua array:",arr)
# Xem kiểu dữ liệu (type) của arr
print(type(arr))
# Xem kích thước (shape) của arr
print("Kich thuoc shape: ",arr.shape)
```

```
Cac phan tu cua array: [0 1 2 3 4 5 6 7 8 9]
<class 'numpy.ndarray'>
Kich thuoc shape:  (10,)
```

Nhấn vào đây để xem kết quả!

In [11]:
```
# Câu 2: Từ array arr ở câu 1 => tạo arr_odd và arr_even
arr_odd = arr[arr%2!=0]
arr_even = arr[arr%2==0]
# Hiển thị các phần tử có trong arr_odd và arr_even
print("Ptu trong arr_odd:\t",arr_odd)
print("Ptu trong arr_even:\t",arr_even)
```

```
Ptu trong arr_odd:	[1 3 5 7 9]
Ptu trong arr_even:	[0 2 4 6 8]
```

Nhấn vào đây để xem kết quả!

In [25]:
```
# Câu 3: Từ array arr ở câu 1=> tạo arr_update_1 với các phần tử chẵn giữ nguyên, các phần tử lẻ thay bằng 100
arr_update = arr.copy()
arr_update[arr_update%2==1]=100

# Hiển thị các phần tử có trong arr_update_1
print("arr_updated", arr_update)
```

```
arr_updated [  0 100   2 100   4 100   6 100   8 100]
```

Nhấn vào đây để xem kết quả!

Chapter 3 - Exercise 1b: Thực hiện các yêu cầu sau, và đối chiếu với kết quả cho trước

In [3]:

```
import numpy as np

# Câu 1: Cho 2 array arr_a = [1,2,3,2,3,4,3,4,5,6] và arr_b = [7,2,10,2,7,4,9,4,9,8]
arr_a = [1,2,3,2,3,4,3,4,5,6]
arr_b = [7,2,10,2,7,4,9,4,9,8]

# Tạo arr_c chỉ lấy duy nhất các phần tử xuất hiện ở cả array arr_a và array arr_b
arr_c = np.intersect1d(arr_a,arr_b)
print(arr_c)
```

```
[2 4]
```

Nhấn vào đây để xem kết quả!

In [4]:

```
# Câu 2: Từ 2 array arr_a và arr_b ở câu 1 => Tạo array mới arr_d chứa các phần tử chỉ xuất hiện ở array arr_a
arr_d = np.setdiff1d(arr_a, arr_b)
print(arr_d)
```

```
[1 3 5 6]
```

Nhấn vào đây để xem kết quả!

In [5]:

```
# Câu 3: Cho array arr_e = np.array([2, 6, 1, 9, 10, 3, 27, 8, 6, 25, 16])
arr_e = np.array([2, 6, 1, 9, 10, 3, 27, 8, 6, 25, 16])
# Tạo array arr_f chỉ chứa các phần tử có giá trị từ 5 đến 10 của arr_e
arr_f = arr_e[arr_e>=5]
print("arr_f:",arr_f)
```

```
arr_f: [ 6  9 10 27  8  6 25 16]
```

Nhấn vào đây để xem kết quả!

numpy.intersect1d

`numpy.intersect1d(ar1, ar2, assume_unique=False, return_indices=False)` [source]

Find the intersection of two arrays.

Return the sorted, unique values that are in both of the input arrays.

Parameters:

ar1, ar2 : *array_like*

Input arrays. Will be flattened if not already 1D.

assume_unique : *bool*

If True, the input arrays are both assumed to be unique, which can speed up the calculation. If True but `ar1` or `ar2` are not unique, incorrect results and out-of-bounds indices could result. Default is False.

return_indices : *bool*

If True, the indices which correspond to the intersection of the two arrays are returned. The first instance of a value is used if there are multiple. Default is False. *New in version 1.15.0.*

Returns:

intersect1d : *ndarray*

Sorted 1D array of common and unique elements.

comm1 : *ndarray*

The indices of the first occurrences of the common values in *ar1*. Only provided if *return_indices* is True.

comm2 : *ndarray*

localhost:8888/notebooks/HV_Chapter3_Exercise1_c.ipynb

jupyter HV_Chapter3_Exercise1_c Last Checkpoint: 3 minutes ago (unsaved changes)

Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

Run Code

Chapter 3 - Exercise 1c: Thực hiện các yêu cầu sau, và đối chiếu với kết quả cho trước

In [2]:
```
import numpy as np
```

In [18]:
```
# Câu 1: Tạo array arr_zeros có 10 phần tử 0, cập nhật phần tử ở vị trí thứ 5 là 1
arr_zeros = np.zeros((10,),dtype=np.int)
print ("arr_zeros:",arr_zeros)
```

```
arr_zeros: [0 0 0 0 0 0 0 0 0 0]
```

```
C:\Users\Abc\AppData\Local\Temp/ipykernel_4804/2743076336.py:2: DeprecationWarning: `np.int` is a deprecated alias for the b
uiltin `int`. To silence this warning, use `int` by itself. Doing this will not modify any behavior and is safe. When replac
ing `np.int`, you may wish to use e.g. `np.int64` or `np.int32` to specify the precision. If you wish to review your current
use, check the release note link for additional information.
Deprecated in NumPy 1.20; for more details and guidance: https://numpy.org/devdocs/release/1.20.0-notes.html#deprecations
  arr_zeros = np.zeros((10,),dtype=np.int)
```

Nhấn vào đây để xem kết quả!

In [19]:
```
# Câu 2: Tạo và in array arr_h có giá trị từ 10 đến 24
arr_h = np.arange(10,24)
print("arr_h:",arr_h)

# In danh sách các phần tử theo tứ tự đảo ngược của arr_h vừa tạo
```

```
arr_h: [10 11 12 13 14 15 16 17 18 19 20 21 22 23]
```

Nhấn vào đây để xem kết quả!

In [20]:
```
# Câu 3: Cho array arr_k = np.array([1, 2, 0, 8, 2, 0, 1, 3, 0, 5, 0])
arr_k = np.array([1, 2, 0, 8, 2, 0, 1, 3, 0, 5, 0])

# Tạo array arr_l từ arr_k với các phần tử khác 0
arr_l = arr_k[arr_k!=0]
print("Arr_l:",arr_l)
```

```
Arr_l: [1 2 8 2 1 3 5]
```

ee?token=08c6b2b0885378ffdf0d7a06eb2a65ef49ed7b6e16... kết quả!

localhost:8888/notebooks/HV_Chapter3_Exercise1_c.ipynb

jupyter HV_Chapter3_Exercise1_c Last Checkpoint: 3 minutes ago (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

Run Code

In [20]:

```
# Câu 3: Cho array arr_k = np.array([1, 2, 0, 8, 2, 0, 1, 3, 0, 5, 0])
arr_k = np.array([1, 2, 0, 8, 2, 0, 1, 3, 0, 5, 0])

# Tạo array arr_l từ arr_k với các phần tử khác 0
arr_l = arr_k[arr_k!=0]
print("Arr_l:",arr_l)
```

```
Arr_l: [1 2 8 2 1 3 5]
```

Nhấn vào đây để xem kết quả!

In [21]:

```
# Câu 4: Từ array arr_l của câu 3, thêm 2 phần tử có giá trị là 10 và 20 vào cuối array
arr_l = np.append(arr_l,[10,20])
print("new arr_l:",arr_l)
```

```
new arr_l: [ 1  2  8  2  1  3  5 10 20]
```

Nhấn vào đây để xem kết quả!

In [22]:

```
# Câu 5: Từ array của câu 4, thêm phần tử có giá trị 100 vào vị trí có index = 5
arr_l = np.insert(arr_l,5,100)
print(arr_l)
```

```
[  1   2   8   2   1 100   3   5  10  20]
```

Nhấn vào đây để xem kết quả!

In [23]:

```
# Câu 6: Từ array của câu 5, xóa các phần tử tại vị trí có index = 0, 1, 2
lstIndex = [0,1,3]
arr_l = np.delete(arr_l,lstIndex)
print(arr_l)
```

```
[  8   1 100   3   5  10  20]
```

Nhấn vào đây để xem kết quả!