LESSON 1 : ÔN TẬP PYTHON

**- Deadline : *Giảng viên / Mentor set deadline và điền vào mục này trước khi gửi cho học viên***

**- Phương thức nộp bài : *Nêu rõ học viên sẽ gửi bài tập trực tiếp cho mentors hay gửi vào 1 folder đã được tạo sẵn. Nếu là folder tạo sẵn, vui lòng đính kèm link nộp bài vào đây !***

# Mục đích

*Bài tập này sẽ giúp các bạn củng cố kiến thức đã học, sau khi hoàn thiện các bài tập, các bạn có có thể đạt được :*

**Kỹ năng :**

* Xây dựng được các đoạn truy vấn Python để tạo ra các script mô tả object

**Kiến thức :**

Các định nghĩa, khái niệm sẽ được củng cố thông qua phần bài tập này, bao gồm :

* Variable
* List, Tuple, Dictionary
* Class, Object

# Bài tập

## **Bài tập 1** :

Bạn hãy xây dựng đoạn chương trình cho người dùng nhập số điểm GPA là số thập phân (với điều kiện >= 0 ) từ bàn phím. Nếu người dùng nhập số < 0, yêu cầu người dùng nhập lại.

Sau khi người dùng nhập xong, chương trình tính toán và trả về kết quả đánh giá học

Lực dựa trên GPA đó.

Biết rằng:

* Dưới 2 —- điểm F
* 2 đến 2.5 —- điểm C
* 2.5 đến 3 —- điểm B
* 3 đến 3.2 –– điểm B+
* 3.2 đến 3.7 —- điểm A
* Trên 3.7 — điểm A+

## Bài tập 2:

Bạn hãy xây dựng đoạn chương trình cho người dùng khởi tạo một danh sách OD chứa các dictionary có key là các mã hoá đơn và value là danh sách các sản phẩm được mua trong hoá đơn đó.

**Ví dụ dictionary của đơn hàng đầu tiên sẽ có dạng như sau :**

**{‘OD001’ : [‘Cá hồi’, ‘Sữa’, ‘Redbull’] }**

| OrderID | item1 | item2 | item3 |
| --- | --- | --- | --- |
| OD001 | Cá hồi | Sữa | Redbull |
| OD002 | Cá hồi | Sữa | Cá ngừ |
| OD003 | Bia | Sữa | Cá hồi |
| OD004 | Bia | Xúc xích |  |
| OD005 | Redbull |  |  |

1. Sau khi khởi tạo danh sách OD chứa các dictionary là các đơn hàng. Bạn hãy tìm ra đơn hàng mua ít sản phẩm nhất trong danh sách OD.
2. Bạn hãy tìm ra đơn hàng mua nhiều sản phẩm nhất trong danh sách OD.
3. Trong danh sách OD, bạn hãy tìm ra item nào được mua nhiều nhất

## Bài tập 3:

Tạo một class có tên là Person có các thuộc tính: Name, Address, DoB(Date of Birth) và các method cho phép khởi tạo object truyền vào các thuộc tính nêu trên.

## 

## 

## 

## 

## 

## 

## 

## 

## 

## Bài tập 4:

Tạo class có tên là Employee gồm các thuộc tính: EmID, Name, Address, DoB, Type(có 2 loại: Văn phòng, sản xuất), salary (lương căn bản) và các phương thức nhập, xuất thông tin của nhân viên đó.

Sau đó, khởi tạo 5 đối tượng nhân viên có các thông tin như bảng bên dưới và lưu trữ vào một list có tên là Employee\_Info:

| EmID | Name | Address | DoB | Type | Salary |
| --- | --- | --- | --- | --- | --- |
| I001 | Tú Anh | Vĩnh Phúc | 22/09/2000 | Văn phòng | 500.000 |
| I002 | Cường | Hưng Yên | 23/06/2000 | Sản xuất | 300.000 |
| I003 | Huyền | Hà Nội | 24/08/1999 | Văn phòng | 570.000 |
| I004 | Hoa | Hà Tĩnh | 27/12/1998 | Văn phòng | 700.000 |
| I005 | Phước | Nghệ An | 02/03/1995 | Sản xuất | 800.000 |

## Bài tập 5:

Giả sử Công ty có hai loại nhân viên: Nhân viên văn phòng và Nhân viên sản xuất. Viết hàm:

* Nhận vào thông tin là số ngày làm việc đối với nhân viên văn phòng
* Nhận vào thông tin là số ngày làm việc và số sản phẩm đã làm được đối với nhân viên sản xuất.

Sau đó tính lương cho từng nhân viên của công ty ở bảng bên trên và hiển thị ra màn hình kết quả có cấu trúc như sau:

* {Mã\_nhân\_viên} lương tháng này là {số tiền lương đã tính}

Biết rằng công ty cần tính lương cho nhân viên như sau:

* + Đối với nhân viên sản xuất:
    - Lương=lương căn bản \* số ngày làm việc + số sản phẩm \* 5.000
  + Đối nhân viên văn phòng:
    - Lương = lương căn bản \* số ngày làm việc + 2.000.000
* Kết quả minh hoạ: 

## Bài tập 6:

Xây dựng hàm Searching\_Employee nhận vào thông tin là mã nhân viên. Hàm sẽ tìm kiếm mã nhân viên đó trong danh sách Employee\_Info đã tạo ở bài 4. Nếu tìm thấy, hàm sẽ trả về toàn bộ thông tin của nhân viên đó. Nếu không tìm thấy, hiển thị thông báo Mã nhân viên vừa nhập không tồn tại.

**Bài tập 7:**

## Cài đặt Python trên máy tính cá nhân theo các tài liệu hướng dẫn ở [link](https://drive.google.com/drive/folders/1SutcJfd9rID3D8TnmoikQnFspvW-c4f-?usp=drive_link)