

1- May the force be with you

En aquest exercici, farem ús de programació orientada a objecte OOP:

- Definirem classes
- Definirem mètodes de classe
- Farem ús d'herència
- Crearem objectes
- Usarem la llibreria *json* per de fitxers JSON
- Editarem objectes usant els seus mètodes
- Exportarem objectes a fitxer JSON

Per aquest exercici, la nostra font de dades serà un fitxer JSON sobre els personatges de Star Wars. El trobaràs a *sallenet* en una carpeta.

Un fitxer JSON és un fitxer de marques per a l'internanvi d'informació equivalent a XML. Si vols més info sobre què és el format de fitxers JSON, et recomano mirar aquest tutorial: https://www.w3schools.com/js/js_json_intro.asp

Si explorem el fitxer, veurem que té la següent estructura:

```
{
  "fields": {
    "edited": "2014-12-20T21:17:56.891Z",
    "name": "Luke Skywalker",
    "created": "2014-12-09T13:50:51.644Z",
    "gender": "male",
    "skin_color": "fair",
    "hair_color": "blond",
    "height": "172",
    "eye_color": "blue",
    "mass": "77",
    "homeworld": 1,
    "birth_year": "19BBY"
  },
  "model": "resources.people",
  "pk": 1
}
```

Cada un d'aquests blocs de text enclausants entre { } són un personatge de Star Wars.

Obriu la consola de python i proveu de fer el següent:

```
>>> star_wars = {  
...   "fields": {  
...     "edited": "2014-12-20T21:17:56.891Z",  
...     "name": "Luke Skywalker",  
...     "created": "2014-12-09T13:50:51.644Z",  
...     "gender": "male",  
...     "skin_color": "fair",  
...     "hair_color": "blond",  
...     "height": "172",  
...     "eye_color": "blue",  
...     "mass": "77",  
...     "homeworld": 1,  
...     "birth_year": "19BBY"  
...   },  
...   "model": "resources.people",  
...   "pk": 1  
... }
```

Com podeu veure, hem creat una variable *star_wars* i li hem assignat el tros de fitxer anterior. Un cop fet, la variable conté l'informació del primer personatge de Star Wars. Per a saber quin és, accedim al seu nom de la següent manera:

```
>>> star_wars['fields']['name']  
'Luke Skywalker'
```

Aquesta manera d'accedir a les dades us sonarà dels diccionaris 😊.

Ara que hem explorat les dades, crearem una classe per a tal de poder guardar les dades de tots els personatges.

Primer de tot crearem un projecte nou de python on anirem afegint les notes classes. Cada classe serà un fitxer .py al mateix estil de java.

Per exemple, creo un fitxer nou que es dirà Alumne.py amb la classe Alumne:

```
>>> class Alumne:  
...     def __init__(self, nom, cognom):  
...         self.nom = nom  
...         self.cognom = cognom
```

Com pots observar, el mètode `__init__` és el constructor de l'objecte.

Seguidament, des del meu main.py creo un objecte de la classe:

```
>>> a1 = Alumne("Miquel","Montoro")
```

Finalment, imprimeixo el nom de l'alumne a1:

```
>>> a1.nom  
'Miquel'
```

Aquí trobaràs l'informació necessària sobre la sintaxis de python per a la creació de classes: https://www.w3schools.com/python/python_classes.asp

Per a guardar les dades dels personatges de Star Wars, necessitarem una classe **character.py** que tingui els següents paràmetres:

- edited
- name
- created
- gender
- skin_color
- hair_color
- height
- eye_color
- mass
- homeworld
- birth_year

Com podeu observar, del fitxer original de JSON, hem descartat les dades de *model* i *pk*.

Un cop tenim la classe **character** creada amb el seu constructor, crearem un getter per tal de que imprimeixi un text "maco" amb la info dels següents camps: *name*, *gender*, *homeworld* i *birth_year*.

L'objectiu és poder accedir a la informació d'aquests camps de forma individual usant mètodes privats de la classe de l'estil:

```
>>> c1.name  
"Luke"
```

Un cop tinguem la classe **character** acabada, crearem una classe extra anomenada **character_films.py** que serà herència de la classe **character**.

Aquesta classe **character_films.py** tindrà herederà els atributs i el constructor de **character** i a més a més tindrà els següents atributs propis extra:

- num_of_films
- first_film
- alive_at_the_end

Aquest tres atributs ens proporcionaran la info sobre el nombre de pel·lícules a les quals ha aparegut cada personatge, quin és el títol de la última pel·lícula on ha aparegut i si està viu al final de la saga o no.

A més a més, aquesta classe, tindrà tres mètodes de set per tal d'informar els camps nous creats: *num_of_films*, *first_film*, *alive_at_the_end*.

Per entendre com fer herència d'una classe a una altra, trobareu la info aquí:
https://www.w3schools.com/python/python_inheritance.asp

Un cop tinguem les dues classes creades, crearem un **main.py** per a fer de gestor del programa.

En aquest main, crearem una estructura de dades que ens permeti guardar tants objectes de tipus **character_film** com siguin necessaris. Com et pots imaginar, el més aconsellable seria una llista de tipus de dades **character_film** que anomenarem **character_film_list**. Pots trobar més info sobre estructures de llistes aquí:
https://www.w3schools.com/python/python_lists.asp

Ara, usarem la llibreria **json** per a llegir el fitxer de dades proporcionat i carregar cada un dels elements dins de la llista. Usa les funcions *append()*, *insert()*, *remove()*, *pop()* de les llistes per a inserir i eliminar elements.

Trobaràs més info sobre tractament de fitxers JSON en python aquí:

https://www.w3schools.com/python/python_json.asp

Un cop tinguem les dades carregades dis de la nostra llista, anirem a buscar els personatges 'Luke Skywalker', 'Chewbacca' i 'Anakin Skywalker' i els hi informarem les dades sobre *num_films*, *first_film*, *alive_at_the_end*. Aquesta info la podem trobar per internet fàcilment.

Per acabar, exportarem les dades de tots els personatges en un nou document JSON.