

Pràctica 7: JSON (part 1)

Treball realitzat per: Eric Fernández Rodríguez i Anguelina Levchenko

PART 1

Exercici 3. Respon les següents qüestions sobre la traducció de l'**exercici 1** (traducció JSON → XML).

- a. Com has manejat el valor *null* en l'element age de l'Anna en la traducció a XML? És aquesta la millor manera de representar la falta d'informació? Proposa totes les alternatives possibles.

Deixant la etiqueta sense valor.

- I. **etiqueta sense contingut:** `<age></age>`
- II. **etiqueta buida amb atribut:** `<age valor=""/>`

- b. Què haurem de tenir en compte quan tenim elements repetits com ara les mascotes o els amics? S'ha mantingut la consistència en la traducció?

Haurem de crear una altra etiqueta dins de mascotes i/o amics per poder representar cada mascota/amic i les dades que volem saber de cadascun.

Si, perquè tots els elements de les llistes de mascotes i amics consten al document.

Exercici 4. Respon les següents qüestions sobre la traducció de l'**exercici 2** (traducció XML → JSON).

- c. Explica què s'ha convertit en objectes, i què en arrays i per què has pres aquestes decisions.

S'ha convertit en objectes les dades de cada device perquè són característiques de cada device i en arrays els diversos devices perquè a l'xml apareixen diversos devices.

- d. Explica què has fet per tal de mantenir junta la informació del preu amb el tipus de moneda pagada. Com has transformat, en aquest cas, els atributs de l'XML a JSON i per què?

Ho hem posat dins d'una clau curva com a objecte.

- e. Hi ha alguna etiqueta en l'XML que no s'ha traduït directament a JSON? Creus que això significa que s'ha perdut informació?
 - i. **<Device> i <characteristic> . No afecta ja que les hem convertit en objecte i llista respectivament.**
- f. Com has gestionat els caràcters especials com les cometes dobles en la traducció? Com afecta això la llegibilitat del JSON?
 - i. **Fent servir scape chars ('\"'), és una modificació mínima que no afecta.**
- g. Explica com has tractat els elements sense informació o amb dades opcionals. Has optat per deixar el camp buit, per fer servir el valor *null* o per ometre el camp? Explica quina creus que és la millor decisió i per què.
 - i. **Hem creat una llista buida. Creiem que és la millor opció per tenir la estructura dissenyada per si s'ha d'afegir informació.**
- h. Quina estructura de dades has utilitzat per representar les característiques de "P50 Pocket"? Explica si hi ha alternatives i per què has pres aquesta decisió.
 - i. **Hem utilitzat una llista, ja que les claus (etiquetes) no es poden repetir dins d'un mateix objecte. La llista és la opció més òptima perquè si convertim les característiques en un objecte hauriem de escriure una clau per cada característica diferenciant-la cada vegada (característica_1, característica_2, ...).**
- i. Si el JSON resultant no té el camp "items_count", creus que s'ha perdut informació? Creus que és útil tenir aquesta informació en un camp?
 - i. **No, perquè llegint el JSON ja pots fer un compte del total. Pot ser útil com a eina visual a l'inici d'un JSON molt extens. Per això la hem deixat nosalt**

Exercici 6.

a)

```
fun getUnitatMesuraAltura (pokemon) {  
    return pokemon["altura"]["unitat"]  
}
```

b)

```
fun isSegonMovimentDeContacte (pokemon) {  
    segonMoviment = pokemon["moviments"][1]  
    es_contacte = segonMoviment["contacte"]  
    return es_contacte  
}
```

c)

```
fun getSumaEstadistiques (pokemon) {  
    suma = 0  
    stats = pokemon ["estadistiques"]  
    for stat in stats:  
        suma = suma + stat  
    return suma  
}
```

d)

```
fun getMitjanaEstadistiques (pokemon) {  
    mitjana= 0  
    stats = pokemon ["estadistiques"]  
    for stat in stats:  
        suma= suma + stat  
    mitjana= suma // (len(stats) +1)  
    return mitjana  
}
```

e)

```
fun getPes (Llista3Pokemon) {  
  
    primerPokemon = Llista3Pokemon[0]  
    segonPokemon = Llista3Pokemon[1]  
    tercerPokemon = Llista3Pokemon[2]  
  
    pes1 = primerPokemon["pes"]["quantitat"]  
    pes2 = segonPokemon["pes"]["quantitat"]  
  
    pes3 = tercerPokemon["pes"]["quantitat"]  
  
    return pes1+pes2+pes3  
}
```

f)

```
fun isEvolucióPossible (pokemon, nivell) {  
  
    nivell_evolucionar = pokemon["evolucions"][0]["nivell_per_evolucionar"]  
    if nivell == nivell_evolucionar:  
        return True  
    else:  
        return False  
}
```

g)

```
fun get_potencia_mes_alta(pokemonList) {  
  
    llista_potencies = []  
    for index in pokemonList:  
        suma = 0  
        moviments = pokemonList[index]["moviments"]  
        for moviment in moviments:  
            potencia = moviments[moviment]["potencia"]  
            suma += potencia  
        llista_potencies.append(suma)  
  
    return max(llibra_potencies)  
}
```