Index des sujets 2021

21-NSIJ1AN1 : Corrigé

Année : **2021**
Centre : **Amérique du nord**
Jour : **1**
Enoncé :

1. Exercice 1

bases de données relationnelles et langage SQL

1. a. Cette requête affiche les champs et `salle` et `marque_ordi` de la table `Ordinateur`.

   b. Cette requête affiche les champs `salle` et `marque_ordi` de la table `Ordinateur` pour les enregistrement dont le champ `video` est à `true`

2. 🗂️ **Requête SQL**

```
SELECT * FROM Ordinateur WHERE annee>=2017 ORDER BY annee ASC
```

> **Note**
>
> L'ordre croissant est l'ordre par défaut, on peut donc se passer du `ASC` dans la requête précédente. Pour mémoire l'ordre décroissant s'obtient avec `DESC`.

1. a. Une clé primaire est unique pour chaque enregistrement. Comme plusieurs ordinateurs peuvent être dans la même salle. Le champ salle n'est pas unique et ne peut donc pas servir de clé primaire.

   b. On souligne la clé primaire, on repère les clés étragères en les faisant précédées du symbole `#`.

2. Exercice 2

routage, processsus et systèmes sur puces

3. Exercice 3

tableaux et programmation de base en Python

1. a.

🐍 Script Python

```
def total_hors_reduction(tab):
    '''Calcul la somme des éléments de tab'''
    assert type(tab)==list, "L'argument doit être une liste"
    thr = 0
    for prix in tab:
        assert type(prix)==int or type(prix)==float,"Les prix doivent être de types numériques"
        thr += prix
    return thr
```

b.

🐍 Script Python

```
def offre_bienvenue(tab):
    """ tableau -> float """
    somme =0
    longueur = len(tab)
    if longueur > 0:
        somme = tab[0]* 0,8
    if longueur > 1:
        somme = somme + tab[1] * 0,7
    if longueur > 2:
        for i in range(2,longueur):
            somme = somme + tab[i]
    return somme
```

2. **🐍 Script Python**

```
def prix_solde(tab):
    if len(tab)>=5:
        return total_hors_reduction(tab)*0.5
    elif len(tab)==4:
        return total_hors_reduction(tab)*0.6
    elif len(tab)==3:
        return total_hors_reduction(tab)*0.7
    elif len(tab)==2:
        return total_hors_reduction(tab)*0.8
    else:
        return total_hors_reduction(tab)*0.9
```

3. a.

🐍 Script Python

```
def minimum(tab):
    min_courant = tab[0]
```

```
    for elt in tab:
        if elt<min_courant:
            min_courant = elt
    return min_courant
```

b.

🐍 **Script Python**

```
def offre_bon_client(tab):
    if len(tab)>=2:
        return total_hors_reduction(tab)-minimum(tab)
    else:
        return total_hors_reduction(tab)
```

4. a.

🐍 **Script Python**

```
tab = [35.0,30.5,20.0,15.0,10.5,5.0,6.0]
```

Le total des prix du panier est de \(35 + 30,5 + 20 + 15 + 10.5 + 6 + 5=122\). Compte tenu de l'ordre des articles les articles coutant 20 € et 5 € seront offerts. Et donc le prix à payer sera 97 €.

b.

🐍 **Script Python**

```
tab = [35.0,30.5,20.0,15.0,10.5,6.0,5.0]
```

Le prix total a payer est de 96 euros.

c. Il faut trier les objets par ordre décroissant de prix.

4. Exercice 4

arbres binaires et algorithmes associés

5. Exercice 5

notion de pile, de file et programmation de base en Python