

Requêtes SQL pour l'application «Showroom»

Profils et comptes :

1. Requêtes d'ajout des données d'un profil + compte associé

```
INSERT INTO t_compte_com (com_pseudo, com_mdp)
VALUES ('gestionnaire1', MD5('gesW0SH_!21'));

INSERT INTO t_profil_pro (pro_nom, pro_prenom, pro_mail, pro_validite, pro_statut, pro_date, com_pseudo)
VALUES ('Philippe', 'Noé', 'noephilippe29@gmail.com', 'A', 'A', '2021-01-27', 'gestionnaire1');
```

2. Requête de vérification des données de connexion (pseudo et mot de passe), c'est à dire requête qui vérifie l'existence (ou non) du couple pseudo / mot de passe dans la base

```
SELECT COUNT(*) FROM t_compte_com
JOIN t_profil_pro USING(com_pseudo)
WHERE com_pseudo="Claire"
AND com_mdp = MD5('mdpclaire')
AND pro_validite ='A';
```

3. Requête de récupération de toutes les données d'un profil (dont on connaît le pseudo)

```
SELECT * FROM t_profil_pro WHERE com_pseudo ='Claire';
```

4. Requête permettant de connaître le statut d'un utilisateur dont on connaît le nom et le prénom

```
SELECT pro_statut FROM t_profil_pro
WHERE pro_nom ='Philippe'
AND pro_prenom ='Noé';
```

5. Requête de modification des données d'un profil (pseudo connu)

```
UPDATE t_profil_pro
SET pro_nom = 'newNom', pro_prenom = 'newPrenom', pro_mail = 'newmail@gmail.com'
WHERE com_pseudo= 'Nono';
```

6. Requête de mise à jour du mot de passe d'un compte (pseudo connu)

```
UPDATE t_compte_com
SET com_mdp = MD5('mdpjojo2')
WHERE com_pseudo = 'Jojo';
```

7. Requête listant toutes les données des profils + comptes associés

```
SELECT * FROM t_profil_pro  
JOIN t_compte_com USING(com_pseudo)  
WHERE com_pseudo = 'Jojo'
```

8. Requête de validation d'un profil (pseudo connu)

```
UPDATE t_profil_pro  
SET pro_validite = 'A'  
WHERE com_pseudo = 'Jojo';
```

9. Requête de désactivation d'un profil (pseudo connu)

```
UPDATE t_profil_pro  
SET pro_validite = 'D'  
WHERE com_pseudo = 'Jojo';
```

Présentation :

10. Requête d'ajout des informations de la structure

```
INSERT INTO t_presentation_pre (pre_nomStruct, pre_adresse, pre_adresseMail, pre_numeroTel,
pre_horaireOuverture, pre_texte, `com_pseudo`)
VALUES ('Galerie Photo', '1 rue des lilas - 29200 - Brest', 'galeriephotos@gmail.com', '0733894383',
'8h', 'Exposition de photos dans la belle ville de Brest', 'Claire');
```

11. Requête vérifiant qu'il n'y a qu'une seule ligne dans la table de gestion de la présentation

```
SELECT COUNT(*) FROM t_selection_sel
```

12. Requête donnant les informations sur la structure

```
SELECT pre_nomStruct, pre_adresse, pre_adresseMail, pre_numeroTel, pre_horaireOuverture,
pre_texte FROM t_presentation_pre
```

13. Requête de modification de l'adresse, du n° de téléphone et de l'adresse e-mail de la structure

```
UPDATE t_presentation_pre
SET pre_adresse = '3 rue de la poste', pre_adresseMail = 'new@gmail.com', pre_numeroTel = '078154165'
WHERE pre_numero=1
```

14. Requête de suppression de toutes les informations de la structure

```
DELETE FROM t_presentation_pre
```

Actualités :

15. Requête d'ajout d'une actualité

```
INSERT INTO t_actualite_actu (actu_titre, actu_texte, actu_date, actu_etat, com_pseudo)
VALUES ('La première actualité', 'Ceci est la première actualité de Jojo', CURDATE(), 'A', 'Jojo');
```

16. Requête donnant la dernière actualité ajoutée

```
SELECT actu_numero, actu_titre FROM t_actualite_actu
WHERE com_pseudo = 'Jojo'
ORDER BY actu_date DESC
LIMIT 1;
```

17. Requête listant toutes les actualités et leur auteur

```
SELECT actu_titre, actu_texte, actu_date, com_pseudo FROM t_actualite_actu
```

18. Requête listant les 5 dernières actualités ajoutées et leur auteur

```
SELECT actu_titre, actu_texte, actu_date, com_pseudo FROM t_actualite_actu
ORDER BY actu_date DESC
LIMIT 5;
```

19. Requête de modification d'une actualité

```
UPDATE t_actualite_actu
SET actu_titre = 'newTitre', actu_texte = 'newTexte'
WHERE actu_numero = 1;
```

20. Requête de suppression d'une actualité à partir de son ID (n°)

```
DELETE FROM t_actualite_actu WHERE actu_numero = 1
```

21. Requête de désactivation de toutes les actualités publiées avant une certaine date (archivage des actualités trop anciennes)

```
UPDATE t_actualite_actu
SET actu_etat = 'D'
WHERE YEAR(actu_date) < '2019'
```

Sélections / éléments :

22. Requête d'ajout d'une sélection

```
INSERT INTO t_selection_sel (sel_intitule, sel_texteIntro, sel_date, com_pseudo)
VALUES ('Paysage', 'Toute les photos de paysage', '2021-01-27', 'gestionnaire1');
```

23. Requête listant tous les éléments d'une sélection particulière

```
SELECT * FROM t_element_ele
JOIN tj_relie_rel USING(ele_numero)
JOIN t_selection_sel USING(sel_numero)
WHERE sel_intitule = 'Paysage'
```

24. Requête comptant le nombre de sélections qui existent dans la base de données

```
SELECT COUNT(sel_numero) FROM t_selection_sel
```

25. Requête listant toutes les sélections et leurs éléments éventuels (+ lien(s))

```
SELECT sel_intitule, ele_intitule, lie_numero FROM t_selection_sel
LEFT OUTER JOIN tj_relie_rel USING(sel_numero)
LEFT OUTER JOIN t_element_ele USING(ele_numero)
LEFT OUTER JOIN t_lien_lie USING(ele_numero);
```

26. Requête de modification d'une sélection

```
UPDATE t_selection_sel
SET sel_intitule = 'newIntitule', sel_texteIntro = 'newTexte'
WHERE com_pseudo = 'Jojo'
```

27. Requête(s) de suppression d'une sélection à partir de son identifiant (ID)

```
DELETE FROM tj_relie_rel WHERE sel_numero = 2;
DELETE FROM t_selection_sel WHERE sel_numero = 2;
```

28. Requête listant toutes les sélections qui n'ont pas d'élément associé

```
SELECT sel_numero FROM t_selection_sel
EXCEPT
SELECT sel_numero FROM tj_relie_rel
```

29. Requête récupérant toutes les données d'un élément dont on connaît l'identifiant (ID)

```
SELECT * FROM t_selection_sel
WHERE sel_numero = 2
```

30. Requête d'ajout d'un élément pour une sélection particulière (ID connu)

```
INSERT INTO t_element_ele (ele_intitule, ele_descriptif, ele_date, ele_fichierImage, ele_etat)
VALUES ('newTitre', 'newTexte', CURDATE(), 'newImg.jpg', 'A');

SET @id_ele=(SELECT MAX(ele_numero) FROM t_element_ele);

INSERT INTO tj_relie_rel (sel_numero, ele_numero) VALUES (2, @id_ele);
```

31. Requête(s) de suppression d'un élément particulier (ID connu)

```
DELETE FROM t_lien_lie
WHERE ele_numero = 1;

DELETE FROM tj_relie_rel
WHERE ele_numero = 1;

DELETE FROM t_element_ele
WHERE ele_numero = 1;
```

32. Requête donnant l'ID de l'élément suivant connaissant l'ID de l'élément actuel choisi dans une sélection d'ID connu (puis même chose pour l'élément précédent)

```
SELECT ele_numero FROM t_element_ele
JOIN tj_relie_rel USING(ele_numero)
JOIN t_selection_sel USING(sel_numero)
WHERE sel_numero = 1
AND ele_numero > 2
LIMIT 1
```

```
SELECT ele_numero FROM t_element_ele
JOIN tj_relie_rel USING(ele_numero)
JOIN t_selection_sel USING(sel_numero)
WHERE sel_numero = 1
AND ele_numero < 2
LIMIT 1
```

33. Requête de modification d'un élément d'une sélection particulière (ID connus)

```
UPDATE t_element_ele
JOIN tj_relie_rel USING(ele_numero)
SET ele_intitule = 'newIntitule'
WHERE sel_numero = 1
AND ele_numero = 2
```

34. Requête(s) de suppression de tous les éléments d'une sélection particulière (ID connu)

```
DELETE FROM tj_relie_rel WHERE sel_numero=5;
```

35. Requête de suppression de toutes les sélections n'ayant pas d'élément associé

```
DELETE FROM t_selection_sel
WHERE sel_numero NOT IN (SELECT sel_numero FROM tj_relie_rel);
```

36. Requête de désactivation (remise à l'état brouillon) d'un élément particulier (ID connu)

```
UPDATE t_element_ele  
SET ele_etat = 'D'  
WHERE ele_numero = 1
```

37. Requête cachant tous les éléments ajoutés par un utilisateur particulier dont on connaît le pseudo

```
UPDATE t_element_ele  
JOIN tj_relie_rel USING(ele_numero)  
JOIN t_selection_sel USING(sel_numero)  
SET ele_etat = 'D'  
WHERE com_pseudo = 'Jojo'
```

38. Requête qui récupère toutes les données associées (sans oublier les liens) à une sélection particulière dont on connaît l'identifiant (ID)

```
SELECT * FROM t_selection_sel  
LEFT OUTER JOIN tj_relie_rel USING(sel_numero)  
LEFT OUTER JOIN t_element_ele USING(ele_numero)  
LEFT OUTER JOIN t_lien_lie USING(ele_numero)  
WHERE sel_numero = 1
```

Liens :

39. Requête ajoutant un lien pour l'élément choisi

```
INSERT INTO t_lien_lie (lie_titre, lie_url, lie_auteur, lie_date, ele_numero)
VALUES ('Article', 'https://photographie-bretagne.fr', 'Jean Michel ', '2021-02-01', '4');
```

40. Requête de suppression d'un lien dont on connaît l'URL

```
DELETE FROM t_lien_lie
WHERE lie_url = 'https://photographie-bretagne.fr'
```

41. Requête listant toutes les URL des liens de la base, sans redondance

```
SELECT DISTINCT lie_url FROM t_lien_lie
```

42. Requête listant tous les éléments, leur sélection et les URL des liens, s'il y en a

```
SELECT ele_intitule, sel_intitule, lie_url FROM t_lien_lie
RIGHT OUTER JOIN t_element_ele USING(ele_numero)
JOIN tj_relie_rel USING(ele_numero)
JOIN t_selection_sel USING(sel_numero)
```

43. Requête listant les URL des liens associés à un élément dont on connaît l'identifiant (ID)

```
SELECT ele_intitule, lie_url FROM t_lien_lie
JOIN t_element_ele USING(ele_numero)
WHERE ele_numero = 1
```

44. Requête qui vérifie l'existence (ou non) d'une URL d'un lien parmi les URL qui existent

```
SELECT COUNT(lie_url) FROM t_lien_lie
WHERE lie_url = 'https://test.com'
```