TP1 - Partie 3 (10% de la note finale)

Comme chacun de vos projets sont différents, les instructions suivantes ne sont pas trop spécifiques.

Vous ferez 1 script qui contiendra 1 vue, 1 fonction scalaire, 2 procédures stockées et 2 déclencheurs.

Chacun des objets de bases de données demandés ci-dessous aura une partie des points alloués à :

- La pertinence de l’objet dans le cadre de votre projet,

- Leur réalisation (écriture du script) avec le respect des contraintes données

- à la démonstration que le script fonctionne bel et bien

|  |  |
| --- | --- |
| OBJET | Description |
| Une vue | La requête SELECT qui compose la vue doit avoir au moins 3 clauses parmi les suivantes :  • WHERE, GROUP BY, HAVING, INNER JOIN et ORDER BY.  Cette vue doit avoir été pensée pour représenter une requête un peu complexe, qui pourrait être utilisée fréquemment dans le contexte de votre projet.  Pour démontrer que la vue fonctionne bien, vous ferez une requête l’utilisant. |
|
| Une fonction scalaire | Une fonction qui retournera une valeur.  Pour démontrer que la fonction fonctionne bien, vous utiliserez cette fonction dans une requête SELECT.  Vous utiliserez AUSSI cette fonction pour mettre à jour un champ d’une de vos tables. (UPDATE)  Si vous n’avez pas présentement de champs dans vos tables pouvant être mis à jour grâce à votre fonction, faites un ALTER TABLE pour ajouter un tel champ dans une de vos tables. |
| Une procédure stockée qui sera une généralisation d’une de vos requêtes | À vous de choisir, tant que c’est une requête pertinente dans le cadre de votre projet. (Ajoutez-lui des paramètres pour la rendre plus flexible possible) |
| Une procédure stockée qui prendra en paramètre un intervalle de dates/valeurs | Si vous avez des dates dans votre projets, utilisez-en une pour faire une procédure stockée avec deux paramètres, dateDebut et dateFin, pour sortir des infos intéressantes entre ces deux dates.  Si ce n’est pas le cas, vous pouvez prendre un champ de type numérique pour faire une procédure stockée avec deux paramètres, valeurMin et valeurMax, pour sortir des infos intéressantes entre ces deux valeurs.  N'oubliez pas de démontrer que vos procédures fonctionnent avec des tests. |
| Un déclencheur AFTER de votre choix | L’important est que votre déclencheur réalise une opération **pertinente** en lien avec vos données. (Exemples : modification de valeur dérivée, archivage de données, etc.)  Les tests doivent couvrir tous les types d’actions possibles qui activent votre déclencheur (INSERT, UPDATE et / ou DELETE) Votre déclencheur doit fonctionner avec des instructions qui modifient une seule rangée ET plusieurs rangées. (Fournissez des tests détaillés qui le prouvent) |
| Un déclencheur INSTEAD OF de votre choix | L’important est que votre déclencheur réalise une opération **pertinente** en lien avec vos données. (Exemples : archivage de données (si vous ne l’avez pas fait dans l’autre déclencheur !), reproduction du comportement de ON DELETE CASCADE pour une table avec deux clés étrangères jumelles, etc.)  Les tests doivent couvrir tous les types d’actions possibles qui activent votre déclencheur (INSERT, UPDATE et / ou DELETE) Votre déclencheur doit fonctionner avec des instructions qui modifient une seule rangée ET plusieurs rangées. (Fournissez des tests détaillés qui le prouvent) |

Précision : Il est possible que vous deviez ajouter ou modifier des tables dans votre BD afin d’accommoder les procédures ou déclencheurs que vous ajouterez.

Grille d’évaluation

|  |  |  |
| --- | --- | --- |
| Objet | Critères | Points (sur 100) |
| Vue | - Pertinence de la vue dans le cadre de votre projet | 2 |
| - Réalisation (écriture du script) avec le respect des contraintes données | 5 |
| - Démonstration que le script fonctionne bel et bien (tests) | 3 |
| Fonction scalaire | - Pertinence de la fonction scalaire dans le cadre de votre projet | 2 |
| - Réalisation (écriture du script) avec le respect des contraintes données | 5 |
| - Démonstration que le script fonctionne bel et bien (tests) | 5 |
| Procédure qui est une généralisation d’une de vos requêtes | - Pertinence de la procédure dans le cadre de votre projet | 2 |
| - Réalisation (écriture du script) avec le respect des contraintes données | 5 |
| - Démonstration que le script fonctionne bel et bien (tests) | 3 |
| Procédure qui donne des infos entre 2 dates/valeurs | - Pertinence de la procédure dans le cadre de votre projet | 2 |
| - Réalisation (écriture du script) avec le respect des contraintes données | 5 |
| - Démonstration que le script fonctionne bel et bien (tests) | 3 |
| Déclencheur AFTER | - Pertinence du déclencheur dans le cadre de votre projet | 3 |
| - Réalisation (écriture du script) avec le respect des contraintes données | 15 |
| - Démonstration que le script fonctionne bel et bien (tests) | 12 |
| Déclencheur INSTEAD OF | - Pertinence du déclencheur dans le cadre de votre projet | 3 |
| - Réalisation (écriture du script) avec le respect des contraintes données | 15 |
| - Démonstration que le script fonctionne bel et bien (tests) | 12 |

Il y a beaucoup de points donnés pour les tests des déclencheurs car on veut qu’ils soient bien faits. Imaginez que vous avez des tables ayant un million d’enregistrements. On veut voir clairement si vos déclencheurs fonctionnent bien.