

Grille d'évaluation de la base de données			
Conception de la base de données	Excellent 2	Bien 1	Insuffisant 0
Choix approprié du type de données	Tous les champs ont un type approprié	Choix du type de données pouvant être optimisé	Choix inapproprié du type de données
Utilisation appropriée des contraintes	Toutes les contraintes nécessaires sont présentes	Utilisation minimale de contraintes. Plusieurs erreurs pourraient survenir lors de l'injection	Absence d'utilisation des contraintes
Identification pertinente des clefs primaires et secondaires	Toutes les tables ont une clé primaire et les clés secondaires établissent clairement les relations entre tables	Des clés primaires et secondaires sont présentes, mais parfois manquantes ou inappropriées	Absence de clefs primaires / secondaires
Division appropriée des tables	Les tables sont des entités thématiques et minimisent la redondance de l'information	La division des champs en tables cause certaines redondances parmi les données	La division des champs en table cause des associations et regroupements de champs de différentes natures (lien logique faible ou absent)
Scripts	3	2	1
Les requêtes SQL de création et d'injection des données sont optimales	Les requêtes de création des tables et d'injection des données sont claires, facilement interprétables et exactes	Les requêtes de création des tables et d'injection des données contiennent une imprécision	Les requêtes de création des tables ne donnent pas le résultat attendu
Bonnes pratiques de programmation	3	2	1
<p>Le code est structuré en étapes et est facile à lire (indentation, une commande par ligne)</p> <p>Le code est commenté</p> <p>Les variables et arguments sont nommés de façon explicite</p> <p>Les opérations qui doivent être répétées sont enchâssées dans des boucles</p>	<p>Le code est structuré en étapes claires et est facile à lire grâce à une indentation appropriée et à une organisation cohérente. Chaque commande est sur une ligne distincte.</p> <p>Les commentaires sont présents et fournissent des explications adéquates sur le fonctionnement du code, facilitant sa compréhension.</p> <p>Les variables et les arguments sont nommés de manière explicite, ce qui rend le code plus compréhensible pour les autres programmeurs.</p> <p>Les opérations répétées sont souvent enchâssées dans des boucles, réduisant ainsi la redondance de code et améliorant sa maintenabilité.</p>	<p>Le code est structuré en étapes, mais la lisibilité pourrait être améliorée avec une meilleure indentation et une organisation plus claire. Il y a une commande par ligne.</p> <p>Certains commentaires sont présents, mais ils ne sont pas toujours suffisamment détaillés pour expliquer pleinement le code.</p> <p>Les variables et les arguments sont nommés de manière relativement explicite, mais il y a place à amélioration pour une meilleure compréhension du code.</p> <p>Quelques opérations répétées sont enchâssées dans des boucles, mais cela n'est pas systématique.</p>	<p>Le code manque de structure en étapes et n'est pas facile à lire en raison d'une indentation et d'une organisation insuffisantes. Il peut y avoir plusieurs commandes par ligne.</p> <p>Les commentaires sont absents ou insuffisants pour expliquer le fonctionnement du code.</p> <p>Les variables et les arguments sont mal nommés, ce qui rend difficile la compréhension du code sans une analyse approfondie.</p> <p>Les opérations répétées ne sont pas enchâssées dans des boucles, ce qui entraîne une redondance inutile de code.</p>

