

Préparation des codingGame

dimanche 2 juin 2024

11:38

Java

dimanche 2 juin 2024

11:39

Angular

dimanche 2 juin 2024

11:39

1. Quelle est la principale caractéristique d'Angular ?
 - A) Il s'agit d'un langage de programmation
 - **B) C'est un framework JavaScript**
 - C) Il est principalement utilisé pour le développement de jeux vidéo
 2. Quel est le langage de programmation utilisé pour développer des applications Angular ?
 - A) Java
 - B) Python
 - C) **TypeScript**
 3. Quel fichier est utilisé pour définir les dépendances d'un projet Angular ?
 - **A) package.json**
 - B) index.html
 - C) app.module.ts
 4. Quel module permet de gérer le routage dans une application Angular ?
 - **A) RouterModule**
 - B) FormsModule
 - C) HttpClientModule
 5. Comment déclarez-vous un composant dans Angular ?
 - A) En l'ajoutant à la liste des déclarations dans AppModule
 - B) En utilisant la balise <component> dans le fichier HTML
 - **C) En important le composant dans le fichier TypeScript**
 6. Quel est le rôle du fichier `app.component.html` dans une application Angular ?
 - A) Il contient les styles globaux de l'application.
 - **B) Il définit la structure HTML du composant racine de l'application.**
 - C) Il contient la logique métier de l'application.
 7. Quelle directive Angular est utilisée pour lier des données du composant à la vue HTML ?
 - A) *ngIf
 - B) *ngFor
 - **C) [(ngModel)]**
 8. Quelle commande Angular CLI est utilisée pour générer un nouveau composant ?
 - A) ng serve
 - **B) ng generate component**
 - C) ng build
 9. Quel décorateur est utilisé pour définir les métadonnées d'un composant Angular ?
 - **A) @Component**
 - B) @Injectable
 - C) @NgModule
 10. Quelle est la meilleure pratique pour gérer les appels HTTP dans une application Angular ?
 - A) Les effectuer directement dans les composants
 - **B) Utiliser le service HttpClient**
 - C) Utiliser des promesses JavaScript
- D'accord ! Voici quelques autres questions à choix multiples (QCM) sur Angular :

11. Quelle directive Angular est utilisée pour lier des événements du DOM à des méthodes du composant ?
- A) *ngIf
 - B) *ngFor
 - **C) (click)**
12. Quelle commande Angular CLI est utilisée pour créer un nouveau service ?
- **A) ng generate service**
 - B) ng create service
 - C) ng new service
13. Quel fichier est utilisé pour définir les routes dans une application Angular ?
- A) app.component.ts
 - B) app.module.ts
 - **C) app-routing.module.ts**
14. Quelle est la fonctionnalité principale du service HttpClientModule dans Angular ?
- A) Gestion des routes
 - B) Gestion des formulaires
 - **C) Effectuer des requêtes HTTP**
15. Comment déclarez-vous un service dans le module racine d'une application Angular ?
- A) En l'ajoutant à la liste des déclarations dans AppModule
 - **B) En l'ajoutant à la liste des fournisseurs dans AppModule**
 - C) En l'important dans le fichier AppModule
16. Quelle est la fonctionnalité principale du module FormsModule dans Angular ?
- A) Gestion des routes
 - **B) Gestion des formulaires**
 - C) Gestion des services HTTP
17. Quelle directive est utilisée pour afficher une liste d'éléments dans un modèle Angular ?
- A) *ngIf
 - **B) *ngFor**
 - C) *ngSwitch
18. Comment pouvez-vous lier des propriétés d'un composant Angular à des attributs HTML ?
- **A) En utilisant la syntaxe [property]="valeur"**
 - B) En utilisant la syntaxe (event)="méthode()"
 - C) En utilisant la syntaxe {{expression}}
19. Quelle est la fonction principale du module CommonModule dans Angular ?
- **A) Il fournit des directives Angular de base comme *ngFor et *ngIf.**
 - B) Il fournit des fonctionnalités de gestion des formulaires.
 - C) Il fournit des services HTTP pour effectuer des requêtes.
20. Comment pouvez-vous ajouter une animation à un élément dans Angular ?
- A) En utilisant le module HttpClientModule
 - B) En ajoutant une directive *ngAnimate à l'élément
 - **C) En utilisant le module BrowserAnimationsModule**
21. Quelle est la fonction principale du décorateur `@NgModule` dans Angular ?
- A) Déclarer un composant Angular
 - **B) Déclarer un module Angular**
 - C) Déclarer un service Angular
22. Comment pouvez-vous passer des données d'un composant parent à un composant enfant dans

Angular ?

- **A) En utilisant les entrées (`@Input()`) dans le composant enfant**
- B) En utilisant les sorties (`@Output()`) dans le composant parent
- C) En utilisant les services Angular

23. Quelle est la fonctionnalité principale du module `BrowserModule` dans Angular ?

- A) Gérer les formulaires HTML
- **B) Gérer les interactions avec le navigateur**
- C) Gérer les requêtes HTTP

24. Quelle directive Angular est utilisée pour afficher ou masquer un élément en fonction d'une expression ?

- **A) *ngIf**
- B) *ngFor
- C) *ngSwitch

25. Comment pouvez-vous gérer les formulaires réactifs dans Angular ?

- **A) En utilisant la classe `FormGroup`**
- B) En utilisant la directive `ngModel`
- C) En utilisant la classe `HttpClient`

Bien sûr ! Voici 5 séries de 15 questions à choix multiples (QCM) sur Angular pour un total de 75 questions supplémentaires :

26. Quelle est la fonction principale du fichier `angular.json` dans un projet Angular ?

- A) Définir les dépendances du projet
- **B) Configurer les paramètres du projet**
- C) Déclarer les modules du projet

27. Quelle méthode est utilisée pour intercepter les requêtes HTTP dans Angular ?

- **A) HttpInterceptor**
- B) HttpRequest
- C) HttpClient

28. Quel décorateur est utilisé pour injecter un service dans un composant Angular ?

- **A) @Injectable**
- B) @Component
- C) @Inject

29. Quelle est la fonction principale du module `HttpModule` dans Angular ?

- A) Gérer les formulaires HTML
- **B) Gérer les interactions avec le serveur HTTP**
- C) Gérer les interactions avec le navigateur

30. Quelle directive est utilisée pour créer un formulaire réactif dans Angular ?

- A) ngForm
- B) ngModel
- **C) formGroup**

31. Quelle est la fonctionnalité principale du service `ActivatedRoute` dans Angular ?

- A) Gérer les routes dans l'application
- **B) Accéder aux informations de la route active**
- C) Gérer les animations dans l'application

32. Quelle méthode est utilisée pour souscrire aux modifications d'un formulaire réactif dans Angular ?

- **A) subscribe()**
- B) getValue()
- C) patchValue()

33. Quelle directive est utilisée pour créer un composant réactif dans Angular ?

- **A) *ngComponent**
- B) *ngIf
- C) *ngFor

34. Quelle fonctionnalité Angular permet de charger des modules de manière asynchrone ?

- **A) Lazy loading**
- B) Eager loading
- C) Preloading

35. Comment pouvez-vous utiliser des services dans Angular ?

- **A) En les important dans AppModule**
- B) En les déclarant dans AppComponent
- C) En les ajoutant à la liste des fournisseurs dans AppModule

36. Quelle méthode est utilisée pour rediriger l'utilisateur vers une autre page dans Angular ?

- A) router.redirect()
- **B) router.navigate()**
- C) router.go()

37. Quelle fonctionnalité Angular permet de partager des données entre des composants qui ne sont pas liés par une relation parent-enfant ?

- **A) Services**
- B) Observables
- C) NgRx

38. Quelle est la fonctionnalité principale du décorateur '@ViewChild' dans Angular ?

- A) Accéder à un élément du DOM dans un composant
- B) Injecter un service dans un composant
- **C) Accéder aux données d'un composant enfant**

39. Quelle méthode est utilisée pour valider un formulaire dans Angular ?

- A) validate()
- B) setValue()
- **C) updateValueAndValidity()**

40. Quelle fonctionnalité Angular permet de rendre les applications plus rapides en préchargeant les modules pendant le chargement initial ?

- A) Lazy loading
- B) Eager loading
- **C) Preloading**

41. Quel est le rôle du fichier 'tsconfig.json' dans un projet Angular ?

- **A) Configurer les paramètres de compilation TypeScript**
- B) Configurer les dépendances du projet
- C) Déclarer les composants du projet

42. Quelle méthode est utilisée pour envoyer des données à un serveur HTTP dans Angular ?

- A) get()
- **B) post()**
- C) put()

43. Quel décorateur est utilisé pour définir les métadonnées d'un service Angular ?

- A) @Service
 - **B) @Injectable**
 - C) @NgModule
44. Quelle est la fonction principale du module `FormsModule` dans Angular ?
- **A) Gérer les formulaires HTML**
 - B) Gérer les interactions avec le serveur HTTP
 - C) Gérer les animations dans l'application
45. Quelle directive est utilisée pour lier des événements DOM à des méthodes de composant dans Angular ?
- **A) (click)**
 - B) [value]
 - C) (model)
46. Quelle méthode est utilisée pour accéder aux paramètres de requête dans une URL dans Angular ?
- **A) route.queryParams**
 - B) route.params
 - C) route.snapshot
47. Quelle directive est utilisée pour lier des données du composant à des éléments HTML dans Angular ?
- A) *ngIf
 - B) *ngFor
 - **C) [(ngModel)]**
48. Quelle fonctionnalité Angular permet de rendre les applications plus rapides en précompilant les modèles pendant la construction ?
- **A) Ahead-of-Time (AOT) compilation**
 - B) Just-in-Time (JIT) compilation
 - C) Hot module reloading
49. Comment pouvez-vous déclarer un service dans Angular ?
- A) En l'ajoutant à la liste des déclarations dans AppModule
 - B) En l'important dans le fichier AppModule
 - **C) En l'ajoutant à la liste des fournisseurs dans AppModule**
50. Quelle méthode est utilisée pour récupérer les paramètres de route dynamiques dans Angular ?
- **A) route.params**
 - B) route.queryParams
 - C) route.snapshot
51. Quelle fonctionnalité Angular permet de partager des données entre des composants qui ne sont pas liés par une relation parent-enfant, mais par une architecture centralisée de gestion de l'état ?
- A) Services
 - B) Observables
 - **C) NgRx**
52. Quelle est la fonctionnalité principale du décorateur `@ContentChild` dans Angular ?
- **A) Accéder à un élément du DOM dans un composant**
 - B) Injecter un service dans un composant
 - C) Acc
53. Quelle est la fonctionnalité principale du fichier `angular-cli.json` dans un projet Angular ?
- A) Configurer les paramètres de compilation TypeScript
 - B) Configurer les dépendances du projet

- **C) Déclarer les composants du projet**

54. Quel service Angular est utilisé pour effectuer des requêtes HTTP ?

- **A) HttpClientService**
- B) HttpService
- C) HttpClient

55. Quel décorateur est utilisé pour injecter des dépendances dans un composant Angular ?

- A) @Inject
- B) @Component
- **C) @Injectable**

56. Quelle est la fonction principale du module `HttpClientModule` dans Angular ?

- A) Gérer les formulaires HTML
- **B) Gérer les interactions avec le serveur HTTP**
- C) Gérer les animations dans l'application

Exercices Angular

lundi 3 juin 2024 09:14

Voici quelques exercices de codage Angular pour améliorer vos compétences :

Exercice 1 : Création d'un Composant

Objectif

Créer un composant Angular qui affiche une liste d'utilisateurs.

Étapes

1. Créez un nouveau composant `user-list`.
2. Ajoutez un tableau d'objets utilisateur dans le composant.
3. Affichez la liste des utilisateurs dans le template HTML du composant.

Solution

1. Créez le composant :

```
```sh
ng generate component user-list
```
```

2. Modifiez `user-list.component.ts` :

```
```typescript
import { Component } from '@angular/core';

@Component({
 selector: 'app-user-list',
 templateUrl: './user-list.component.html',
 styleUrls: ['./user-list.component.css']
})
export class UserListComponent {
 users = [
 { name: 'Alice', age: 25 },
 { name: 'Bob', age: 30 },
 { name: 'Charlie', age: 35 }
];
}
```
```

3. Modifiez `user-list.component.html` :

```
```html

 <li *ngFor="let user of users">
 {{ user.name }} ({{ user.age }} years old)


```
```

Exercice 2 : Services et Injection de Dépendances

Objectif

Créer un service Angular pour gérer les données des utilisateurs et l'injecter dans un composant.

Étapes

1. Créez un service `user`.
2. Déplacez la liste des utilisateurs du composant `user-list` vers le service `user`.

3. Injectez le service dans le composant `user-list` et utilisez-le pour obtenir les utilisateurs.

Solution

1. Créez le service :

```
```sh
ng generate service user
```
```

2. Modifiez `user.service.ts` :

```
```typescript
import { Injectable } from '@angular/core';

@Injectable({
 providedIn: 'root'
})
export class UserService {
 private users = [
 { name: 'Alice', age: 25 },
 { name: 'Bob', age: 30 },
 { name: 'Charlie', age: 35 }
];

 getUsers() {
 return this.users;
 }
}
```
```

3. Modifiez `user-list.component.ts` pour injecter le service :

```
```typescript
import { Component, OnInit } from '@angular/core';
import { UserService } from '../user.service';

@Component({
 selector: 'app-user-list',
 templateUrl: './user-list.component.html',
 styleUrls: ['./user-list.component.css']
})
export class UserListComponent implements OnInit {
 users: { name: string, age: number }[];

 constructor(private userService: UserService) {}

 ngOnInit() {
 this.users = this.userService.getUsers();
 }
}
```
```

Exercice 3 : Formulaires Réactifs

Objectif

Créer un formulaire réactif pour ajouter de nouveaux utilisateurs à la liste.

Étapes

1. Configurez un formulaire réactif dans `user-list.component.ts`.
2. Ajoutez un formulaire dans le template HTML pour ajouter de nouveaux utilisateurs.
3. Mettez à jour la liste des utilisateurs lorsque le formulaire est soumis.

Solution

1. Modifiez `user-list.component.ts` pour inclure le formulaire réactif :

```
``typescript
import { Component, OnInit } from '@angular/core';
import { FormBuilder, FormGroup } from '@angular/forms';
import { UserService } from '../user.service';

@Component({
  selector: 'app-user-list',
  templateUrl: './user-list.component.html',
  styleUrls: ['./user-list.component.css']
})
export class UserListComponent implements OnInit {
  users: { name: string, age: number }[];
  userForm: FormGroup;

  constructor(private userService: UserService, private fb: FormBuilder) {}

  ngOnInit() {
    this.users = this.userService.getUsers();
    this.userForm = this.fb.group({
      name: [''],
      age: ['']
    });
  }

  addUser() {
    const newUser = this.userForm.value;
    this.users.push(newUser);
    this.userForm.reset();
  }
}
...`
```

2. Modifiez `user-list.component.html` pour ajouter le formulaire :

```
``html
<form [formGroup]="userForm" (ngSubmit)="addUser()">
  <label for="name">Name:</label>
  <input id="name" formControlName="name" type="text">
  <label for="age">Age:</label>
  <input id="age" formControlName="age" type="number">
  <button type="submit">Add User</button>
</form>

<ul>
  <li *ngFor="let user of users">
    {{ user.name }} ({{ user.age }} years old)
  </li>
</ul>
...`
```

Exercice 4 : Utilisation de l'API HTTP

Objectif

Utiliser le service HTTP d'Angular pour récupérer des données utilisateur depuis une API REST et les afficher dans le composant `user-list`.

Étapes

1. Modifiez le service `user` pour récupérer les utilisateurs depuis une API.
2. Injectez le service HTTP dans le service `user`.
3. Modifiez le composant `user-list` pour afficher les utilisateurs récupérés depuis l'API.

Solution

1. Modifiez `user.service.ts` pour utiliser l'API HTTP :

```
``typescript
import { Injectable } from '@angular/core';
import { HttpClient } from '@angular/common/http';
import { Observable } from 'rxjs';

@Injectable({
  providedIn: 'root'
})
export class UserService {
  private apiUrl = 'https://api.example.com/users';

  constructor(private http: HttpClient) {}

  getUsers(): Observable<{ name: string, age: number }[]> {
    return this.http.get<{ name: string, age: number }[]>(this.apiUrl);
  }
}
...

```

2. Modifiez `user-list.component.ts` pour utiliser l'API :

```
``typescript
import { Component, OnInit } from '@angular/core';
import { UserService } from '../user.service';

@Component({
  selector: 'app-user-list',
  templateUrl: './user-list.component.html',
  styleUrls: ['./user-list.component.css']
})
export class UserListComponent implements OnInit {
  users: { name: string, age: number }[];

  constructor(private userService: UserService) {}

  ngOnInit() {
    this.userService.getUsers().subscribe(data => {
      this.users = data;
    });
  }
}
...

```

Ces exercices couvrent des concepts clés d'Angular, notamment les composants, les services, les formulaires réactifs et les appels HTTP. Ils devraient vous aider à renforcer vos compétences et à vous familiariser davantage avec le framework Angular.

Hibernate

dimanche 2 juin 2024

11:39

Hibernate medium

lundi 3 juin 2024 09:21

1. **Quelle annotation est utilisée pour marquer une classe comme une entité persistante dans Hibernate ?**
 - A) **@Entity**
 - B) @Table
 - C) @Persistent
 - D) @Data
2. **Quelle annotation est utilisée pour marquer une clé primaire dans Hibernate ?**
 - A) @PrimaryKey
 - B) @Key
 - C) **@Id**
 - D) @Primary
3. **Quelle annotation est utilisée pour définir une table en base de données dans Hibernate ?**
 - A) @Entity
 - B) **@Table**
 - C) @Column
 - D) @DatabaseTable
4. **Quelle annotation est utilisée pour définir une colonne dans Hibernate ?**
 - A) @Entity
 - B) @Table
 - C) **@Column**
 - D) @Field
5. **Quelle annotation est utilisée pour spécifier une relation "Many-to-One" dans Hibernate ?**
 - A) **@ManyToOne**
 - B) @OneToMany
 - C) @ManyToMany
 - D) @OneToOne
6. **Quelle annotation est utilisée pour spécifier une relation "One-to-Many" dans Hibernate ?**
 - A) @ManyToOne
 - B) **@OneToMany**
 - C) @ManyToMany
 - D) @OneToOne
7. **Quelle annotation est utilisée pour spécifier une relation "Many-to-Many" dans Hibernate ?**
 - A) @ManyToOne
 - B) @OneToMany
 - C) **@ManyToMany**
 - D) @OneToOne
8. **Quelle annotation est utilisée pour spécifier une relation "One-to-One" dans Hibernate ?**
 - A) @ManyToOne
 - B) @OneToMany
 - C) @ManyToMany
 - D) **@OneToOne**
9. **Quelle méthode est utilisée pour sauvegarder une entité dans la base de données en utilisant Hibernate ?**

- **A) save()**
 - B) persist()
 - C) store()
 - D) create()
10. **Quelle méthode est utilisée pour mettre à jour une entité dans la base de données en utilisant Hibernate ?**
- **A) update()**
 - B) modify()
 - C) change()
 - D) alter()
11. **Quelle méthode est utilisée pour supprimer une entité de la base de données en utilisant Hibernate ?**
- **A) delete()**
 - B) remove()
 - C) destroy()
 - D) discard()
12. **Quelle méthode est utilisée pour charger une entité de la base de données en utilisant Hibernate ?**
- A) get()
 - **B) load()**
 - C) fetch()
 - D) retrieve()
13. **Quelle annotation est utilisée pour spécifier une séquence de génération de clé primaire dans Hibernate ?**
- A) @Sequence
 - **B) @GeneratedValue**
 - C) @GenerationType
 - D) @KeySequence
14. **Quelle stratégie de génération de clé primaire est utilisée pour les bases de données qui supportent les séquences ?**
- A) GenerationType.AUTO
 - B) GenerationType.IDENTITY
 - **C) GenerationType.SEQUENCE**
 - D) GenerationType.TABLE
15. **Quelle méthode est utilisée pour ouvrir une session Hibernate ?**
- **A) openSession()**
 - B) createSession()
 - C) getSession()
 - D) startSession()
16. **Quelle méthode est utilisée pour fermer une session Hibernate ?**
- **A) close()**
 - B) stop()
 - C) end()
 - D) finish()
17. **Quelle méthode est utilisée pour démarrer une transaction Hibernate ?**
- **A) beginTransaction()**
 - B) startTransaction()
 - C) createTransaction()
 - D) openTransaction()

18. **Quelle méthode est utilisée pour valider une transaction Hibernate ?**
- **A) commit()**
 - B) confirm()
 - C) validate()
 - D) save()
19. **Quelle méthode est utilisée pour annuler une transaction Hibernate ?**
- **A) rollback()**
 - B) cancel()
 - C) undo()
 - D) revert()
20. **Quelle annotation est utilisée pour spécifier une colonne avec une valeur par défaut dans Hibernate ?**
- A) @Default
 - **B) @ColumnDefault**
 - C) @DefaultValue
 - D) @DefaultColumn
21. **Quelle méthode est utilisée pour exécuter une requête HQL dans Hibernate ?**
- A) createQuery()
 - **B) executeQuery()**
 - C) runQuery()
 - D) query()
22. **Quelle interface est utilisée pour définir des critères de recherche en Hibernate ?**
- A) CriteriaBuilder
 - B) Criteria
 - **C) CriteriaQuery**
 - D) SearchCriteria
23. **Quelle annotation est utilisée pour marquer un champ comme versionné pour l'optimistic locking en Hibernate ?**
- **A) @Version**
 - B) @OptimisticLock
 - C) @LockVersion
 - D) @Lock
24. **Quelle annotation est utilisée pour marquer un champ comme étant un ID composite dans Hibernate ?**
- A) @Compositeld
 - **B) @EmbeddedId**
 - C) @IdComposite
 - D) @PrimaryComposite
25. **Quelle annotation est utilisée pour spécifier l'ordre des éléments dans une relation "One-to-Many" ?**
- **A) @OrderBy**
 - B) @Order
 - C) @Sorted
 - D) @ArrangeBy
26. **Quelle annotation est utilisée pour spécifier une colonne qui est automatiquement mise à jour avec l'horodatage actuel ?**
- A) @CurrentTimestamp
 - **B) @UpdatedTimestamp**

- C) @Timestamp
- D) @UpdateTimestamp

27. **Quelle interface Hibernate est utilisée pour la pagination des résultats de requêtes ?**

- **A) ScrollableResults**
- B) QueryPagination
- C) PagedQuery
- D) PageableResults

28. **Quelle annotation est utilisée pour définir une clé étrangère dans Hibernate ?**

- A) @ForeignKey
- **B) @JoinColumn**
- C) @ForeignColumn
- D) @KeyReference

29. **Quelle méthode est utilisée pour rafraîchir l'état d'une entité à partir de la base de données dans Hibernate ?**

- **A) refresh()**
- B) reload()
- C) reset()
- D) renew()

30. **Quelle annotation est utilisée pour indiquer qu'un champ ne doit pas être persisté dans la base de données ?**

- **A) @Transient**
- B) @Ignore
- C) @NotPersisted
- D) @NonPersistent

Hibernate avancés

lundi 3 juin 2024 09:37

1. ****Quelle est la différence entre ``get()`` et ``load()`` dans Hibernate ?****
 - **A) ``get()`` renvoie null si l'objet n'est pas trouvé, tandis que ``load()`` lève une exception.**
 - B) ``load()`` renvoie null si l'objet n'est pas trouvé, tandis que ``get()`` lève une exception.
 - C) Ils ont le même comportement.
 - D) ``get()`` et ``load()`` sont obsolètes en Hibernate.
2. ****Qu'est-ce que le Lazy Loading dans Hibernate ?****
 - **A) Charger paresseusement les données uniquement lorsqu'elles sont demandées.**
 - B) Charger toutes les données en une seule fois.
 - C) Charger les données avant qu'elles ne soient demandées.
 - D) Charger les données uniquement lors de la mise à jour de l'objet.
3. ****Quelle est la signification de l'optimistic locking dans Hibernate ?****
 - A) Verrouillage de la base de données pour garantir la cohérence.
 - B) Verrouillage des enregistrements pendant la lecture pour éviter les conflits.
 - **C) Gestion des versions des enregistrements pour éviter les conflits lors des mises à jour.**
 - D) Utilisation de verrous exclusifs pour éviter les lectures concurrentes.
4. ****Qu'est-ce que l'Eager Loading dans Hibernate ?****
 - A) Charger paresseusement les données uniquement lorsqu'elles sont demandées.
 - **B) Charger toutes les données en une seule fois.**
 - C) Charger les données avant qu'elles ne soient demandées.
 - D) Charger les données uniquement lors de la mise à jour de l'objet.
5. ****Quelle est la différence entre ``save()`` et ``persist()`` dans Hibernate ?****
 - **A) ``save()`` renvoie l'ID de l'objet persisté, tandis que ``persist()`` ne le fait pas.**
 - B) ``persist()`` renvoie l'ID de l'objet persisté, tandis que ``save()`` ne le fait pas.
 - C) Ils ont le même comportement.
 - D) ``save()`` et ``persist()`` sont obsolètes en Hibernate.
6. ****Qu'est-ce que le Batch Processing dans Hibernate ?****
 - A) Traitement des données en une seule transaction.
 - **B) Traitement de plusieurs opérations en une seule requête.**
 - C) Traitement des données en plusieurs transactions séparées.
 - D) Traitement des données en utilisant plusieurs threads.
7. ****Quelle est la différence entre ``update()`` et ``merge()`` dans Hibernate ?****
 - A) ``update()`` met à jour l'objet détaché, tandis que ``merge()`` renvoie une nouvelle instance persistante.
 - **B) ``merge()`` met à jour l'objet détaché, tandis que ``update()`` renvoie une nouvelle instance persistante.**
 - C) Ils ont le même comportement.
 - D) ``update()`` et ``merge()`` sont obsolètes en Hibernate.
8. ****Qu'est-ce que le Criteria API dans Hibernate ?****
 - **A) Une API pour créer des critères de recherche de manière dynamique.**
 - B) Une API pour gérer les transactions dans Hibernate.
 - C) Une API pour interagir avec la base de données de manière asynchrone.
 - D) Une API pour gérer les verrous dans Hibernate.

9. ****Quelle est la signification de l'inverse dans une relation Hibernate ?****
- A) L'entité parente possède la relation.
 - **B) L'entité enfant possède la relation.**
 - C) L'entité parente ne possède pas la relation.
 - D) L'entité enfant ne possède pas la relation.
10. ****Quelle est la fonction de l'interface `Interceptor` dans Hibernate ?****
- **A) Pour intercepter et gérer les événements liés à la session et aux transactions.**
 - B) Pour définir des stratégies de génération de clés primaires.
 - C) Pour mapper les entités aux tables de base de données.
 - D) Pour gérer les requêtes SQL personnalisées.
11. ****Qu'est-ce que l'inverse dans une relation `@OneToMany` ou `@ManyToMany` en Hibernate ?**
- ****
- **A) La relation est bidirectionnelle.**
 - B) La relation est unidirectionnelle.
 - C) La relation est inversement proportionnelle.
 - D) La relation est aléatoire.
12. ****Quelle est la différence entre `session.get()` et `session.load()` dans Hibernate ?****
- **A) `get()` renvoie une instance proxy, tandis que `load()` renvoie l'instance réelle.**
 - B) `load()` renvoie une instance proxy, tandis que `get()` renvoie l'instance réelle.
 - C) Ils ont le même comportement.
 - D) `session.get()` et `session.load()` sont obsolètes en Hibernate.
13. ****Qu'est-ce que l'optimistic concurrency control dans Hibernate ?****
- A) Utilisation de verrous exclusifs pour éviter les lectures concurrentes.
 - **B) Gestion des conflits de mise à jour en vérifiant les versions des enregistrements.**
 - C) Utilisation de verrous partagés pour permettre les lectures concurrentes.
 - D) Gestion des conflits de mise à jour en verrouillant les enregistrements pendant la lecture.
14. ****Quelle est la fonction de l'interface `EntityInterceptor` dans Hibernate ?****
- **A) Pour intercepter et gérer les événements liés aux entités.**
 - B) Pour définir des stratégies de génération de clés primaires.
 - C) Pour mapper les entités aux tables de base de données.
 - D) Pour gérer les requêtes SQL personnalisées.
15. ****Qu'est-ce que le second-level cache dans Hibernate ?****
- A) Cache de niveau supérieur pour stocker les objets de session.
 - B) Cache de niveau inférieur pour stocker les résultats de requêtes.
 - **C) Cache de niveau application pour stocker les données partagées entre les sessions.**
 - D) Cache de niveau base de données pour stocker les requêtes SQL.
16. ****Qu'est-ce que le query cache dans Hibernate ?****
- A) Cache de niveau supérieur pour stocker les objets de session.
 - **B) Cache de niveau inférieur pour stocker les résultats de requêtes.**
 - C) Cache de niveau application pour stocker les données partagées entre les sessions.
 - D) Cache de niveau base de données pour stocker les requêtes SQL.
17. ****Quelle est la fonction de l'interface `UserType` dans Hibernate ?****
- **A) Pour définir des types de données personnalisés.**
 - B) Pour gérer les utilisateurs et les rôles dans l'application.
 - C) Pour mapper les entités aux tables de base de données.
 - D) Pour définir des stratégies de génération de clés primaires.
18. ****Qu'est-ce que l'association `@Embedded` dans Hibernate ?****
- A) Une relation entre deux entités.

- **B) Une classe imbriquée directement dans une autre classe.**
 - C) Une liaison entre les tables de base de données.
 - D) Une association entre les entités et les tables de jointure.
19. ****Quelle méthode est utilisée pour vider le cache de session en Hibernate ?****
- A) clear()
 - **B) evict()**
 - C) flush()
 - D) refresh()
20. ****Quelle méthode est utilisée pour rafraîchir une entité en Hibernate ?****
- A) clear()
 - B) evict()
 - **C) refresh()**
 - D) reload()
21. ****Quelle est la différence entre `saveOrUpdate()` et `persist()` dans Hibernate ?****
- **A) `saveOrUpdate()` met à jour l'objet s'il existe déjà en base de données, sinon il l'insère, tandis que `persist()` lève une exception si l'objet existe déjà.**
 - B) `persist()` met à jour l'objet s'il existe déjà en base de données, sinon il l'insère, tandis que `saveOrUpdate()` lève une exception si l'objet existe déjà.
 - C) Ils ont le même comportement.
 - D) `saveOrUpdate()` et `persist()` sont obsolètes en Hibernate.
22. ****Quelle est la fonction de l'interface `PostInsertEventListener` dans Hibernate ?****
- **A) Pour intercepter les événements après l'insertion d'une entité.**
 - B) Pour gérer les événements liés aux insertions de données.
 - C) Pour mapper les entités aux tables de base de données.
 - D) Pour définir des stratégies de génération de clés primaires.
23. ****Quelle est la signification de l'annotation `@NaturalId` dans Hibernate ?****
- **A) Pour spécifier un identifiant naturel dans une entité.**
 - B) Pour spécifier un identifiant artificiel dans une entité.
 - C) Pour marquer une propriété comme obligatoire.
 - D) Pour marquer une propriété comme facultative.
24. ****Qu'est-ce que l'association `@Formula` dans Hibernate ?****
- **A) Une formule utilisée pour calculer la valeur d'une colonne.**
 - B) Une association entre deux entités.
 - C) Une liaison entre les tables de base de données.
 - D) Une association entre les entités et les tables de jointure.
25. ****Quelle est la fonction de l'interface `BytecodeProvider` dans Hibernate ?****
- **A) Pour fournir le bytecode nécessaire à l'interprétation des entités.**
 - B) Pour gérer les transactions dans Hibernate.
 - C) Pour mapper les entités aux tables de base de données.
 - D) Pour définir des stratégies de génération de clés primaires.
26. ****Quelle est la signification de l'annotation `@Proxy` dans Hibernate ?****
- **A) Pour activer la création de proxy pour une classe.**
 - B) Pour désactiver la création de proxy pour une classe.
 - C) Pour mapper une classe à une table de base de données.
 - D) Pour spécifier les stratégies de génération de clés primaires.
27. ****Quelle est la fonction de l'interface `IdentifierGenerator` dans Hibernate ?****
- **A) Pour générer des identifiants uniques pour les entités.**
 - B) Pour gérer les transactions dans Hibernate.

- C) Pour mapper les entités aux tables de base de données.
 - D) Pour définir des stratégies de génération de clés primaires.
28. ****Quelle est la signification de l'annotation `@BatchSize` dans Hibernate ?****
- **A) Pour spécifier la taille du batch lors de l'exécution de requêtes.**
 - B) Pour spécifier la taille du batch lors de l'insertion d'entités.
 - C) Pour spécifier la taille du batch lors de la mise à jour d'entités.
 - D) Pour spécifier la taille du batch lors de la suppression d'entités.
29. ****Quelle est la fonction de l'interface `PostDeleteEventListener` dans Hibernate ?****
- **A) Pour intercepter les événements après la suppression d'une entité.**
 - B) Pour gérer les événements liés aux suppressions de données.
 - C) Pour mapper les entités aux tables de base de données.
 - D) Pour définir des stratégies de génération de clés primaires.
30. ****Quelle méthode est utilisée pour obtenir le cache de session en Hibernate ?****
- A) getSessionCache()
 - B) getCache()
 - **C) sessionFactory().getCache()**
 - D) getCurrentSession().getCache()
31. ****Qu'est-ce que l'association `@Filter` dans Hibernate ?****
- **A) Une annotation utilisée pour filtrer les résultats de requêtes.**
 - B) Une association entre deux entités.
 - C) Une liaison entre les tables de base de données.
 - D) Une annotation utilisée pour spécifier des filtres au niveau de l'entité.
32. ****Quelle est la signification de l'annotation `@Immutable` dans Hibernate ?****
- **A) Pour spécifier que l'entité est immuable et ne peut pas être modifiée.**
 - B) Pour spécifier que l'entité ne peut pas être chargée de manière paresseuse.
 - C) Pour spécifier que l'entité est nullable.
 - D) Pour spécifier que l'entité est non nullable.
33. ****Quelle est la fonction de l'interface `FlushEventListener` dans Hibernate ?****
- A) Pour intercepter les événements avant le flush de la session.
 - **B) Pour gérer les événements liés au flush de la session.**
 - C) Pour mapper les entités aux tables de base de données.
 - D) Pour définir des stratégies de génération de clés primaires.
34. ****Quelle est la signification de l'annotation `@EntityListeners` dans Hibernate ?****
- **A) Pour spécifier les écouteurs d'événements au niveau de l'entité.**
 - B) Pour spécifier les relations entre les entités.
 - C) Pour spécifier les écouteurs d'événements au niveau de la session.
 - D) Pour spécifier les écouteurs d'événements au niveau de la session factory.
35. ****Qu'est-ce que le lazy initialization exception dans Hibernate ?****
- **A) Une exception lancée lorsque Hibernate ne peut pas initialiser une relation paresseuse.**
 - B) Une exception lancée lorsque Hibernate ne peut pas initialiser une relation Eager.
 - C) Une exception lancée lorsque Hibernate ne peut pas initialiser une entité.
 - D) Une exception lancée lorsque Hibernate ne peut pas initialiser une collection.
36. ****Quelle est la fonction de l'interface `PostUpdateEventListener` dans Hibernate ?****
- **A) Pour intercepter les événements après la mise à jour d'une entité.**
 - B) Pour gérer les événements liés aux mises à jour de données.
 - C) Pour mapper les entités aux tables de base de données.
 - D) Pour définir des stratégies de génération de clés primaires.

37. ****Quelle est la signification de l'annotation `@Any` dans Hibernate ?****
- A) **Pour spécifier une association polymorphe.**
 - B) Pour spécifier une association unidirectionnelle.
 - C) Pour spécifier une association bidirectionnelle.
 - D) Pour spécifier une association many-to-many.
38. ****Quelle méthode est utilisée pour récupérer les métadonnées de session en Hibernate ?****
- A) getSessionMetadata()
 - B) getMetadata()
 - C) **getSessionFactory().getMetadata()**
 - D) getCurrentSession().getMetadata()
39. ****Quelle est la fonction de l'interface `PostCollectionUpdateEventListener` dans Hibernate ?****
- A) Pour intercepter les événements après la mise à jour d'une collection.
 - B) **Pour gérer les événements liés aux mises à jour de collections.**
 - C) Pour mapper les collections aux tables de base de données.
 - D) Pour définir des stratégies de génération de clés primaires.
40. ****Quelle est la signification de l'annotation `@BatchFetch` dans Hibernate ?****
- A) **Pour spécifier la taille du batch lors du chargement de collections.**
 - B) Pour spécifier la taille du batch lors du chargement d'entités.
 - C) Pour spécifier la taille du batch lors du chargement de résultats de requêtes.
 - D) Pour spécifier la taille du batch lors du chargement de données.
41. ****Quelle est la signification de l'annotation `@MappedSuperclass` dans Hibernate ?****
- A) **Pour spécifier une classe comme une superclasse mappée.**
 - B) Pour spécifier une classe comme une sous-classe mappée.
 - C) Pour spécifier une classe comme une entité.
 - D) Pour spécifier une classe comme une classe non mappée.
42. ****Qu'est-ce que l'association `@Any` dans Hibernate ?****
- A) Une annotation utilisée pour mapper une association many-to-many.
 - B) Une annotation utilisée pour mapper une association one-to-many.
 - C) **Une annotation utilisée pour mapper une association polymorphe.**
 - D) Une annotation utilisée pour mapper une association one-to-one.
43. ****Quelle est la fonction de l'interface `PostCollectionRecreateEventListener` dans Hibernate ?****
- A) **Pour intercepter les événements après la recreation d'une collection.**
 - B) Pour gérer les événements liés à la recreation de collections.
 - C) Pour mapper les collections aux tables de base de données.
 - D) Pour définir des stratégies de génération de clés primaires.
44. ****Quelle méthode est utilisée pour obtenir le dialecte SQL en Hibernate ?****
- A) getDialect()
 - B) getSessionDialect()
 - C) **getSessionFactory().getDialect()**
 - D) getCurrentSession().getDialect()

Spring

dimanche 2 juin 2024

11:39

Spring Meduim

lundi 3 juin 2024 11:04

SQL/PLSQL

dimanche 2 juin 2024

14:58

SQL Basics

dimanche 2 juin 2024

14:44

1. Quelle instruction SQL est utilisée pour sélectionner des données d'une table ?
 - A) FETCH
 - **B) SELECT**
 - C) RETRIEVE
2. Quelle clause est utilisée pour filtrer les résultats d'une requête SELECT en SQL ?
 - A) FILTER
 - **B) WHERE**
 - C) LIMIT
3. Quelle instruction SQL est utilisée pour insérer de nouvelles données dans une table ?
 - A) ADD
 - **B) INSERT INTO**
 - C) CREATE
4. Quelle clause SQL est utilisée pour trier les résultats d'une requête SELECT par ordre croissant ?
 - **A) ORDER BY ASC**
 - B) SORT BY
 - C) GROUP BY
5. Quelle instruction SQL est utilisée pour mettre à jour des données existantes dans une table ?
 - **A) UPDATE**
 - B) MODIFY
 - C) CHANGE
6. Quelle clause SQL est utilisée pour supprimer des données d'une table ?
 - A) ERASE
 - **B) DELETE FROM**
 - C) DROP
7. Quelle instruction SQL est utilisée pour créer une nouvelle table ?
 - **A) CREATE TABLE**
 - B) MAKE TABLE
 - C) NEW TABLE
8. Quelle clause SQL est utilisée pour regrouper les résultats d'une requête SELECT ?
 - **A) GROUP BY**
 - B) SORT BY
 - C) GROUP WITH
9. Quelle fonction SQL est utilisée pour compter le nombre de lignes dans un résultat de requête ?
 - **A) COUNT()**
 - B) SUM()
 - C) AVG()
10. Quelle instruction SQL est utilisée pour sélectionner des données distinctes d'une colonne ?
 - **A) DISTINCT**
 - B) UNIQUE
 - C) DIFFERENT

11. Quelle clause SQL est utilisée pour spécifier une condition logique dans une requête SELECT ?
- A) CONDITION
 - B) LOGIC
 - **C) WHERE**
12. Quelle instruction SQL est utilisée pour supprimer une table existante ?
- A) DELETE TABLE
 - **B) DROP TABLE**
 - C) REMOVE TABLE
13. Quelle fonction SQL est utilisée pour obtenir la valeur maximale d'une colonne ?
- **A) MAX()**
 - B) HIGHEST()
 - C) TOP()
14. Quelle clause SQL est utilisée pour limiter le nombre de résultats retournés par une requête SELECT ?
- A) TOP
 - **B) LIMIT**
 - C) FETCH FIRST
15. Quelle instruction SQL est utilisée pour sélectionner toutes les colonnes d'une table ?
- A) SELECT ALL
 - **B) SELECT ***
 - C) SELECT COLUMNS
16. Quelle fonction SQL est utilisée pour obtenir la valeur minimale d'une colonne ?
- **A) MIN()**
 - B) LOWEST()
 - C) BOTTOM()
17. Quelle clause SQL est utilisée pour combiner les résultats de deux requêtes SELECT ?
- A) MERGE
 - B) COMBINE
 - **C) UNION**
18. Quelle instruction SQL est utilisée pour créer une nouvelle base de données ?
- A) NEW DATABASE
 - **B) CREATE DATABASE**
 - C) MAKE DATABASE
19. Quelle clause SQL est utilisée pour spécifier une jointure entre deux tables ?
- **A) JOIN**
 - B) LINK
 - C) CONNECT
20. Quelle fonction SQL est utilisée pour obtenir la somme des valeurs d'une colonne ?
- A) ADD()
 - B) TOTAL()
 - **C) SUM()**
21. Quelle clause SQL est utilisée pour trier les résultats d'une requête SELECT par ordre décroissant ?
- **A) ORDER BY DESC**
 - B) SORT DESC
 - C) DESCENDING

22. Quelle instruction SQL est utilisée pour mettre à jour plusieurs colonnes dans une table ?
- A) MODIFY
 - **B) UPDATE**
 - C) CHANGE
23. Quelle clause SQL est utilisée pour spécifier une condition de jointure dans une requête SELECT ?
- A) JOIN
 - **B) ON**
 - C) WHERE
24. Quelle fonction SQL est utilisée pour obtenir la moyenne des valeurs d'une colonne ?
- A) MEAN()
 - B) AVERAGE()
 - **C) AVG()**
25. Quelle instruction SQL est utilisée pour sélectionner des données aléatoires dans une table ?
- A) SELECT RANDOM
 - B) SELECT RANDOM ROW
 - **C) SELECT RANDOM() FROM**
26. Quelle clause SQL est utilisée pour spécifier une condition de regroupement dans une requête SELECT ?
- **A) GROUP BY**
 - B) HAVING
 - C) FILTER BY
27. Quelle instruction SQL est utilisée pour limiter les résultats retournés par une requête SELECT à un nombre spécifique de lignes ?
- **A) LIMIT**
 - B) FETCH
 - C) TOP
28. Quelle clause SQL est utilisée pour spécifier une condition d'exclusion dans une requête SELECT ?
- **A) EXCEPT**
 - B) NOT
 - C) EXCLUDE
29. Quelle fonction SQL est utilisée pour arrondir une valeur numérique à l'entier le plus proche ?
- **A) ROUND()**
 - B) CEIL()
 - C) FLOOR()
30. Quelle instruction SQL est utilisée pour sélectionner une sous-ensemble de résultats d'une requête SELECT ?
- A) SUBSET
 - B) PART
 - **C) LIMIT**
31. Quelle clause SQL est utilisée pour spécifier une condition de tri dans une requête SELECT ?
- **A) ORDER BY**
 - B) SORT BY
 - C) ARRANGE BY
32. Quelle fonction SQL est utilisée pour extraire une sous-chaîne d'une valeur de colonne ?
- A) STR()
 - **B) SUBSTRING()**
 - C) EXTRACT()

33. Quelle instruction SQL est utilisée pour sélectionner des données d'une table et les trier dans un ordre aléatoire ?
- A) SELECT RANDOM
 - B) SELECT RANDOM ORDER
 - **C) SELECT ORDER BY RAND()**
34. Quelle clause SQL est utilisée pour spécifier une condition de regroupement dans une requête SELECT ?
- **A) GROUP BY**
 - B) HAVING
 - C) WHERE GROUP
35. Quelle instruction SQL est utilisée pour supprimer toutes les lignes d'une table ?
- A) DELETE ALL
 - B) DELETE
 - **C) TRUNCATE TABLE**
36. Quelle clause SQL est utilisée pour spécifier une condition d'inclusion dans une requête SELECT ?
- A) INCLUDE
 - **B) IN**
 - C) WHERE IN
37. Quelle instruction SQL est utilisée pour sélectionner les premiers N enregistrements dans une table ?
- A) SELECT TOP N
 - B) SELECT FIRST N
 - **C) SELECT LIMIT N**
38. Quelle fonction SQL est utilisée pour convertir une valeur de chaîne en majuscules ?
- **A) UPPER()**
 - B) UPPERCASE()
 - C) TOUPPER()
39. Quelle clause SQL est utilisée pour spécifier une condition d'union dans une requête SELECT ?
- **A) UNION**
 - B) AND
 - C) OR
40. Quelle instruction SQL est utilisée pour insérer plusieurs lignes de données dans une table à la fois ?
- A) INSERT INTO VALUES
 - B) INSERT INTO ROWS
 - **C) INSERT INTO SELECT**
41. Quelle fonction SQL est utilisée pour convertir une valeur de chaîne en minuscules ?
- **A) LOWER()**
 - B) LOWERCASE()
 - C) TOLOWER()
42. Quelle clause SQL est utilisée pour spécifier une condition d'exclusion dans une requête SELECT ?
- A) EXCLUDE
 - **B) NOT**
 - C) EXCEPT
43. Quelle instruction SQL est utilisée pour ajouter une nouvelle colonne à une table existante ?
- **A) ADD COLUMN**

- B) ALTER COLUMN
- C) MODIFY COLUMN

44. Quelle clause SQL est utilisée pour spécifier une condition d'ordre dans une requête SELECT ?

- A) SORT BY
- **B) ORDER BY**
- C) ARRANGE BY

45. Quelle instruction SQL est utilisée pour modifier la structure d'une table existante ?

- A) MODIFY TABLE
- B) CHANGE TABLE
- **C) ALTER TABLE**

46. Quelle clause SQL est utilisée pour spécifier une condition d'intersection dans une requête SELECT ?

- **A) INTERSECT**
- B) JOIN
- C) AND

47. Quelle instruction SQL est utilisée pour sélectionner une seule valeur à partir d'une table ?

- **A) SELECT VALUE**
- B) SELECT SINGLE
- C) SELECT ONE

48. Quelle clause SQL est utilisée pour spécifier une condition de négation dans une requête SELECT ?

- A) NEGATE
- **B) NOT**
- C) EXCLUDE

49. Quelle instruction SQL est utilisée pour sélectionner les derniers N enregistrements dans une table ?

- A) SELECT LAST N
- B) SELECT BOTTOM N
- **C) SELECT TOP N ORDER BY DESC**

50. Quelle fonction SQL est utilisée pour obtenir la date et l'heure actuelles ?

- A) NOW()
- **B) GETDATE()**
- C) CURRENT_TIMESTAMP()

SQL Medium

dimanche 2 juin 2024 14:57

1. Quelle clause SQL est utilisée pour spécifier une jointure externe dans une requête SELECT ?
 - A) INNER JOIN
 - **B) LEFT JOIN**
 - C) CROSS JOIN
2. Quelle instruction SQL est utilisée pour créer une vue dans une base de données ?
 - **A) CREATE VIEW**
 - B) MAKE VIEW
 - C) VIEW TABLE
3. Quelle fonction SQL est utilisée pour concaténer des chaînes de caractères ?
 - **A) CONCAT()**
 - B) JOIN()
 - C) MERGE()
4. Quelle clause SQL est utilisée pour spécifier une condition de jointure à l'aide de colonnes non équivalentes ?
 - A) JOIN
 - **B) ON**
 - C) USING
5. Quelle instruction SQL est utilisée pour supprimer une vue dans une base de données ?
 - A) DELETE VIEW
 - B) REMOVE VIEW
 - **C) DROP VIEW**
6. Quelle fonction SQL est utilisée pour obtenir la longueur d'une chaîne de caractères ?
 - A) LEN()
 - **B) LENGTH()**
 - C) SIZE()
7. Quelle clause SQL est utilisée pour spécifier une condition de jointure à l'aide de colonnes équivalentes ?
 - A) JOIN
 - B) ON
 - **C) USING**
8. Quelle instruction SQL est utilisée pour ajouter une contrainte UNIQUE à une colonne dans une table ?
 - A) ADD UNIQUE
 - B) UNIQUE COLUMN
 - **C) CONSTRAINT UNIQUE**
9. Quelle fonction SQL est utilisée pour obtenir la date actuelle ?
 - A) CURRENT_DATE()
 - **B) NOW()**
 - C) GETDATE()
10. Quelle clause SQL est utilisée pour trier les résultats d'une requête SELECT dans un ordre spécifié ?
 - A) SORT BY

- **B) ORDER BY**
- C) ARRANGE BY

11. Quelle instruction SQL est utilisée pour ajouter une contrainte de clé étrangère à une colonne dans une table ?

- **A) ADD FOREIGN KEY**
- B) FOREIGN COLUMN
- C) CONSTRAINT FOREIGN KEY

12. Quelle fonction SQL est utilisée pour obtenir la partie entière d'un nombre décimal ?

- **A) TRUNCATE()**
- B) ROUND()
- C) INTEGER()

13. Quelle clause SQL est utilisée pour spécifier une condition de jointure croisée entre deux tables ?

- A) JOIN
- **B) CROSS JOIN**
- C) INNER JOIN

14. Quelle instruction SQL est utilisée pour créer une index sur une colonne dans une table ?

- **A) CREATE INDEX**
- B) ADD INDEX
- C) INDEX COLUMN

15. Quelle fonction SQL est utilisée pour obtenir la date et l'heure actuelles ?

- A) CURRENT_TIMESTAMP()
- **B) NOW()**
- C) GETDATE()

16. Quelle clause SQL est utilisée pour spécifier une condition de jointure externe à droite dans une requête SELECT ?

- A) LEFT JOIN
- **B) RIGHT JOIN**
- C) OUTER JOIN

17. Quelle instruction SQL est utilisée pour modifier une vue existante dans une base de données ?

- A) MODIFY VIEW
- **B) ALTER VIEW**
- C) CHANGE VIEW

18. Quelle fonction SQL est utilisée pour obtenir le nombre d'éléments dans un groupe résultant d'une clause GROUP BY ?

- **A) COUNT()**
- B) GROUP_COUNT()
- C) GROUP_SIZE()

19. Quelle clause SQL est utilisée pour spécifier une condition de jointure externe complète dans une requête SELECT ?

- **A) FULL JOIN**
- B) OUTER JOIN
- C) COMPLETE JOIN

20. Quelle instruction SQL est utilisée pour ajouter une nouvelle colonne à une table existante en spécifiant sa position ?

- **A) ADD COLUMN**
- B) ALTER COLUMN
- C) INSERT COLUMN

21. Quelle fonction SQL est utilisée pour obtenir le reste d'une division entière ?
- A) REMAINDER()
 - **B) MOD()**
 - C) REM()
22. Quelle clause SQL est utilisée pour spécifier une condition de jointure naturelle dans une requête SELECT ?
- A) JOIN NATURAL
 - **B) NATURAL JOIN**
 - C) JOIN ON NATURAL
23. Quelle instruction SQL est utilisée pour renommer une table existante dans une base de données ?
- A) MODIFY TABLE
 - **B) RENAME TABLE**
 - C) CHANGE TABLE
24. Quelle fonction SQL est utilisée pour obtenir la position d'une sous-chaîne dans une chaîne de caractères ?
- A) SUBSTRING_INDEX()
 - B) POSITION()
 - **C) LOCATE()**
25. Quelle clause SQL est utilisée pour spécifier une condition de jointure externe à gauche dans une requête SELECT ?
- **A) LEFT JOIN**
 - B) RIGHT JOIN
 - C) OUTER JOIN
26. Quelle instruction SQL est utilisée pour supprimer une contrainte de clé primaire d'une colonne dans une table ?
- A) DELETE PRIMARY KEY
 - **B) DROP PRIMARY KEY**
 - C) REMOVE PRIMARY KEY
27. Quelle fonction SQL est utilisée pour obtenir la valeur maximale d'une colonne uniquement si elle satisfait une condition spécifiée ?
- **A) MAXIF()**
 - B) MAX()
 - C) GREATEST()
28. Quelle clause SQL est utilisée pour spécifier une condition de jointure interne dans une requête SELECT ?
- A) JOIN
 - **B) INNER JOIN**
 - C) CROSS JOIN
29. Quelle instruction SQL est utilisée pour créer une séquence dans une base de données ?
- **A) CREATE SEQUENCE**
 - B) MAKE SEQUENCE
 - C) SEQUENCE TABLE
30. Quelle fonction SQL est utilisée pour obtenir la date précédente à partir d'une date spécifiée ?
- A) DATE_BEFORE()
 - **B) PREVIOUS_DATE()**
 - C) DATEADD()

31. Quelle clause SQL est utilisée pour spécifier une condition de jointure externe complète dans une requête SELECT ?

- **A) FULL JOIN**
- B) OUTER JOIN
- C) COMPLETE JOIN

32. Quelle instruction SQL est utilisée pour ajouter une nouvelle colonne à une table existante en spécifiant sa position ?

- **A) ADD COLUMN**
- B) ALTER COLUMN
- C) INSERT COLUMN

33. Quelle fonction SQL est utilisée pour obtenir le reste d'une division entière ?

- A) REMAINDER()
- **B) MOD()**
- C) REM()

34. Quelle clause SQL est utilisée pour spécifier une condition de jointure naturelle dans une requête SELECT ?

- A) JOIN NATURAL
- **B) NATURAL JOIN**
- C) JOIN ON NATURAL

35. Quelle instruction SQL est utilisée pour renommer une table existante dans une base de données ?

- A) MODIFY TABLE
- **B) RENAME TABLE**
- C) CHANGE TABLE

36. Quelle fonction SQL est utilisée pour obtenir la position d'une sous-chaîne dans une chaîne de caractères ?

- A) SUBSTRING_INDEX()
- B) POSITION()
- **C) LOCATE()**

37. Quelle clause SQL est utilisée pour spécifier une condition de jointure externe à gauche dans une requête SELECT ?

- **A) LEFT JOIN**
- B) RIGHT JOIN
- C) OUTER JOIN

38. Quelle instruction SQL est utilisée pour supprimer une contrainte de clé primaire d'une colonne dans une table ?

- A) DELETE PRIMARY KEY
- **B) DROP PRIMARY KEY**
- C) REMOVE PRIMARY KEY

39. Quelle fonction SQL est utilisée pour obtenir la valeur maximale d'une colonne uniquement si elle satisfait une condition spécifiée ?

- **A) MAXIF()**
- B) MAX()
- C) GREATEST()

40. Quelle clause SQL est utilisée pour spécifier une condition de jointure interne dans une requête SELECT ?

- A) JOIN

- **B) INNER JOIN**
- C) CROSS JOIN

41. Quelle instruction SQL est utilisée pour créer une séquence dans une base de données ?

- **A) CREATE SEQUENCE**
- B) MAKE SEQUENCE
- C) SEQUENCE TABLE

42. Quelle fonction SQL est utilisée pour obtenir la date précédente à partir d'une date spécifiée ?

- A) DATE_PREVIOUS()
- **B) PREVIOUS_DATE()**
- C) DATEADD()

43. Quelle clause SQL est utilisée pour spécifier une condition de jointure externe à droite dans une requête SELECT ?

- A) LEFT JOIN
- **B) RIGHT JOIN**
- C) OUTER JOIN

44. Quelle instruction SQL est utilisée pour ajouter une contrainte de vérification à une colonne dans une table ?

- A) ADD CHECK
- **B) CONSTRAINT CHECK**
- C) CHECK COLUMN

45. Quelle fonction SQL est utilisée pour obtenir le nombre total de lignes dans une table ?

- **A) COUNT(*)**
- B) COUNT_ROWS()
- C) TOTAL_ROWS()

46. Quelle clause SQL est utilisée pour spécifier une condition de jointure externe complète dans une requête SELECT ?

- **A) FULL JOIN**
- B) OUTER JOIN
- C) COMPLETE JOIN

47. Quelle instruction SQL est utilisée pour ajouter une colonne à une table sans spécifier sa position ?

- **A) ADD COLUMN**
- B) ALTER COLUMN
- C) INSERT COLUMN

48. Quelle fonction SQL est utilisée pour obtenir le quotient d'une division entière ?

- A) QUOTIENT()
- **B) DIV()**
- C) INTEGER_DIVISION()

49. Quelle clause SQL est utilisée pour spécifier une condition de jointure naturelle dans une requête SELECT ?

- A) JOIN NATURAL
- **B) NATURAL JOIN**
- C) JOIN ON NATURAL

50. Quelle instruction SQL est utilisée pour renommer une colonne dans une table existante ?

- A) MODIFY COLUMN
- **B) RENAME COLUMN**
- C) CHANGE COLUMN

SQL Avancés

dimanche 2 juin 2024 15:14

1. Quelle clause SQL est utilisée pour spécifier une condition de jointure externe complète dans une requête SELECT ?

- A) FULL OUTER JOIN

2. Quelle fonction SQL est utilisée pour obtenir le nombre de jours entre deux dates ?

- A) DATEDIFF()

3. Quelle clause SQL est utilisée pour spécifier une condition de regroupement basée sur des expressions dans une requête SELECT ?

- C) GROUP BY

4. Quelle instruction SQL est utilisée pour renommer une colonne dans une table existante ?

- B) RENAME COLUMN

5. Quelle fonction SQL est utilisée pour générer une série de dates dans une plage spécifiée ?

- C) DATE_GENERATE_SERIES()

6. Quelle clause SQL est utilisée pour spécifier une condition de jointure basée sur une sous-requête dans une requête SELECT ?

- C) JOIN ON SUBQUERY

7. Quelle instruction SQL est utilisée pour ajouter une contrainte de vérification à une table existante ?

- B) ALTER TABLE ADD CHECK

8. Quelle fonction SQL est utilisée pour obtenir le nombre de mois entre deux dates ?

- A) MONTHS_BETWEEN()

9. Quelle clause SQL est utilisée pour spécifier une condition de jointure basée sur une expression de table commune (CTE) dans une requête SELECT ?

- C) JOIN ON CTE

10. Quelle instruction SQL est utilisée pour ajouter une colonne calculée à une table existante ?

- B) ADD COMPUTED COLUMN

11. Quelle fonction SQL est utilisée pour obtenir le rang d'une ligne dans un résultat trié ?

- A) ROW_NUMBER()

12. Quelle clause SQL est utilisée pour spécifier une condition de jointure basée sur une table dérivée dans une requête SELECT ?

- C) JOIN ON DERIVED TABLE

13. Quelle instruction SQL est utilisée pour ajouter une contrainte de clé étrangère à une colonne dans une table ?

- B) ALTER TABLE ADD FOREIGN KEY

14. Quelle fonction SQL est utilisée pour obtenir le rang d'une ligne dans un résultat trié, en gérant les valeurs ex aequo ?

- C) DENSE_RANK()

15. Quelle clause SQL est utilisée pour spécifier une condition de jointure basée sur un opérateur de

comparaison dans une requête SELECT ?

- B) JOIN ON

16. Quelle instruction SQL est utilisée pour ajouter une contrainte de clé primaire composée à une table ?

- C) ALTER TABLE ADD COMPOSITE PRIMARY KEY

17. Quelle fonction SQL est utilisée pour obtenir la liste des valeurs distinctes dans une colonne ?

- A) DISTINCT()

18. Quelle clause SQL est utilisée pour spécifier une condition de jointure basée sur une expression logique dans une requête SELECT ?

- C) JOIN ON LOGICAL

19. Quelle instruction SQL est utilisée pour ajouter une contrainte de vérification à une colonne dans une table ?

- B) ALTER COLUMN ADD CHECK CONSTRAINT

20. Quelle fonction SQL est utilisée pour obtenir le nombre total de rangs dans un résultat trié ?

- C) COUNT_ROWS()

21. Quelle clause SQL est utilisée pour spécifier une condition de jointure basée sur une table temporaire dans une requête SELECT ?

- A) JOIN TEMPORARY TABLE

22. Quelle instruction SQL est utilisée pour ajouter une contrainte de clé étrangère référençant plusieurs colonnes dans une table ?

- C) ALTER TABLE ADD COMPOSITE FOREIGN KEY

23. Quelle fonction SQL est utilisée pour obtenir le percentile d'une valeur dans un ensemble de données ?

- A) PERCENTILE()

24. Quelle clause SQL est utilisée pour spécifier une condition de jointure basée sur une fonction dans une requête SELECT ?

- C) JOIN ON FUNCTION

25. Quelle instruction SQL est utilisée pour ajouter une contrainte UNIQUE à plusieurs colonnes dans une table ?

- C) ALTER TABLE ADD UNIQUE COMPOSITE

26. Quelle fonction SQL est utilisée pour obtenir le nombre total de rangs affectés par une requête ?

- B) AFFECTED_ROWS()

27. Quelle clause SQL est utilisée pour spécifier une condition de jointure basée sur une expression de fonction dans une requête SELECT ?

- C) JOIN ON FUNCTION

28. Quelle instruction SQL est utilisée pour ajouter une contrainte de clé étrangère conditionnelle dans une table ?

- C) ALTER TABLE ADD CONDITIONAL FOREIGN KEY

29. Quelle fonction SQL est utilisée pour obtenir le nombre total de rangs dans un ensemble de données filtré par une condition spécifique ?

- C) COUNT()

30. Quelle clause SQL est utilisée pour spécifier une condition de jointure basée sur une colonne

calculée dans une requête SELECT ?

- C) JOIN ON CALCULATED COLUMN

Exercices SQL

dimanche 2 juin 2024 15:17

Exercice 1 : CTE et Fenêtres

Objectif

Utiliser une expression de table commune (CTE) et des fonctions de fenêtre pour trouver le deuxième plus haut salaire de chaque département.

Schéma de la table

```
```sql
```

```
CREATE TABLE Employees (
 EmployeeID INT PRIMARY KEY,
 Name VARCHAR(100),
 DepartmentID INT,
 Salary DECIMAL(10, 2)
);
```

```
CREATE TABLE Departments (
 DepartmentID INT PRIMARY KEY,
 DepartmentName VARCHAR(100)
);
```
```

Exercice

Écrire une requête pour afficher le deuxième plus haut salaire de chaque département.

Solution

```
```sql
```

```
WITH SalaryRank AS (
 SELECT
 DepartmentID,
 Salary,
 ROW_NUMBER() OVER (PARTITION BY DepartmentID ORDER BY Salary DESC) AS Rank
 FROM Employees
)
SELECT
 DepartmentID,
 Salary AS SecondHighestSalary
FROM SalaryRank
WHERE Rank = 2;
```
```

Exercice 2 : Agrégation et Sous-requêtes

Objectif

Calculer le nombre total de produits vendus par catégorie et le revenu total généré par chaque catégorie.

Schéma de la table

```
```sql
```

```
CREATE TABLE Products (
 ProductID INT PRIMARY KEY,
 ProductName VARCHAR(100),
 CategoryID INT,
 Price DECIMAL(10, 2)
```



```
);
```

```
CREATE TABLE Sales (
 SaleID INT PRIMARY KEY,
 ProductID INT,
 Quantity INT,
 SaleDate DATE
```

```
);
```

```
CREATE TABLE Categories (
 CategoryID INT PRIMARY KEY,
 CategoryName VARCHAR(100)
```

```
);
...
```

#### #### Exercice

Écrire une requête pour afficher le nombre total de produits vendus par catégorie et le revenu total généré par chaque catégorie.

#### #### Solution

```
```sql  
SELECT  
    c.CategoryName,  
    COUNT(s.SaleID) AS TotalSales,  
    SUM(p.Price * s.Quantity) AS TotalRevenue  
FROM  
    Categories c  
    JOIN Products p ON c.CategoryID = p.CategoryID  
    JOIN Sales s ON p.ProductID = s.ProductID  
GROUP BY  
    c.CategoryName;  
...
```

Exercice 3 : Jointure et Filtrage

Objectif

Trouver les employés qui n'ont pas encore été affectés à un projet et lister leurs noms et départements.

Schéma de la table

```
```sql  
CREATE TABLE Employees (
 EmployeeID INT PRIMARY KEY,
 Name VARCHAR(100),
 DepartmentID INT
);
```

```
CREATE TABLE Projects (
 ProjectID INT PRIMARY KEY,
 ProjectName VARCHAR(100)
);
```

```
CREATE TABLE EmployeeProjects (
 EmployeeID INT,
 ProjectID INT,
 PRIMARY KEY (EmployeeID, ProjectID)
);
```

```
CREATE TABLE Departments (
 DepartmentID INT PRIMARY KEY,
 DepartmentName VARCHAR(100)
);

```

#### #### Exercice

Écrire une requête pour trouver les employés qui n'ont pas encore été affectés à un projet et afficher leurs noms et départements.

#### #### Solution

```
--sql
SELECT
 e.Name,
 d.DepartmentName
FROM
 Employees e
 JOIN Departments d ON e.DepartmentID = d.DepartmentID
LEFT JOIN EmployeeProjects ep ON e.EmployeeID = ep.EmployeeID
WHERE
 ep.ProjectID IS NULL;

```

#### ### Exercice 4 : Fenêtres et Agrégation

##### #### Objectif

Calculer le salaire cumulé de chaque employé par département.

##### #### Schéma de la table

```
--sql
CREATE TABLE Employees (
 EmployeeID INT PRIMARY KEY,
 Name VARCHAR(100),
 DepartmentID INT,
 Salary DECIMAL(10, 2)
);
```

```
CREATE TABLE Departments (
 DepartmentID INT PRIMARY KEY,
 DepartmentName VARCHAR(100)
);

```

#### #### Exercice

Écrire une requête pour afficher le salaire cumulé de chaque employé par département.

#### #### Solution

```
--sql
SELECT
 e.Name,
 d.DepartmentName,
 e.Salary,
 SUM(e.Salary) OVER (PARTITION BY e.DepartmentID ORDER BY e.Salary DESC) AS
CumulativeSalary
FROM
 Employees e
 JOIN Departments d ON e.DepartmentID = d.DepartmentID
ORDER BY
```

```
d.DepartmentName, e.Salary DESC;
...
```

### ### Exercice 5 : Analyse et Pivot

#### #### Objectif

Créer un rapport de ventes mensuelles par produit.

#### #### Schéma de la table

```
```sql  
CREATE TABLE Products (  
    ProductID INT PRIMARY KEY,  
    ProductName VARCHAR(100)  
);
```

```
CREATE TABLE Sales (  
    SaleID INT PRIMARY KEY,  
    ProductID INT,  
    Quantity INT,  
    SaleDate DATE  
);  
...
```

Exercice

Écrire une requête pour afficher un rapport de ventes mensuelles par produit.

Solution

```
```sql  
SELECT
 p.ProductName,
 SUM(CASE WHEN MONTH(s.SaleDate) = 1 THEN s.Quantity ELSE 0 END) AS Jan,
 SUM(CASE WHEN MONTH(s.SaleDate) = 2 THEN s.Quantity ELSE 0 END) AS Feb,
 SUM(CASE WHEN MONTH(s.SaleDate) = 3 THEN s.Quantity ELSE 0 END) AS Mar,
 SUM(CASE WHEN MONTH(s.SaleDate) = 4 THEN s.Quantity ELSE 0 END) AS Apr,
 SUM(CASE WHEN MONTH(s.SaleDate) = 5 THEN s.Quantity ELSE 0 END) AS May,
 SUM(CASE WHEN MONTH(s.SaleDate) = 6 THEN s.Quantity ELSE 0 END) AS Jun,
 SUM(CASE WHEN MONTH(s.SaleDate) = 7 THEN s.Quantity ELSE 0 END) AS Jul,
 SUM(CASE WHEN MONTH(s.SaleDate) = 8 THEN s.Quantity ELSE 0 END) AS Aug,
 SUM(CASE WHEN MONTH(s.SaleDate) = 9 THEN s.Quantity ELSE 0 END) AS Sep,
 SUM(CASE WHEN MONTH(s.SaleDate) = 10 THEN s.Quantity ELSE 0 END) AS Oct,
 SUM(CASE WHEN MONTH(s.SaleDate) = 11 THEN s.Quantity ELSE 0 END) AS Nov,
 SUM(CASE WHEN MONTH(s.SaleDate) = 12 THEN s.Quantity ELSE 0 END) AS Dec
FROM
 Products p
 JOIN Sales s ON p.ProductID = s.ProductID
GROUP BY
 p.ProductName
ORDER BY
 p.ProductName;
...
```

Ces exercices devraient fournir une bonne pratique pour des concepts SQL avancés tels que les CTE, les fonctions de fenêtre, les agrégations complexes, et les jointures élaborées.

# PLSQL

dimanche 2 juin 2024

14:58

# PL/SQL avancées

dimanche 2 juin 2024 15:20

1. Quelle commande PL/SQL est utilisée pour créer un bloc anonyme ?
  - **A) BEGIN**
  - B) DECLARE
  - C) CREATE
2. Quelle clause PL/SQL est utilisée pour gérer les exceptions dans un bloc PL/SQL ?
  - **A) EXCEPTION**
  - B) HANDLE
  - C) CATCH
3. Quelle méthode est utilisée pour lever une exception définie par l'utilisateur dans PL/SQL ?
  - **A) RAISE\_APPLICATION\_ERROR**
  - B) RAISE\_USER\_ERROR
  - C) RAISE\_ERROR
4. Quelle est la portée d'une variable déclarée dans un bloc anonyme PL/SQL ?
  - A) Globale
  - **B) Locale au bloc**
  - C) Locale à la procédure
5. Quelle instruction PL/SQL est utilisée pour démarrer une transaction ?
  - A) BEGIN TRANSACTION
  - B) START TRANSACTION
  - **C) Il n'y a pas d'instruction explicite**
6. Quelle clause PL/SQL est utilisée pour renvoyer des valeurs à partir d'une fonction ?
  - **A) RETURN**
  - B) OUTPUT
  - C) RESULT
7. Quelle commande est utilisée pour créer un déclencheur (trigger) dans PL/SQL ?
  - **A) CREATE TRIGGER**
  - B) CREATE EVENT
  - C) CREATE PROCEDURE TRIGGER
8. Quelle clause PL/SQL est utilisée pour vérifier si un curseur est ouvert ?
  - **A) IF CURSOR%ISOPEN**
  - B) IF CURSOR\_OPEN
  - C) IF CURSOR\_ISOPEN
9. Quelle méthode est utilisée pour fermer explicitement un curseur dans PL/SQL ?
  - A) CLOSE CURSOR
  - B) CURSOR CLOSE
  - **C) CLOSE**
10. Quelle clause PL/SQL est utilisée pour créer un paquetage (package) ?
  - **A) CREATE PACKAGE**
  - B) CREATE BUNDLE
  - C) CREATE SET

11. Quelle est la portée d'une variable déclarée dans un paquetage (package) PL/SQL ?
- **A) Locale au package**
  - B) Globale
  - C) Locale à la procédure
12. Quelle clause PL/SQL est utilisée pour itérer sur un ensemble de résultats ?
- **A) FOR LOOP**
  - B) WHILE LOOP
  - C) REPEAT LOOP
13. Quelle clause est utilisée pour déclarer un curseur explicite dans PL/SQL ?
- A) DECLARE CURSOR
  - B) OPEN CURSOR
  - **C) CURSOR**
14. Quelle clause PL/SQL est utilisée pour lancer une boucle simple ?
- **A) LOOP ... END LOOP**
  - B) BEGIN ... END LOOP
  - C) DO ... END LOOP
15. Quelle commande est utilisée pour assigner une valeur à une variable dans PL/SQL ?
- **A) :=**
  - B) =
  - C) <-
16. Quelle clause est utilisée pour capturer une exception pré-définie dans PL/SQL ?
- A) WHEN OTHERS THEN
  - B) EXCEPTION
  - **C) WHEN**
17. Quelle clause est utilisée pour définir un curseur FOR UPDATE ?
- **A) CURSOR ... FOR UPDATE**
  - B) FOR UPDATE CURSOR
  - C) DECLARE CURSOR ... FOR UPDATE
18. Quelle méthode est utilisée pour ouvrir un curseur dans PL/SQL ?
- A) OPEN CURSOR
  - B) CURSOR OPEN
  - **C) OPEN**
19. Quelle méthode est utilisée pour récupérer la prochaine ligne d'un curseur dans PL/SQL ?
- **A) FETCH CURSOR INTO**
  - B) FETCH INTO CURSOR
  - C) CURSOR FETCH INTO
20. Quelle clause PL/SQL est utilisée pour boucler jusqu'à une condition spécifique ?
- **A) WHILE ... END LOOP**
  - B) LOOP UNTIL ... END LOOP
  - C) REPEAT ... UNTIL
21. Quelle méthode est utilisée pour appeler une procédure stockée dans PL/SQL ?
- A) EXECUTE PROCEDURE
  - B) CALL PROCEDURE
  - **C) EXECUTE**
22. Quelle clause est utilisée pour déclarer une exception définie par l'utilisateur dans PL/SQL ?
- A) EXCEPTION

- **B) DECLARE EXCEPTION**
- C) RAISE EXCEPTION

23. Quelle clause est utilisée pour déclarer une constante dans PL/SQL ?

- **A) CONSTANT**
- B) CONSTANT DECLARE
- C) DECLARE CONSTANT

24. Quelle clause est utilisée pour spécifier les paramètres d'une procédure stockée dans PL/SQL ?

- A) IN/OUT/IN OUT
- B) INPUT/OUTPUT/INOUT
- C) PARAMETER

25. Quelle méthode est utilisée pour engager une transaction dans PL/SQL ?

- A) COMMIT TRANSACTION
- B) TRANSACTION COMMIT
- **C) COMMIT**

26. Quelle clause PL/SQL est utilisée pour lancer une exception définie par l'utilisateur ?

- **A) RAISE**
- B) THROW
- C) SIGNAL

27. Quelle clause PL/SQL est utilisée pour créer une fonction stockée ?

- **A) CREATE FUNCTION**
- B) CREATE PROCEDURE
- C) CREATE ROUTINE

28. Quelle méthode est utilisée pour annuler une transaction dans PL/SQL ?

- A) ROLLBACK TRANSACTION
- B) TRANSACTION ROLLBACK
- **C) ROLLBACK**

29. Quelle clause PL/SQL est utilisée pour vérifier si une variable est NULL ?

- **A) IS NULL**
- B) NULL CHECK
- C) IF NULL

30. Quelle méthode est utilisée pour déclarer une variable dans PL/SQL ?

- **A) DECLARE**
- B) SET
- C) VAR

# Exercices PL/SQL

lundi 3 juin 2024 09:19

## ### Exercice 1 : Création d'un Bloc Anonyme

### #### Objectif

Créer un bloc PL/SQL anonyme qui affiche "Hello, World!" dans la console.

### #### Étapes

1. Écrivez un bloc PL/SQL anonyme.
2. Utilisez la procédure `DBMS\_OUTPUT.PUT\_LINE` pour afficher le texte.

### #### Solution

```
``sql
BEGIN
 DBMS_OUTPUT.PUT_LINE('Hello, World!');
END;
/
``
```

## ### Exercice 2 : Variables et Calculs

### #### Objectif

Créer un bloc PL/SQL anonyme qui calcule la somme de deux nombres et affiche le résultat.

### #### Étapes

1. Déclarez deux variables numériques et une variable pour stocker la somme.
2. Assignez des valeurs aux deux premières variables.
3. Calculez la somme et assignez-la à la troisième variable.
4. Affichez le résultat avec `DBMS\_OUTPUT.PUT\_LINE`.

### #### Solution

```
``sql
DECLARE
 num1 NUMBER := 10;
 num2 NUMBER := 20;
 sum NUMBER;
BEGIN
 sum := num1 + num2;
 DBMS_OUTPUT.PUT_LINE('La somme de ' || num1 || ' et ' || num2 || ' est ' || sum);
END;
/
``
```

## ### Exercice 3 : Boucles et Conditions

### #### Objectif

Créer un bloc PL/SQL anonyme qui affiche les nombres de 1 à 10, mais indique pour chaque nombre s'il est pair ou impair.

### #### Étapes

1. Utilisez une boucle `FOR` pour itérer de 1 à 10.
2. Utilisez une condition `IF` pour vérifier si un nombre est pair ou impair.
3. Affichez le nombre et son statut (pair ou impair).

### #### Solution

```
``sql
```



```

BEGIN
 FOR i IN 1..10 LOOP
 IF MOD(i, 2) = 0 THEN
 DBMS_OUTPUT.PUT_LINE(i || ' est pair');
 ELSE
 DBMS_OUTPUT.PUT_LINE(i || ' est impair');
 END IF;
 END LOOP;
END;
/
...

```

### ### Exercice 4 : Utilisation de Cursors

#### #### Objectif

Créer un curseur pour récupérer et afficher tous les noms d'employés d'une table `EMPLOYEES`.

#### #### Étapes

1. Déclarez un curseur pour sélectionner les noms d'employés.
2. Ouvrez le curseur.
3. Utilisez une boucle `FETCH` pour récupérer les noms d'employés un par un et les afficher.
4. Fermez le curseur.

#### #### Solution

```

```sql
DECLARE
  CURSOR emp_cursor IS
    SELECT employee_name FROM employees;
  emp_name employees.employee_name%TYPE;
BEGIN
  OPEN emp_cursor;
  LOOP
    FETCH emp_cursor INTO emp_name;
    EXIT WHEN emp_cursor%NOTFOUND;
    DBMS_OUTPUT.PUT_LINE('Nom de l'employé : ' || emp_name);
  END LOOP;
  CLOSE emp_cursor;
END;
/
...

```

Exercice 5 : Procédures Stockées

Objectif

Créer une procédure stockée qui prend un ID d'employé et affiche le nom de l'employé correspondant.

Étapes

1. Créez une procédure nommée `get_employee_name`.
2. La procédure doit accepter un paramètre `employee_id`.
3. Utilisez une instruction `SELECT INTO` pour récupérer le nom de l'employé correspondant à l'ID fourni.
4. Affichez le nom de l'employé avec `DBMS_OUTPUT.PUT_LINE`.

Solution

```

```sql
CREATE OR REPLACE PROCEDURE get_employee_name (employee_id IN employees.employee_id%
TYPE) IS
 emp_name employees.employee_name%TYPE;

```

```

BEGIN
 SELECT employee_name INTO emp_name
 FROM employees
 WHERE employee_id = employee_id;
 DBMS_OUTPUT.PUT_LINE('Nom de l'employé : ' || emp_name);
EXCEPTION
 WHEN NO_DATA_FOUND THEN
 DBMS_OUTPUT.PUT_LINE('Aucun employé trouvé avec l''ID ' || employee_id);
 WHEN OTHERS THEN
 DBMS_OUTPUT.PUT_LINE('Erreur : ' || SQLERRM);
END get_employee_name;
/
'''

```

### ### Exercice 6 : Fonctions Stockées

#### #### Objectif

Créer une fonction stockée qui prend un ID de département et retourne le nombre d'employés dans ce département.

#### #### Étapes

1. Créez une fonction nommée `get\_employee\_count`.
2. La fonction doit accepter un paramètre `department\_id` et retourner un nombre.
3. Utilisez une instruction `SELECT COUNT(\*)` pour obtenir le nombre d'employés dans le département spécifié.

#### #### Solution

```

'''sql
CREATE OR REPLACE FUNCTION get_employee_count (department_id IN
employees.department_id%TYPE) RETURN NUMBER IS
 emp_count NUMBER;
BEGIN
 SELECT COUNT(*)
 INTO emp_count
 FROM employees
 WHERE department_id = department_id;
 RETURN emp_count;
END get_employee_count;
/
'''

```

### ### Exercice 7 : Gestion des Exceptions

#### #### Objectif

Créer un bloc PL/SQL qui tente de diviser un nombre par zéro et gère l'exception en affichant un message d'erreur approprié.

#### #### Étapes

1. Déclarez deux variables numériques et assignez-leur des valeurs.
2. Tentez de diviser l'une par l'autre.
3. Gérez l'exception `ZERO\_DIVIDE` en affichant un message d'erreur.

#### #### Solution

```

'''sql
DECLARE
 num1 NUMBER := 10;
 num2 NUMBER := 0;
 result NUMBER;
BEGIN

```

```
result := num1 / num2;
DBMS_OUTPUT.PUT_LINE('Le résultat est ' || result);
EXCEPTION
 WHEN ZERO_DIVIDE THEN
 DBMS_OUTPUT.PUT_LINE('Erreur : Division par zéro');
END;
/
...
```

Ces exercices couvrent des concepts clés de PL/SQL, y compris les blocs anonymes, les variables, les boucles, les conditions, les curseurs, les procédures et fonctions stockées, ainsi que la gestion des exceptions. Ils devraient vous aider à renforcer vos compétences en PL/SQL.

# Git

dimanche 2 juin 2024

15:21