



GESTION DES COMPTES UTILISATEURS ORACLE
SOUS DELPHI

Document créé par : Charef Zitouni

*Plus
d'Actions*

1. Créer un compte
2. Activer un compte
3. Désactiver un compte
4. Supprimer un compte
5. Changer mot de passe
6. Afficher les comptes
7. Créer un rôle
8. Supprimer un rôle

En savoir plus

TUTORIEL GESTION DES COMPTES ORACLE AVEC DELPHI

Ce court tutoriel est destiné à ceux qui commencent à créer d'une forme Delphi qui permet d'administrer et gérer les comptes oracle.

La gestion des comptes Oracle avec le langage Delphi offre diverses fonctionnalités pour administrer les comptes d'utilisateurs dans une base de données Oracle. Voici une introduction aux principales fonctionnalités de gestion des comptes :

1. Création des comptes : Avec Delphi, vous pouvez facilement créer de nouveaux comptes utilisateur Oracle en utilisant des requêtes SQL pour insérer les informations d'utilisateur dans la table appropriée. Vous pouvez spécifier des détails tels que le nom d'utilisateur, le mot de passe et les autorisations associées au compte.
2. Activation des comptes : Lors de la création d'un compte utilisateur, il est souvent nécessaire de l'activer pour permettre à l'utilisateur de se connecter. Vous pouvez utiliser Delphi pour exécuter une requête SQL qui mettra à jour l'état du compte pour le rendre actif. Cela permettra à l'utilisateur de se connecter à la base de données.
3. Verrouillage des comptes : Dans certains cas, vous voudrez peut-être verrouiller un compte utilisateur pour des raisons de sécurité ou de gestion. Cela empêchera l'utilisateur de se connecter à la base de données. Avec Delphi, vous pouvez envoyer une requête SQL pour modifier l'état du compte et le verrouiller. Vous pouvez également spécifier des conditions, telles que le nombre de tentatives de connexion infructueuses, pour déclencher automatiquement le verrouillage du compte.
4. Déverrouillage des comptes : Si un compte utilisateur est verrouillé, vous pouvez utiliser Delphi pour envoyer une requête SQL qui le déverrouillera. Cela permettra à l'utilisateur de se connecter à nouveau à la base de données.
5. Modification des informations du compte : Delphi offre des fonctionnalités pour modifier les informations d'un compte utilisateur existant. Vous pouvez utiliser des requêtes SQL pour mettre à jour des éléments tels que le nom d'utilisateur, le mot de passe, les autorisations ou d'autres détails associés au compte.
6. Suppression des comptes : Si vous souhaitez supprimer un compte utilisateur, Delphi vous permet d'exécuter des requêtes SQL pour supprimer les informations du compte de la base de données. Cela peut être utile lorsque vous devez désactiver ou supprimer des comptes utilisateur obsolètes ou inactifs.

En utilisant les fonctionnalités de gestion des comptes Oracle avec Delphi, vous pouvez contrôler et administrer efficacement les comptes d'utilisateurs dans votre application. Ces fonctionnalités vous offrent la flexibilité nécessaire pour créer, activer, verrouiller, déverrouiller, modifier et supprimer des comptes utilisateur en fonction des besoins spécifiques de votre application et des politiques de sécurité de votre base de données Oracle.

Figure forme 'Gestion des comptes'.

Plus d'Actions

1. Créer un compte
2. Activer un compte
3. Désactiver un compte
4. Supprimer un compte
5. Changer mot de passe
6. Afficher les comptes
7. Créer un rôle
8. Supprimer un rôle

En savoir plus

I: LA CRÉATION D'UN COMPTE

Pour créer un compte Oracle en utilisant AdoCommand de Delphi, vous pouvez suivre les étapes suivantes :

1. Ajouter les composants nécessaires : Assurez-vous d'avoir les composants ADO nécessaires dans la palette d'outils de Delphi. Si vous ne les avez pas, vous pouvez les ajouter en sélectionnant "Component" -> "Install Packages" dans l'IDE de Delphi et en activant les packages correspondants à ADO.
2. Créer une connexion à la base de données Oracle : Ajoutez un composant de connexion ADO à votre formulaire Delphi. Cela peut être le composant TADOConnection. Configurez les paramètres de connexion, tels que le nom d'hôte, le port, le nom d'utilisateur et le mot de passe pour la base de données Oracle à laquelle vous souhaitez vous connecter.
3. Créer un composant ADOCommand : Ajoutez un composant ADOCommand à votre formulaire Delphi. Cela peut être le composant TADOCommand. Associez ce composant à la connexion ADO créée précédemment en définissant sa propriété Connection sur le composant TADOConnection.
4. Écrire la requête SQL de création du compte : Utilisez la propriété CommandText du composant ADOCommand pour définir la requête SQL qui créera le compte Oracle. La requête SQL doit être au format valide Oracle.

Cette approche vous permet de créer un compte Oracle en utilisant AdoCommand de Delphi. Veillez à prendre en compte les autorisations nécessaires pour exécuter la commande CREATE USER et assurez-vous de disposer des droits appropriés pour effectuer cette opération dans la base de données Oracle cible.

Copy code

```
procedure TForm14.SpeedButton1Click(Sender: TObject);
begin
  ADOCommand1.CommandText:='';
  if (edit1.Text='') OR (edit2.Text='') then
    showmessage('Champ vide non autorisé')
  else
    if edit2.Text<>edit3.Text then
      Showmessage('Les mots de passes ne sont pas identiques !... Merci de vérifier')
    else
      Begin
        ADOCommand1.commandtext:='CREATE USER '+edit1.text+' IDENTIFIED BY '+edit2.text;
```

```

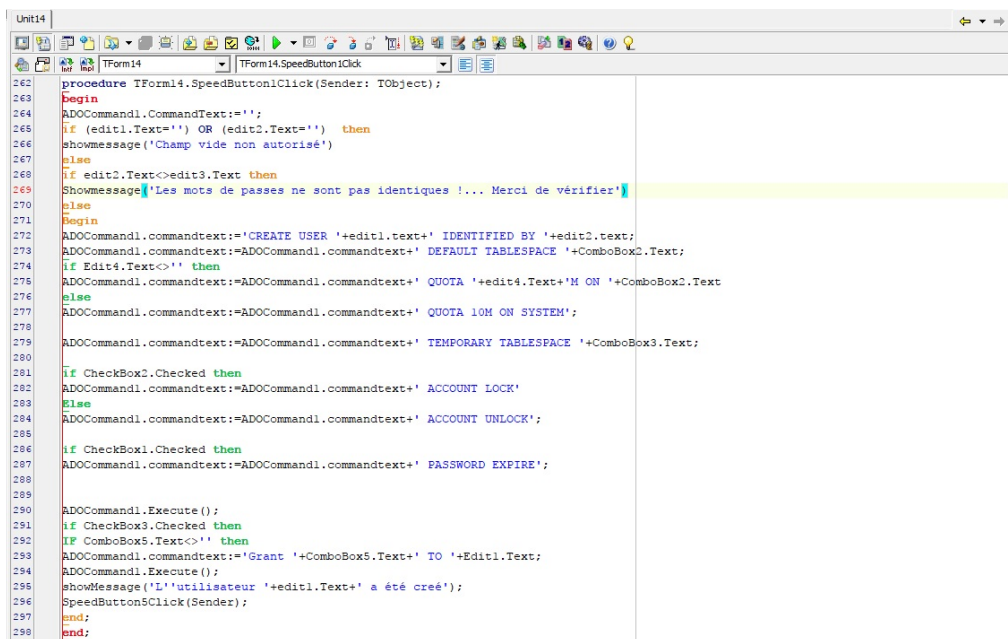
ADOCommand1.commandtext:=ADOCommand1.commandtext+' DEFAULT TABLESPACE '+ComboBox2.Text;
if Edit4.Text<>'' then
ADOCommand1.commandtext:=ADOCommand1.commandtext+' QUOTA '+edit4.Text+'M ON '+ComboBox2.Text
else
ADOCommand1.commandtext:=ADOCommand1.commandtext+' QUOTA 10M ON SYSTEM';
ADOCommand1.commandtext:=ADOCommand1.commandtext+' TEMPORARY TABLESPACE '+ComboBox3.Text;
if CheckBox2.Checked then
ADOCommand1.commandtext:=ADOCommand1.commandtext+' ACCOUNT LOCK'
Else
ADOCommand1.commandtext:=ADOCommand1.commandtext+' ACCOUNT UNLOCK';
if CheckBox1.Checked then
ADOCommand1.commandtext:=ADOCommand1.commandtext+' PASSWORD EXPIRE';
ADOCommand1.Execute();
if CheckBox3.Checked then
IF ComboBox5.Text<>'' then
ADOCommand1.commandtext:='Grant '+ComboBox5.Text+' TO '+Edit1.Text;
ADOCommand1.Execute();
showMessage('L'utilisateur '+edit1.Text+' a été créé');
SpeedButton5Click(Sender);
end;
end;

```

Plus d'Actions

1. Créer un compte
 2. Activer un compte
 3. Désactiver un compte
 4. Supprimer un compte
 5. Changer mot de passe
 6. Afficher les comptes
 7. Créer un rôle
 8. Supprimer un rôle
- En savoir plus

En fait, vous n'avez pas à le taper: vous pouvez le copier et coller depuis cette page Web dans votre éditeur Delphi.



Capture affichant le code source de la création d'un compte Oracle.

2: ACTIVER UN COMPTE

Pour activer un compte Oracle en utilisant AdoCommand de Delphi, vous pouvez suivre ces étapes :

1. Assurez-vous d'avoir déjà créé une connexion à la base de données Oracle en utilisant un composant ADOConnection. Si vous n'avez pas encore créé la connexion, veuillez vous référer à la section précédente sur la création d'un compte Oracle avec AdoCommand de Delphi.
2. Ajoutez un composant ADOCommand à votre formulaire Delphi (par exemple, TADOCommand). Associez ce composant à la connexion ADO créée précédemment en définissant sa propriété Connection sur le composant TADOConnection.
3. Écrivez la requête SQL pour activer le compte Oracle. La requête peut varier en fonction de la version d'Oracle que vous utilisez. Voici un exemple de requête pour activer un compte Oracle :

Copy code

```

procedure TForm14.SpeedButton4Click(Sender: TObject);
begin
if edit1.text<>'' then
begin
if ADOQListeUsers.Locate('username',edit1.Text,[]) then

```

```

Begin
if MessageDlg('Voulez-vous vraiment activer l'utilisateur : '+edit1.Text+'?',mtConfirmation, [mbYes, mbNo]
begin
ADOCommand1.CommandText := 'ALTER USER '+UpperCase(edit1.text)+' ACCOUNT UNLOCK';
ADOCommand1.execute();
Showmessage('L'utilisateur '+Edit1.text+ ' est activé');
SpeedButton5Click(Sender);
end;
end
else
showmessage('utilisateur introuvable');
end
else
showmessage('Champ vide , veuillez saisir le nom d'utilisateur')
end;

```

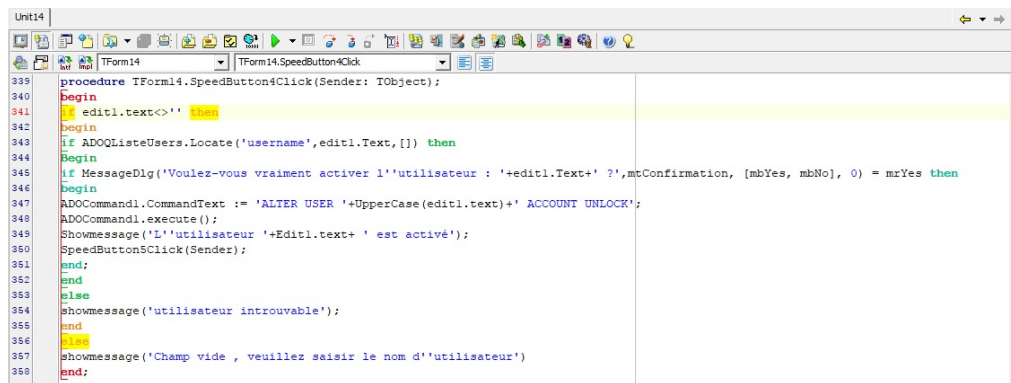
Plus d'Actions

1. Créer un compte
2. Activer un compte
3. Désactiver un compte
4. Supprimer un compte
5. Changer mot de passe
6. Afficher les comptes
7. Créer un rôle
8. Supprimer un rôle

En savoir plus

- Il est important de noter que l'activation d'un compte Oracle peut nécessiter des privilèges spécifiques. Assurez-vous d'avoir les droits appropriés pour exécuter la commande ALTER USER et activer le compte Oracle dans la base de données.

- En suivant ces étapes, vous pourrez activer un compte Oracle en utilisant AdoCommand de Delphi. Assurez-vous d'adapter ces instructions en fonction de votre environnement Oracle spécifique et des besoins de votre application. Veuillez à adapter ces instructions en fonction de votre environnement Oracle spécifique et des besoins de votre application.



Capture affichant le code source de l'activation d'un compte Oracle sous Delphi.

3: DÉSACTIVER UN COMPTE

Pour désactiver un compte Oracle à l'aide d'AdoCommand de Delphi, vous pouvez suivre ces étapes :

1. Assurez-vous d'avoir déjà créé une connexion à la base de données Oracle en utilisant un composant ADOConnection. Si vous n'avez pas encore créé la connexion, veuillez vous référer à la section précédente sur la création d'un compte Oracle avec AdoCommand de Delphi.
2. Ajoutez un composant ADOCommand à votre formulaire Delphi (par exemple, TADOCommand). Associez ce composant à la connexion ADO créée précédemment en définissant sa propriété Connection sur le composant TADOConnection.
3. Écrivez la requête SQL pour désactiver le compte Oracle. La requête peut varier en fonction de la version d'Oracle que vous utilisez. Voici un exemple de requête pour désactiver un compte Oracle :

☐ Copy code

```

procedure TForm14.SpeedButton3Click(Sender: TObject);
begin
if edit1.text<>'' then
begin
if ADOQListeUsers.Locate('username',edit1.Text,[]) then
Begin
if MessageDlg('Voulez-vous vraiment désactiver l'utilisateur : '+edit1.Text+'?',mtConfirmation, [mbYes, mt
begin
ADOCommand1.CommandText := 'ALTER USER '+UpperCase(edit1.text)+' ACCOUNT LOCK';
ADOCommand1.execute();
Showmessage('L'utilisateur '+Edit1.text+ ' est désactivé');

```

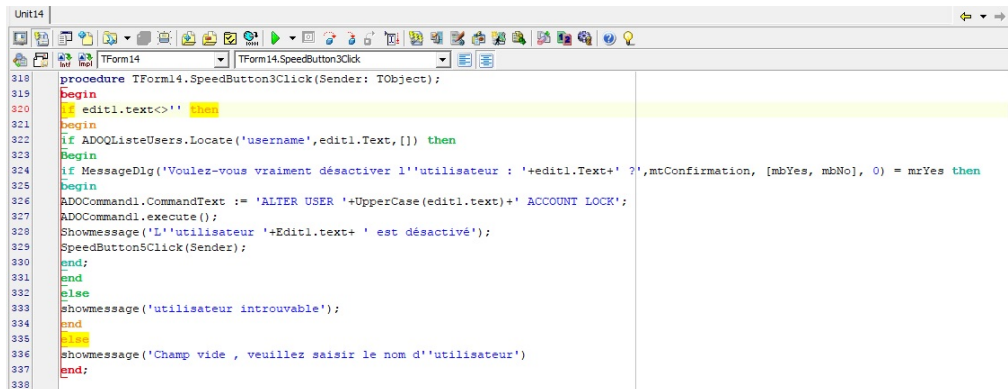
```

SpeedButton5Click(Sender);
end;
end
else
showmessage('utilisateur introuvable');
end
else
showmessage('Champ vide , veuillez saisir le nom d''utilisateur')
end;

```

- Il est important de noter que la désactivation d'un compte Oracle peut nécessiter des privilèges spécifiques. Assurez-vous d'avoir les droits appropriés pour exécuter la commande ALTER USER et désactiver le compte Oracle dans la base de données.

- En suivant ces étapes, vous pourrez désactiver un compte Oracle à l'aide d'AdoCommand de Delphi. Veillez à adapter ces instructions en fonction de votre environnement Oracle spécifique et des besoins de votre application.



Capture affichant le code source de la désactivation d'un compte Oracle sous Delphi.

Plus d'Actions

1. Créer un compte
2. Activer un compte
3. Désactiver un compte
4. Supprimer un compte
5. Changer mot de passe
6. Afficher les comptes
7. Créer un rôle
8. Supprimer un rôle

En savoir plus

4: SUPPRIMER UN COMPTE

Pour supprimer un compte Oracle à l'aide d'AdoCommand de Delphi, vous pouvez suivre ces étapes :

1. Assurez-vous d'avoir déjà créé une connexion à la base de données Oracle en utilisant un composant ADOConnection. Si vous n'avez pas encore créé la connexion, veuillez vous référer à la section précédente sur la création d'un compte Oracle avec AdoCommand de Delphi.
2. Ajoutez un composant ADOCommand à votre formulaire Delphi (par exemple, TADOCommand). Associez ce composant à la connexion ADO créée précédemment en définissant sa propriété Connection sur le composant TADOConnection.
3. Écrivez la requête SQL pour supprimer le compte Oracle. La requête peut varier en fonction de la version d'Oracle que vous utilisez. Voici un exemple de requête pour supprimer un compte Oracle :

Copy code

```

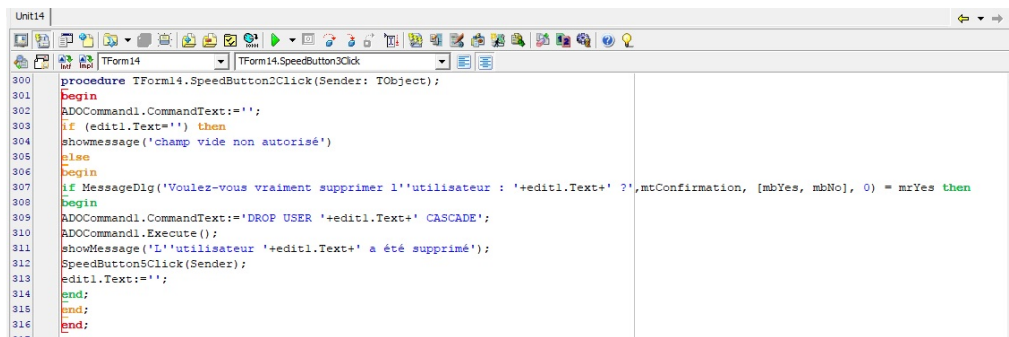
procedure TForm14.SpeedButton2Click(Sender: TObject);
begin
ADOCommand1.CommandText:='';
if (edit1.Text='') then
showmessage('champ vide non autorisé')
else
begin
if MessageDlg('Voulez-vous vraiment supprimer l''utilisateur : '+edit1.Text+' ?',mtConfirmation, [mbYes, mbNo], 0) = mrYes then
begin
ADOCommand1.CommandText:='DROP USER '+edit1.Text+' CASCADE';
ADOCommand1.Execute();
showMessage('L''utilisateur '+edit1.Text+' a été supprimé');
SpeedButton5Click(Sender);
edit1.Text:='';
end;
end;
end;
end;

```

- Il est important de noter que la suppression d'un compte Oracle peut nécessiter des privilèges spécifiques. Assurez-vous d'avoir les droits appropriés pour exécuter la commande DROP USER et supprimer le compte Oracle dans la base de données.

- En suivant ces étapes, vous pourrez supprimer un compte Oracle à l'aide d'AdoCommand de Delphi. Veillez à adapter ces instructions en

fonction de votre environnement Oracle spécifique et des besoins de votre application.



Capture affichant le code source de la suppression d'un compte Oracle sous Delphi.

Plus d'Actions

1. Créer un compte
2. Activer un compte
3. Désactiver un compte
4. Supprimer un compte
5. Changer mot de passe
6. Afficher les comptes
7. Créer un rôle
8. Supprimer un rôle

[En savoir plus](#)

5: CHANGER LE MOT DE PASSE

Pour changer le mot de passe d'un compte Oracle à l'aide d'AdoCommand de Delphi, vous pouvez suivre ces étapes :

1. Assurez-vous d'avoir déjà créé une connexion à la base de données Oracle en utilisant un composant ADOConnection. Si vous n'avez pas encore créé la connexion, veuillez vous référer à la section précédente sur la création d'un compte Oracle avec AdoCommand de Delphi.
2. Ajoutez un composant ADOCommand à votre formulaire Delphi (par exemple, TADOCommand). Associez ce composant à la connexion ADO créée précédemment en définissant sa propriété Connection sur le composant TADOConnection.
3. Écrivez la requête SQL pour changer le mot de passe du compte Oracle. La requête peut varier en fonction de la version d'Oracle que vous utilisez. Voici un exemple de requête pour changer le mot de passe d'un compte Oracle :

☐ Copy code

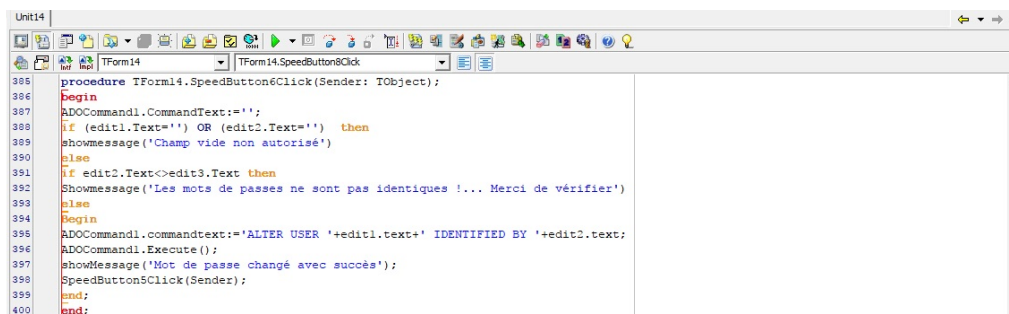
```

procedure TForm14.SpeedButton6Click(Sender: TObject);
begin
  ADOCommand1.CommandText:='';
  if (edit1.Text='') OR (edit2.Text='') then
    showmessage('Champ vide non autorisé')
  else
    if edit2.Text<>edit3.Text then
      Showmessage('Les mots de passes ne sont pas identiques !... Merci de vérifier')
    else
      Begin
        ADOCommand1.commandtext:='ALTER USER '+edit1.text+' IDENTIFIED BY '+edit2.text;
        ADOCommand1.Execute();
        showMessage('Mot de passe changé avec succès');
        SpeedButton5Click(Sender);
      end;
    end;
end;

```

- Il est important de noter que le changement de mot de passe d'un compte Oracle peut nécessiter des privilèges spécifiques. Assurez-vous d'avoir les droits appropriés pour exécuter la commande ALTER USER et changer le mot de passe du compte Oracle dans la base de données.

- En suivant ces étapes, vous pourrez changer le mot de passe d'un compte Oracle à l'aide d'AdoCommand de Delphi. Veuillez à adapter ces instructions en fonction de votre environnement Oracle spécifique et des besoins de votre application.



Capture affichant le code source du changement de mot de passe d'un compte Oracle sous Delphi.

6: AFFICHER LES COMPTES

Pour afficher les comptes Oracle avec Delphi, vous pouvez utiliser des composants de base de données tels que TADOQuery ou TADODataSet pour exécuter une requête SQL et récupérer les informations des comptes Oracle. Voici une description des étapes générales :

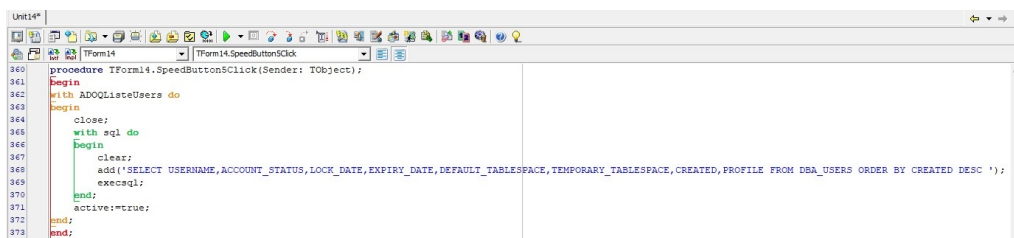
1. Créer une connexion à la base de données Oracle : Ajoutez un composant de connexion ADO à votre formulaire Delphi, par exemple, TADOConnection. Configurez les paramètres de connexion tels que le nom d'utilisateur, le mot de passe, l'hôte et le port Oracle pour vous connecter à la base de données Oracle.
2. Créer un composant de requête : Ajoutez un composant de requête ADO à votre formulaire Delphi, par exemple, TADOQuery ou TADODataSet. Associez ce composant à la connexion ADO créée précédemment en définissant sa propriété Connection sur le composant TADOConnection.
3. Écrire une requête SQL pour sélectionner les informations des comptes Oracle : Utilisez la propriété SQL de votre composant de requête ADO pour écrire une requête SQL qui sélectionne les informations des comptes Oracle.
4. Exécuter la requête et récupérer les résultats : Appelez la méthode Open ou Active du composant de requête ADO pour exécuter la requête SQL et récupérer les résultats dans un ensemble de données. Vous pouvez utiliser des boucles pour parcourir les enregistrements et afficher les informations des comptes Oracle dans votre application Delphi. Par exemple :

 Copy code

```
procedure TForm14.SpeedButton5Click(Sender: TObject);
begin
  with ADOQListeUsers do
  begin
    close;
    with sql do
    begin
      clear;
      add('SELECT USERNAME,ACCOUNT_STATUS,LOCK_DATE,EXPIRY_DATE,DEFAULT_TABLESPACE,TEMPORARY_TABLESPACE,CREA
      add('FROM DBA_USERS');
      add('ORDER BY CREATED DESC');
      execsql;
    end;
    active:=true;
  end;
end;
```

- Assurez-vous d'adapter la requête SQL en fonction de votre schéma de base de données Oracle et des informations que vous souhaitez afficher. Vous pouvez également personnaliser l'affichage des informations des comptes Oracle en utilisant des contrôles ou des composants Delphi appropriés, tels que des listes, des tableaux ou des grilles.

- En suivant ces étapes, vous pourrez afficher les comptes Oracle dans votre application Delphi en utilisant des composants ADO pour exécuter des requêtes SQL et récupérer les résultats.



Capture affichant le code source des comptes oracle sous Delphi.

7: CRÉER UN RÔLE

Cette partie,est en cours de construction .

8: SUPPRIMER UN RÔLE

Cette partie,est en cours de construction .

EN SAVOIR PLUS

Prochainement,Vous trouverez d'autres tutoriels sur autres options d'administration d'une base de données Oracle sous Delphi.

Plus d'Actions

1. Créer un compte
2. Activer un compte
3. Désactiver un compte
4. Supprimer un compte
5. Changer mot de passe
6. Afficher les comptes
7. Créer un rôle
8. Supprimer un rôle

En savoir plus



Dernière modification Dim, 16 juillet 202023



Plus d'Actions

1. *Créer un compte*
 2. *Activer un compte*
 3. *Désactiver un compte*
 4. *Supprimer un compte*
 5. *Changer mot de passe*
 6. *Afficher les comptes*
 7. *Créer un rôle*
 8. *Supprimer un rôle*
- En savoir plus*