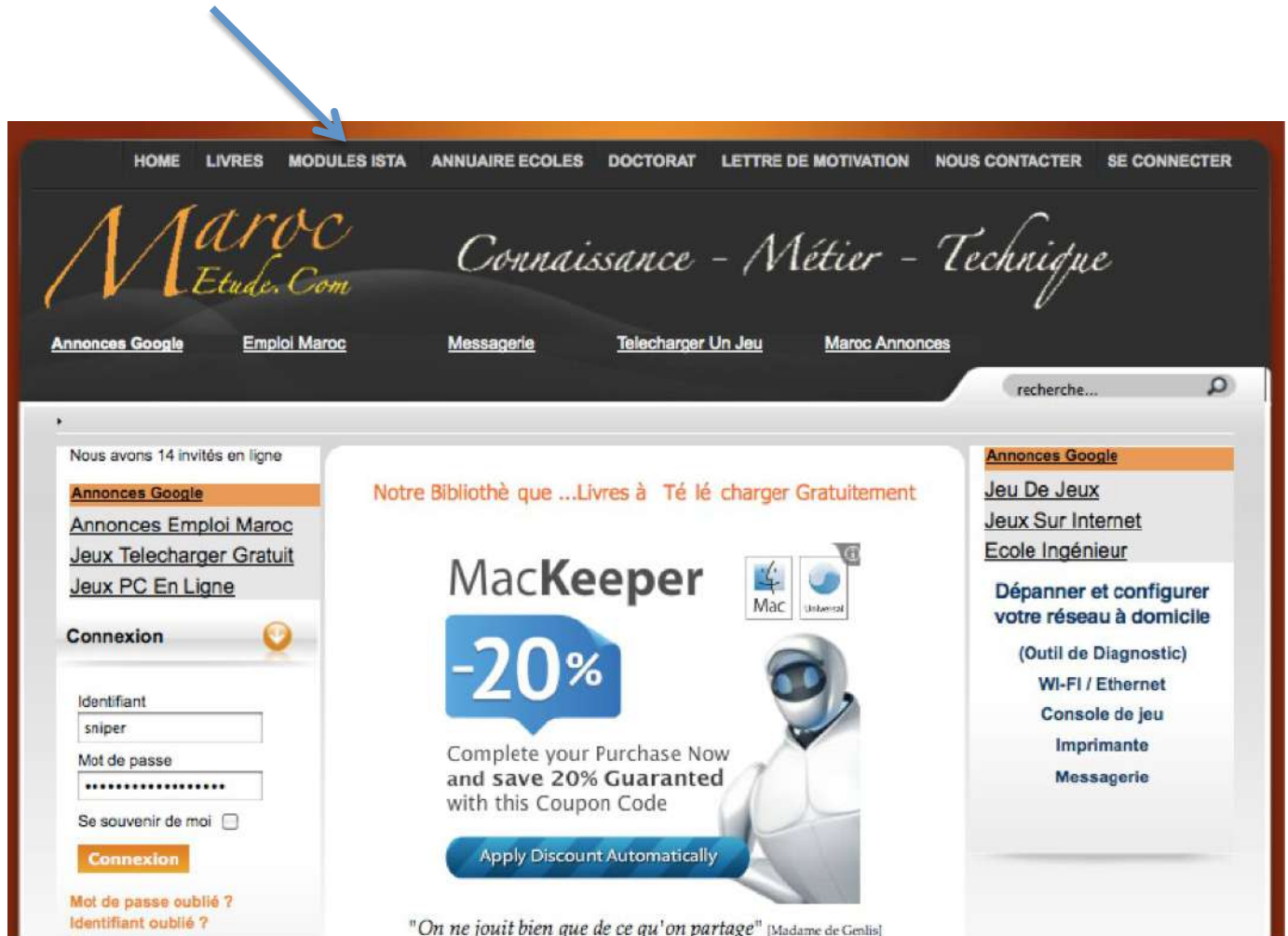


PORTAIL DE LA FORMATION PROFESSIONNELLE AU MAROC

Télécharger tous les modules de toutes les filières de l'OFPPT sur le site dédié à la formation professionnelle au Maroc : www.marocetude.com

Pour cela visiter notre site www.marocetude.com et choisissez la rubrique :

[MODULES ISTA](#)



The image shows a screenshot of the website www.marocetude.com. The top navigation bar includes links for HOME, LIVRES, **MODULES ISTA**, ANNUAIRE ECOLES, DOCTORAT, LETTRE DE MOTIVATION, NOUS CONTACTER, and SE CONNECTER. The main header features the logo "Maroc Etude.Com" and the tagline "Connaissance - Métier - Technique". Below the header, there are links for "Annonces Google", "Emploi Maroc", "Messagerie", "Telecharger Un Jeu", and "Maroc Annonces". A search bar is located in the top right corner.

The sidebar on the left contains the following links:

- Nous avons 14 invités en ligne
- Annonces Google**
- Annonces Emploi Maroc
- Jeux Telecharger Gratuit
- Jeux PC En Ligne
- Connexion

The main content area features a promotional banner for MacKeeper with a -20% discount. The banner includes the text "Notre Bibliothèque que ...Livres à Télé charger Gratuitement", "MacKeeper -20%", "Complete your Purchase Now and save 20% Guaranteed with this Coupon Code", and "Apply Discount Automatically". Below the banner is the quote: "On ne jouit bien que de ce qu'on partage" [Madame de Genlis].

The sidebar on the right contains the following links:

- Annonces Google**
- Jeu De Jeux
- Jeux Sur Internet
- Ecole Ingénieur
- Dépanner et configurer votre réseau à domicile (Outil de Diagnostic)
- Wi-Fi / Ethernet
- Console de jeu
- Imprimante
- Messagerie

Déploiement d'une application Client Serveur

Le but de ce TP est de déployer une application utilisant une base de données.

1. Créer une page qui permet l'affichage de la liste des client dans un Datagrid
2. Créer un fichier de configuration pour le stockage du nom de la base de données et du nom du serveur. Ces informations seront utilisées par l'application pour établir une connexion
3. Pour créer ce fichier Menu Projet/Ajouter un nouvel élément / fichier de configuration de l'application. Ce fichier prend par défaut le nom app.config.
4. A votre avis, à quoi sert ce fichier ?
5. Dans ce fichier saisir les éléments suivants :

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <appSettings>
    <add key="NomServeur" value="Nom_serveur" />
    <add key="NomBaseDeDonnees" value="BD" />
  </appSettings>
</configuration>
```

6. Déclarer deux variables globales :

```
Public NS as String ' Pour stocker le nom du serveur
Public NBD as String ' Pour stocker le nom de la base de
données
```

7. Au démarrage de l'application, récupérer le nom du serveur et le nom de la base de données à partir du fichier app.config et les stocker respectivement dans les variables NS et NBD :

```
Imports System.Configuration
...
NBD = ConfigurationSettings.AppSettings.Get("NomBaseDeDonnees")
NS = ConfigurationSettings.AppSettings.Get("NomServeur")
```

8. Utiliser ces informations dans la chaîne de connexion :

9. Création de la base de données

10. Générer le script de la base de données dans le fichier mabase.sql dans la racine de votre projet

11. au niveau de votre projet d'installation, ajouter le fichier de votre base de données mabase.sql

12. Dans votre projet, créer un élément Class Installer (Projet / Ajouter un nouvel élément / Class Installer)

```
Public Overrides Sub Install(ByVal Savestate As
System.Collections.IDictionary)
```

13. créer la procédure qui permet la création de notre base de données

```
Protected Sub CreationBase(ByVal nomBase As String)
Try
'Création de la base de données
ExecuteSql("master", "CREATE DATABASE " & nomBase)
'Exécution du script de création des tables
ExecuteSql(nomBase, RequeteSQL("sql.txt"))
'Gestion des exceptions
Catch ex As Exception
MsgBox("Une exception à été levée : " & ex.Message)
Throw ex
End Try
End Sub
```

14. RequeteSQL :

```
Private Function RequeteSQL(ByVal Name As String) As String
Try
'Récupère l'assembly de l'application
Dim Asm As System.Reflection.Assembly =
System.Reflection.Assembly.GetExecutingAssembly
'Récupère le chemin du fichier contenant le script SQL
Dim chemin As String = Asm.Location.Substring(0,
Asm.Location.Length - Asm.GetName.Name.Length - 8) + "\maBase.sql"
Dim monReader As StreamReader = New StreamReader(chemin)
'Lecture du fichier SQL
Return monReader.ReadToEnd()
Catch ex As Exception
MsgBox("In GetSQL: " & ex.Message)
Throw ex
End Try
End Function
```

15. ExecuteSql :

```
Private Sub ExecuteSql(ByVal DatabaseName As String, ByVal Sql As String)
'Récupération des informations saisies par l'utilisateur
Dim UserDB As String = Me.Context.Parameters.Item("UserSQL")
Dim PasswordDB As String =
Me.Context.Parameters.Item("PasswordSQL")
```

```
Dim ServerDB As String = Me.Context.Parameters.Item("ServerName")
'Concaténation des identifiants dans la chaine de connexion
Dim chaine As String = "packet size=4096;user id=" & UserDB &
";data source=" & ServerDB & ";persist security info=True;initial
catalog=master;password=" & PasswordDB & ""
Dim SqlConnection1 = New System.Data.SqlClient.SqlConnection
SqlConnection1.ConnectionString = chaine
Dim Command As New SqlCommand(Sql, SqlConnection1)
Command.Connection.Open()
Command.Connection.ChangeDatabase(DatabaseName)
Try
    'Execution de la requete (script SQL de création)
    Command.ExecuteNonQuery()
Finally
    'Fermeture de la connexion
    Command.Connection.Close()
End Try
End Sub
```

16. maintenant il suffit d'appeler la procédure de création de base de données au niveau de

Public Overrides Sub Install(...)