

Le développement des applications

Gestion des frais d'hébergement

Propriétés	Description
Intitulé long	Observation puis modification d'une procédure de l'application du contexte Gestion des formations.
Formation concernée	Gestion des Systèmes d'Information en classe de terminale GSI
Matière	Gestion des Systèmes d'Information
Présentation	Ce support de cours est destiné aux enseignants, il propose une démarche pour aborder le développement des applications en classe de terminale GSI. Il s'appuie sur le contexte de niveau II Gestion des formations. L'élève est invité à observer le résultat produit par l'application puis à utiliser les outils de débogage pour analyser comment ce résultat est produit.
Notions	2. L'adaptation de l'application 2.1. Prise en charge du dialogue homme-machine Interface homme-machine. - Maquette. - Ergonomie. - Contrôle graphique. - Événement 2.2. Exploitation des données Structure de données : tableau, indice, jeu d'enregistrements. 2.3. Programmation des traitements - Procédure, fonction, paramètre. - Structure de contrôle. - Mise au point de l'application.
Transversalité	Le cours d'algorithmique de la classe de première
Pré-requis	Application informatique, utilisateurs, base de données, requête SQL.
Outils	Logiciel de développement d'application Microsoft Access, le lecteur Flash pour les démonstrations.
Mots-clés	Application, programmation, procédure, fonction, débogage, algorithmique
Durée	8 heures
Auteur(es)	Eric Deschaintre, Christian Draux, Jean-Philippe Pujol
Version	v 1.0
Date de publication	Janvier 2007

Ce support propose une approche de la logique algorithmique d'une application informatique par l'utilisation de la fonction de débogage d'un environnement de développement.

L'exercice proposé doit permettre de montrer le lien entre une action d'un utilisateur sur une application et le traitement qui réalise cette action. En l'occurrence il s'agit de comprendre comment un état de remboursement de frais de déplacement est produit à la demande du responsable financier.

L'exemple est basé sur les documents fournis dans le contexte de système d'information « Gestion des formations à la SNCF ». L'application utilisée dans ce contexte utilise le logiciel Access côté client et une base de données (MySQL ou SQL Server) comme serveur de données.

Les questions posées dans cet exercice sont conçues pour être adressées aux élèves, mais il s'agit souvent d'engager une réflexion afin de préparer une réponse qui est fournie par le professeur.

Quelques réflexions didactiques

Le programme d'enseignement prévoit de partir de l'observation d'une application existante dans un contexte de système d'information précis. L'élève se trouve ainsi face à une application informatique qui est un objet technique complexe difficile à appréhender dans sa globalité. Ce support propose une démarche permettant de réduire cette complexité.

Il s'agit notamment de permettre aux élèves de faire progressivement le lien entre l'utilisation d'une application et sa programmation. On propose à l'élève de lire un programme en observant comment celui-ci s'exécute. On aura pour cela recours à l'environnement de développement de l'application, plus précisément à son éditeur de programme et à son débogueur.

Dans un premier temps il est important que l'élève soit mis en situation d'utiliser l'application pour réaliser des tâches précises comme s'il était un acteur en charge de la gestion des formations. Ce temps d'apprentissage doit lui permettre de connaître les fonctionnalités de l'application ainsi que la meilleure façon de l'utiliser.

Dans un deuxième temps l'élève est invité à « passer de l'autre côté de l'écran » pour observer comment les actions de l'utilisateur sont prise en charge dans l'application. Ce transfert doit être réalisé mentalement par l'élève : il doit voir qu'en tant qu'utilisateur il dialogue avec l'application (lecture des écrans, saisie de données et clics..); alors qu'en tant qu'informaticien il se situe « dans » l'application et décrit ce qu'elle doit faire en réponse aux actions de l'utilisateur. Il faudra savoir adopter alternativement ces deux rôles pour comprendre l'exécution de l'application : agir en tant qu'utilisateur d'abord puis observer les effets dans l'application ; ou bien modifier l'application et observer les conséquences au niveau de l'utilisateur.

Dans un troisième temps, après avoir compris comment est prise en charge telle ou telle action de l'utilisateur, on pourra modifier le comportement de l'application pour qu'elle réponde à un nouveau besoin exprimé par ses utilisateurs. Cette étape constitue assurément une évaluation des deux étapes précédentes car il est nécessaire de bien connaître une application pour savoir interpréter un nouveau besoin et il faut comprendre les traitements associés pour être à même de les modifier.

Programmer une application suppose de maîtriser le codage dans un langage de programmation alors qu'observer une application existante peut permettre de révéler le sens en s'affranchissant des contraintes de syntaxe.

Cet exercice propose une mise en œuvre pratique de ces quelques principes. Il s'adresse aux enseignants qui sauront en faire le meilleur usage auprès de leurs élèves, au besoin en modifiant la démarche, voire en adaptant l'application ou la base de données.

Utiliser l'application

Cet exercice s'intéresse à la gestion des frais de d'hébergement et de déplacement qui sont dus aux personnes qui ont suivi une formation.

Démonstration de l'application

Dans un premier temps on fait une démonstration de l'utilisation de l'application à l'aide d'un vidéoprojecteur. Un exemple de cette démonstration est proposé dans le fichier <etatDeFrais.htm>, les élèves peuvent ainsi la visualiser plusieurs fois si nécessaire. Il est toutefois souhaitable d'organiser un dialogue avec la classe pendant la démonstration en posant des questions sur le sens du travail réalisé. Dans la suite on propose des exemples de questions et de réponses.

- Une démonstration *flash* permettant d'illustrer cette partie est associée à ce document : fichier <etatDeFrais.htm> (nécessite le fichier <etatDeFrais.swf>).

Qui a besoin de produire un état des frais à rembourser après le déroulement d'une session de formation ?

Le responsable de la gestion financière des formations, voici son entretien tel qu'il apparaît dans le dossier du contexte de système d'information « Gestion des formations » :

Un responsable de la gestion financière

Quel est votre rôle au sein de ce service ?

Je m'occupe de la gestion financière des formations. Je procède aux remboursements des frais des stagiaires à partir des documents que ces derniers me fournissent. Je réalise également un certain nombre de statistiques, relatives aux bénéficiaires et aussi aux coûts des sessions proposées. Je peux également être amené à suivre le dossier des agents et communiquer ces informations aux responsables des services.

Qu'est-ce que vous remboursez aux agents ?

Pas les frais de déplacement bien sûr, puisque la SNCF leur fournit le billet ! Mais il nous faut rembourser les frais d'hébergement lorsque le lieu de formation est trop éloigné du domicile pour permettre à l'agent de rentrer chez lui le soir.

Comment déterminez-vous alors ce droit à remboursement ?

C'est une règle de gestion simple : un agent est supposé résider dans la ville où il travaille. Nous connaissons la distance qu'il y a entre cette résidence et le lieu de formation : si elle est supérieure à 60 km, nous estimons que l'agent a droit à loger sur place.

Et pour les agents qui n'habitent pas près de leur lieu de travail ?

Je prends par exemple un agent qui travaille en gare d'Agen : Agen est sa résidence administrative, c'est là qu'il se rend tous les jours. S'il préfère habiter à 40 km de là, c'est son choix ; mais cela ne nous regarde pas, il est supposé résider à Agen !

Qu'est-ce que les agents doivent fournir pour justifier leurs frais ?

Ils doivent obligatoirement nous renvoyer les factures justificatives pour l'hébergement.

Vous remboursez la totalité de la facture ?

On ne prend pas en charge les nuitées dans les « 5 étoiles » ! Il y a un plafond, révisable annuellement, qui est actuellement fixé à 54 euros. Nous prenons seulement en charge les nuits entre deux jours ouvrés, c'est à dire celles des lundi, mardi, mercredi et jeudi soir ; le vendredi soir, l'agent retourne à son domicile.

Comment se fait ce remboursement ?

Nous envoyons un chèque à l'intéressé dans les 30 jours qui suivent la fin de la session, si nous avons reçu les justificatifs bien sûr.

En quoi consistent les statistiques que vous réalisez ?

D'abord, nous devons savoir quelles formations ont été suivies par chaque agent. Nous cherchons également à mesurer l'intérêt des formations proposées. Nous réalisons régulièrement des enquêtes internes auprès des chefs de service pour évaluer les besoins de formation.

Est-ce que l'application informatique vous apporte une aide ?

L'application actuelle me permet de calculer les montants à rembourser à chaque agent ayant participé à une formation. C'est déjà important, car c'est un calcul long et fastidieux !

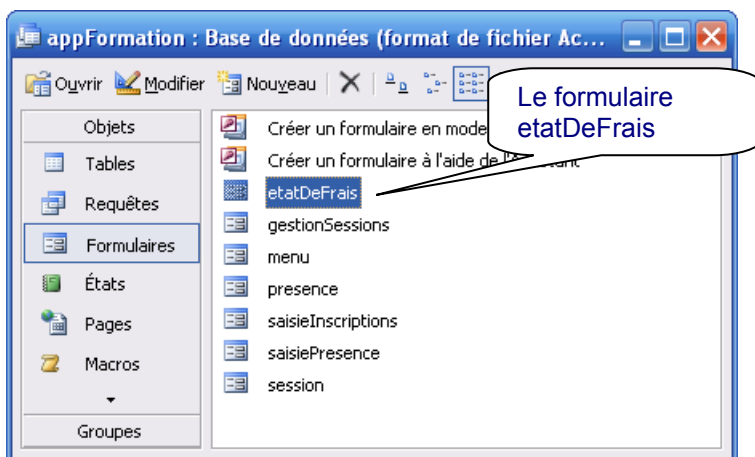
Pour pouvoir produire un état des frais à rembourser pour les stagiaires d'une session de formation quelles informations doivent avoir été saisies au préalable dans l'application ?

Il faut avoir naturellement créé la session de formation (menu « Session ») ;

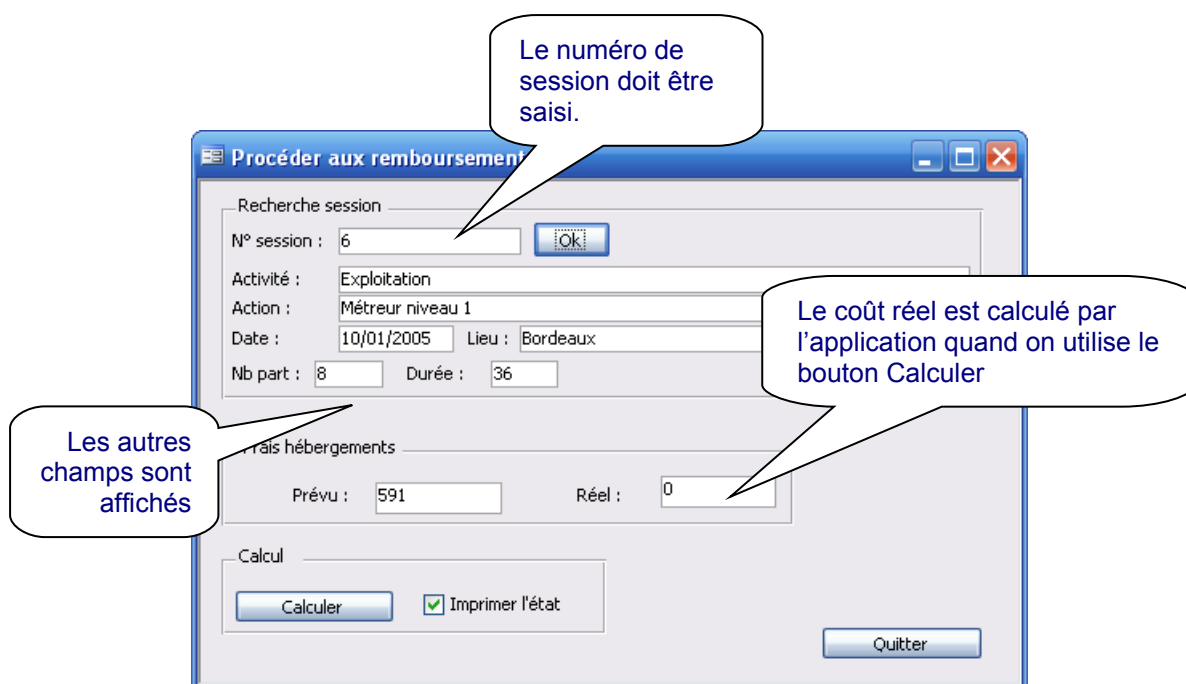
Puis avoir inscrit des agents à cette formation (menu « Inscription ») ;

Enfin, après la formation, avoir saisi les présences ainsi que le montant des frais d'hébergement communiqué par les agents à l'issue de la formation. (menu « Présence »).

Quel formulaire de l'application permet de demander la production d'un état des frais à rembourser ?



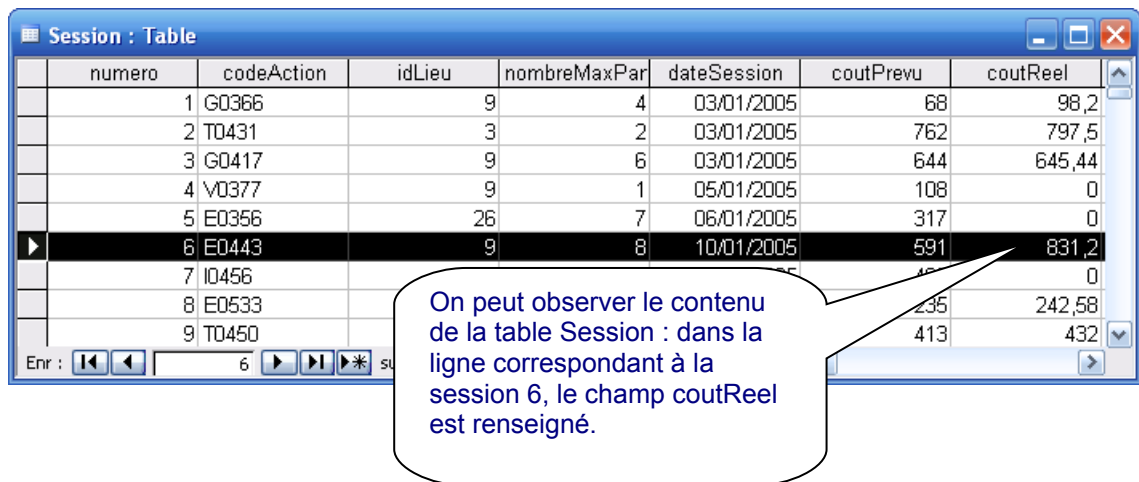
Parmi les informations présentes dans les champs du formulaire pour la session 6, lesquelles sont saisies, affichées, ou calculées puis affichées ?



Le coût réel de la formation est-il enregistré après avoir été calculé ?

Oui. Pour s'en persuader il suffit de calculer le coût d'une session de formation, de fermer le formulaire puis de le réouvrir en saisissant le même numéro de formation : le coût réel est immédiatement renseigné cette fois, sans qu'il soit nécessaire d'utiliser le bouton Calculer.

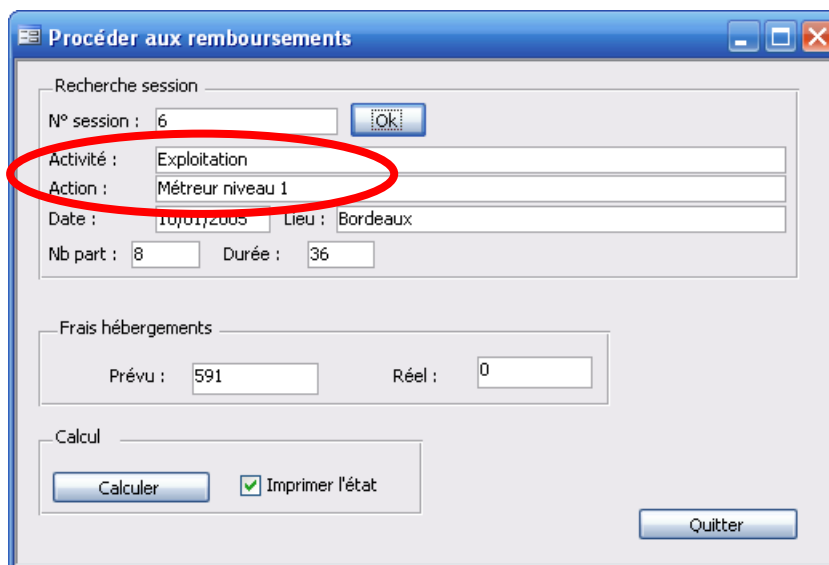
Où le coût réel de la session de formation est-il enregistré ?



numero	codeAction	idLieu	nombreMaxPar	dateSession	coutPrevu	coutReel
1	G0366	9	4	03/01/2005	68	98,2
2	T0431	3	2	03/01/2005	762	797,5
3	G0417	9	6	03/01/2005	644	645,44
4	V0377	9	1	05/01/2005	108	0
5	E0356	26	7	06/01/2005	317	0
6	E0443	9	8	10/01/2005	591	831,2
7	I0456					0
8	E0533				235	242,58
9	T0450				413	432

Il peut être utile de supprimer cette valeur dans la table session (ceci correspondant à la valeur NULL) pour pouvoir réitérer la présentation sur le même exemple de session.

Prenons la session de formation n° 6, à quelle activité est-elle associée, à quelle action de formation ?



Procéder aux remboursements

Recherche session

N° session : 6 [OK]

Activité : Exploitation

Action : Mètreur niveau 1

Date : 10/01/2005 Lieu : Bordeaux

Nb part : 8 Durée : 36

Frais hébergements

Prévu : 591 Réel : 0

Calcul

Calculer Imprimer l'état

Quitter

Quelle est la signification des autres informations présentes dans le formulaire ?

Date : date de début de la session formation
Durée : durée de la session formation exprimée en heures de formation
Prévu : budget prévu pour cette session de formation
Réel : coût réel à rembourser aux agents pour cette session de formation (calculé par l'application quand on utilise le bouton Calculer)

Si on ne coche pas la case « Imprimer l'état », que se passe-t-il quand on utilise le bouton Calculer ?

Le coût réel à rembourser aux agents pour cette session de formation est calculé puis affiché dans le champ « Réel » du formulaire.

Comment interpréter les informations suivantes dans l'état produit : forfait nuitée, seuil de remboursement ? (voir l'interview du responsable financier)

(Voir la réponse sur page suivante)

Quel agent n'a pas parcouru suffisamment de kilomètres pour avoir droit à un remboursement ?

Tout agent dont le nombre de kms parcourus est inférieur à 60, ils sont 3 :
Stéphane Maxiola a parcouru 48 km pour se rendre à Bordeaux depuis Langon où il travaille habituellement. On considère qu'il a pu rentrer chez lui. Sabine Dageville et Christophe Ginocchio, étant sur place à Bordeaux, n'ont pas eu à se déplacer.

Etat des remboursements des frais d'hébergements

Référence Action **E0443**
Référence session **6**
Lieu **Bordeaux**
Montant budgété **591**
Durée **36**
Nombre participant **8**

Intitulé de la formation **Métreur niveau 1**

Forfait nuitée **54**

Seuil de remboursement **60**

Montant maximum auquel une nuit d'hôtel est remboursée

Nombre minimum de kilomètres à parcourir pour avoir droit au remboursement d'une nuit d'hôtel

Code	Nom	Prénom	Présent (O/N)	Résidence administrative	Distance	Frais engagés	Droits à Remb.	Montant	
6560748	Gomez	Sandrine	<input checked="" type="checkbox"/>	Nontron	156	235	<input checked="" type="checkbox"/>	216	
6851923	Nguyen-Ngoc	Annie-Claire	<input checked="" type="checkbox"/>	Rochefort	132	202	<input checked="" type="checkbox"/>	202	
7140024	Dageville	Sabine	<input checked="" type="checkbox"/>	Bordeaux	0	0	<input type="checkbox"/>	0	
7430099	Maxiola	Stéphane	<input checked="" type="checkbox"/>	Langon	48	0	<input type="checkbox"/>	0	
7803406	Pasti	Stéphanie	<input checked="" type="checkbox"/>	Biarritz	197	198	<input checked="" type="checkbox"/>	198	
7950856	Ginocchio	Christophe	<input checked="" type="checkbox"/>	Bordeaux	0	0	<input type="checkbox"/>	0	
7967964	Wetterwald	Dominique	<input checked="" type="checkbox"/>	Mont de Marsan	121	359	<input checked="" type="checkbox"/>	216	
Nombre de présents							7	Total	832
Taux de présence							0,875	Ecart	241

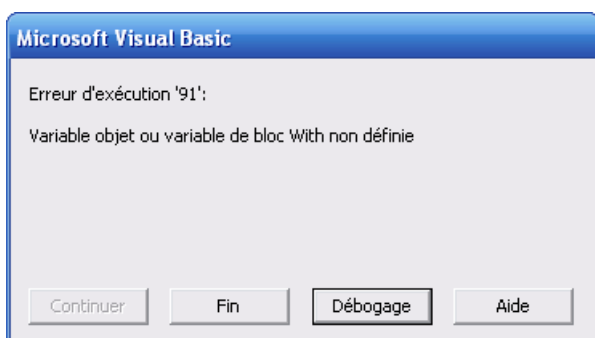
Quelles améliorations pourrait-on envisager pour rendre plus lisible cet état de frais ?

- Faire figurer les unités des valeurs affichées (euros, heures, kilomètres).
- Rappeler la date de la session de formation
- L'intitulé « Nombre de participant » devrait être « Nombre maximum de participants »

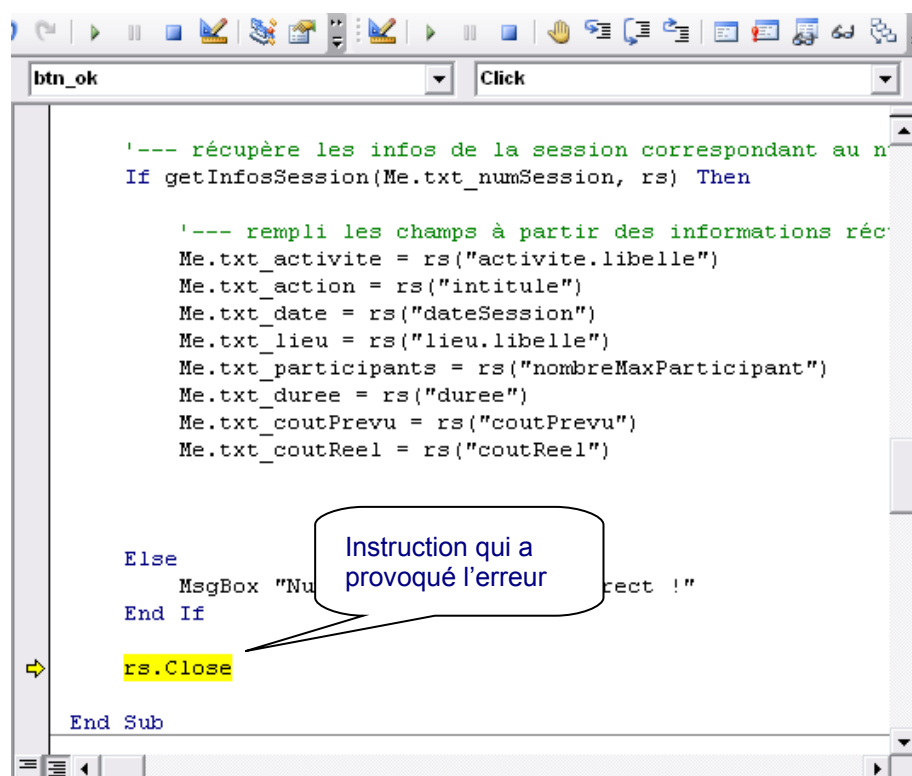
Accéder au code de l'application

A la place d'un numéro de session on saisit par erreur des lettres dans le champ N° session, que se passe-t-il quand on valide la saisie avec le bouton Ok ?

Le message « Numéro de session incorrect » s'affiche normalement mais quand on confirme la lecture du message en cliquant sur Ok, un message d'erreur d'exécution s'affiche :



En utilisant le bouton Débogage on accède directement au code de l'application à la ligne qui provoque l'erreur :



Nous chercherons à élucider plus loin la cause de cette erreur puis à corriger le programme.

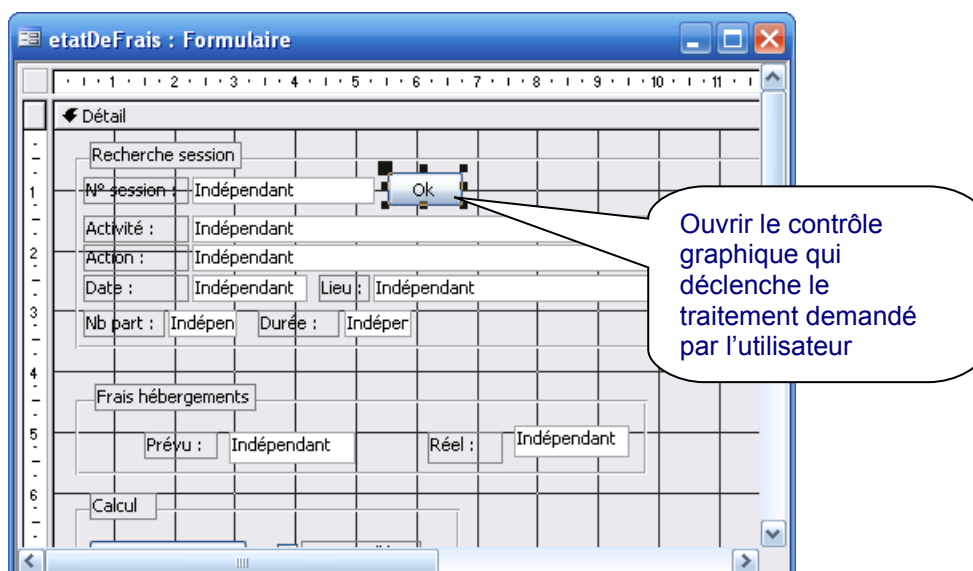
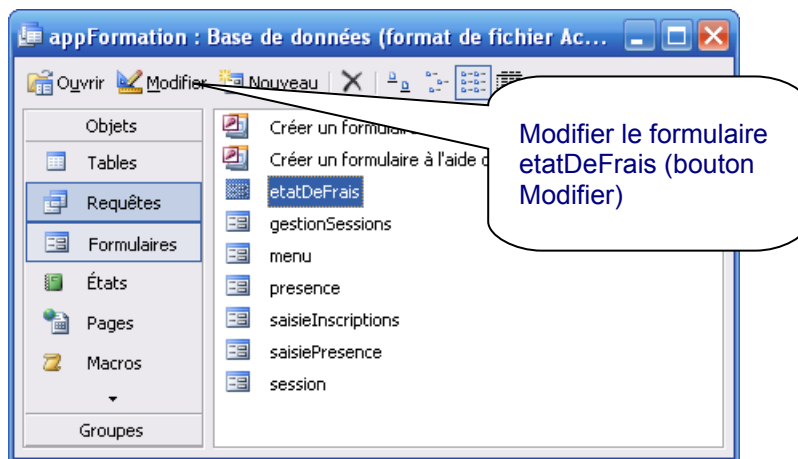
Analyser l'application

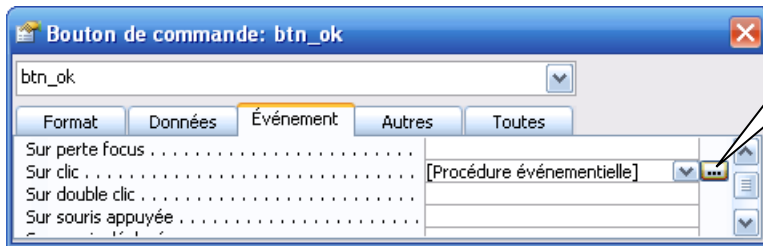
Pourquoi analyser l'application ? Essentiellement parce ce qu'il peut être nécessaire de modifier son comportement. Pour être réalisées, les modifications souhaitées sur l'état de frais impliquent de modifier cet objet dans l'application. Nous pouvons aussi, par exemple, corriger le problème lié à la saisie d'un texte dans le numéro de session : pourquoi cela arrive t-il ? Comment modifier l'application pour que cela ne se produise plus ?

- Une démonstration *flash* permettant d'illustrer cette partie est associée à ce document : fichiers <pasApasDetaille.htm> et <pasApasDetaille.swf>.

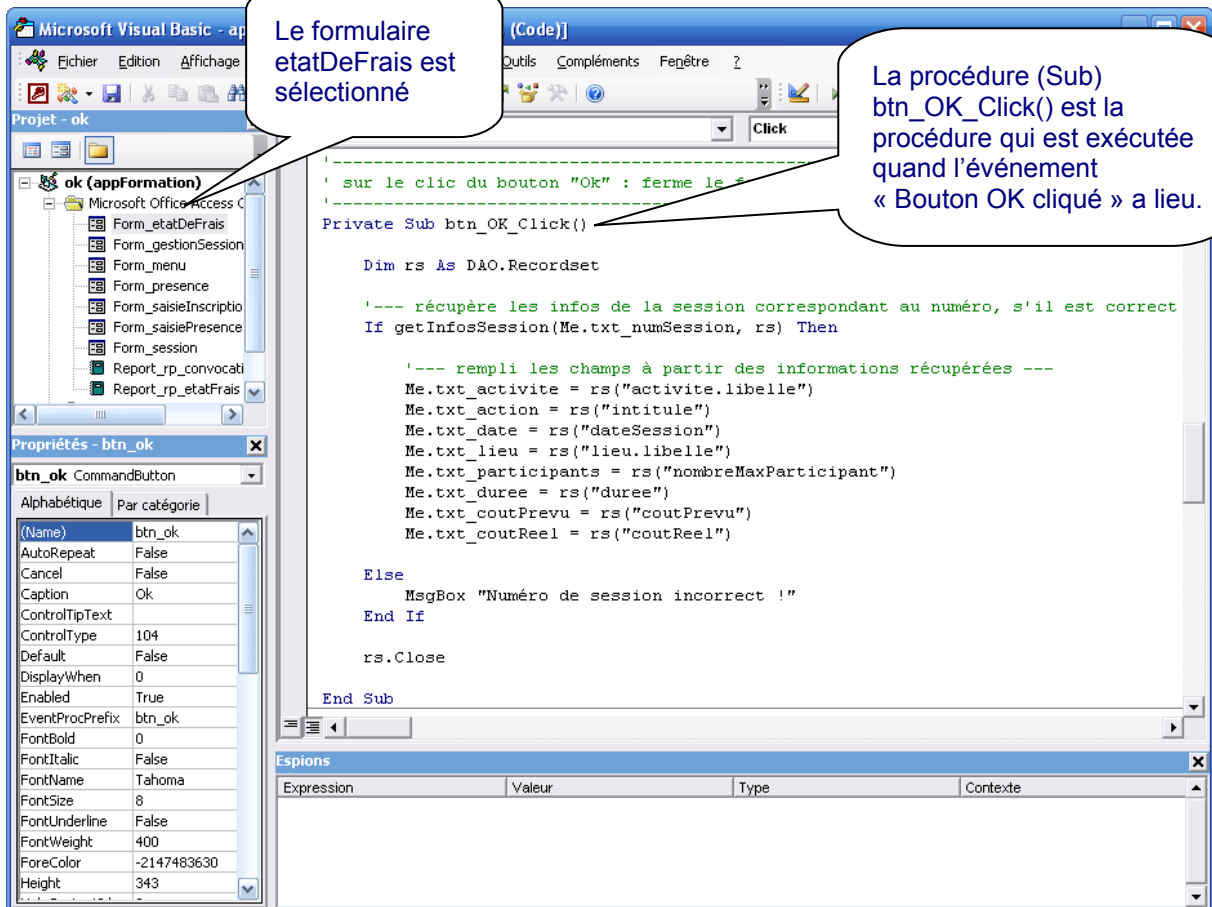
Nous nous intéressons au formulaire de gestion des remboursements des frais d'une session de formation. Posons-nous quelques questions à son propos :

Quel programme est exécuté quand on clique sur le bouton Ok après avoir saisi un numéro de session ?





Ouvrir la procédure qui s'exécute quand on clique sur le bouton (sur clic)

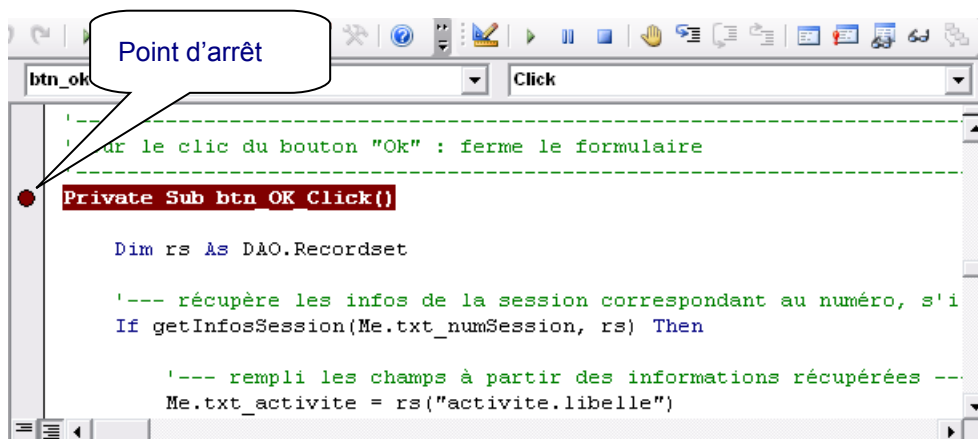


Suivre l'exécution de l'application pas à pas.

Comment observer l'exécution de ce programme ?

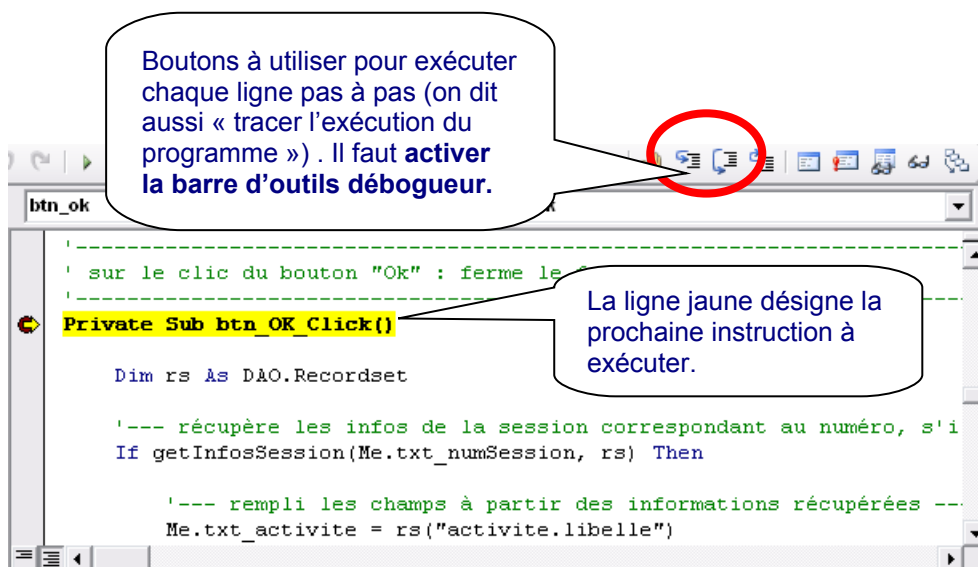
En plaçant un « point d'arrêt » dans la marge à côté de l'instruction sur laquelle l'exécution de l'application doit s'arrêter (un clic pour placer le point, un autre pour l'enlever).

Désormais, à chaque fois que cette procédure sera exécutée, on sera placé dans l'éditeur de programme et on pourra suivre pas à pas l'exécution des instructions de l'application au rythme qui nous convient.





Après avoir placé le point d'arrêt on peut fermer l'éditeur et toutes les fenêtres ouvertes (sauf la fenêtre de l'application elle-même « appFormations »).

On ouvre ensuite le formulaire etatDeFrais pour saisir un numéro de session (ici on choisit la session n°6) et utiliser le bouton OK. Dès que le bouton est utilisé l'éditeur s'affiche et l'exécution de l'application est interrompue comme prévu sur notre point d'arrêt.



Voici le rôle des boutons d'exécution :

-  « Pas à pas principal » (touche Majuscule F8) : exécute chaque ligne en restant dans la procédure en cours.
-  « Pas à pas détaillé » (touche F8) : exécute chaque ligne en entrant dans les procédures appelées

Pour lire la valeur d'une variable il suffit d'amener le pointeur de la souris sur le nom de cette variable pendant l'exécution pas à pas.

Observer à la trace l'exécution de l'application.

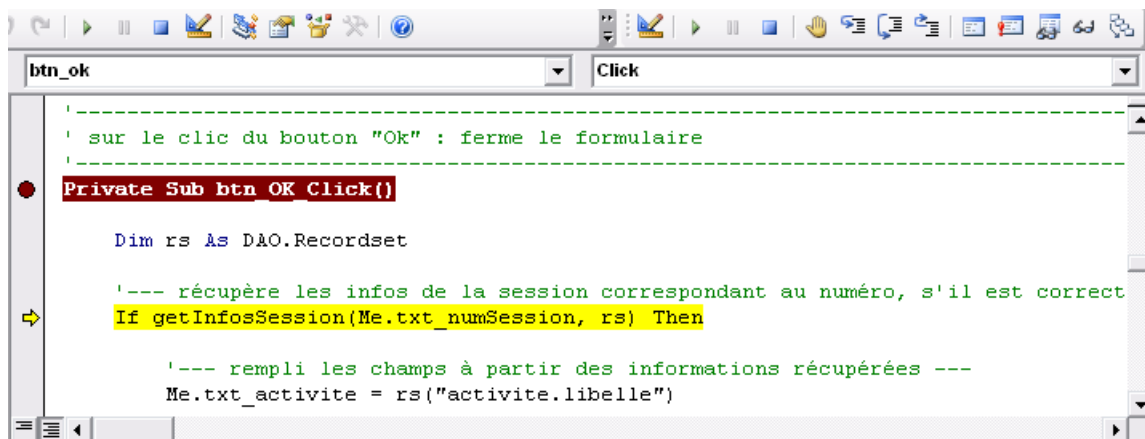
A quoi correspond la variable nommée « rs » ?

Cette variable est déclarée avec la ligne suivante :
Dim rs As DAO.Recordset

La variable « rs » est un jeu d'enregistrements (*Record Set* en anglais). Une telle variable est destinée à contenir toutes les lignes de résultats retournées par une requête SQL.

Quelle est la première ligne exécutable de la procédure ?

Il suffit de cliquer sur le bouton « Pas à pas principal » (ou bien utiliser les touches majuscule-F8) pour atteindre la première ligne à exécuter.



```
Private Sub btn_OK Click()  
  
    Dim rs As DAO.Recordset  
  
    '--- récupère les infos de la session correspondant au numéro, s'il est correct  
    If getInfosSession(Me.txt_numSession, rs) Then  
  
        '--- rempli les champs à partir des informations récupérées ---  
        Me.txt_activite = rs("activite.libelle")  
  
    End If  
  
End Sub
```

De quel type d'instruction s'agit-il ?

Il s'agit d'une alternative (instruction « If »).

D'après le commentaire situé juste au dessus de cette ligne, quel est le rôle de la fonction *getInfosSession()* ? De l'alternative (If) ?

Cette fonction récupère (dans la base de données) les informations qui concernent la session correspondant au numéro saisi et vérifie si ce numéro est correct.

L'alternative permet d'exécuter des instructions différentes selon que le numéro saisi est correct ou non.

Que doit-il se passer si le numéro saisi n'est pas correct ?

L'instruction « MsgBox "Numéro de session incorrect !" » (clause « Else » de l'alternative)

Quelle est la valeur de la variable « Me.txt_numSession » au moment de l'appel de la fonction *getInfosSession()* ?

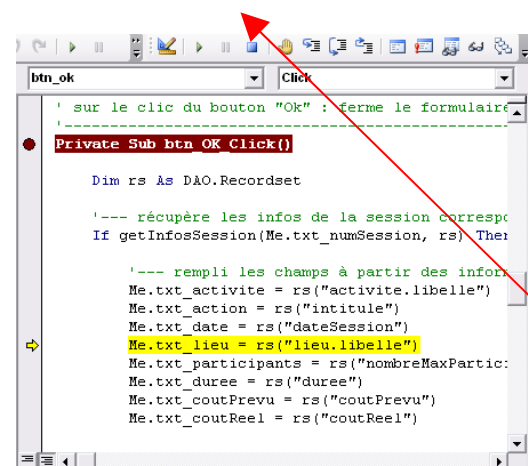
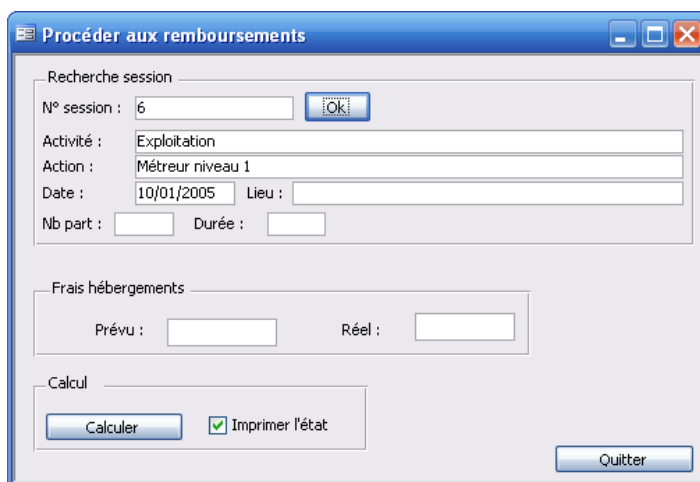
Il suffit d'amener le pointeur de la souris sur le nom de cette variable dans l'instruction alternative « If », pour lire la valeur contenue actuellement pas la variable. Cette valeur est 6.

Le mot « Me » désigne le formulaire en cours d'exécution. « txt_numSession » est le nom du champ dans lequel on saisit le numéro de session.

Que font les instructions suivantes ?

Elles affichent dans les champs du formulaire les données qui concernent la session de formation n°6.

Pour s'en persuader, il suffit d'observer en même temps l'exécution du code de la procédure et le formulaire de calcul des frais :



```
Private Sub btn_OK Click()  
  
Dim rs As DAO.Recordset  
  
'--- récupère les infos de la session correspo  
If getInfosSession(Me.txt_numSession, rs) Then  
  
    '--- remplit les champs à partir des info  
    Me.txt_activite = rs("activite.libelle")  
    Me.txt_action = rs("intitule")  
    Me.txt_date = rs("dateSession")  
    Me.txt_lieu = rs("lieu.libelle")  
    Me.txt_participants = rs("nombreMaxPartic  
    Me.txt_duree = rs("duree")  
    Me.txt_coutPrevu = rs("coutPrevu")  
    Me.txt_coutReel = rs("coutReel")  
  
End Sub
```

A chaque pression sur la touche F8 (pas à pas principal), l'instruction d'affectation place la valeur dans le champ correspondant du formulaire.

L'expression « rs(« intitulé ») » retourne la valeur de ce champ contenue dans le jeu d'enregistrements.

Arrivé à l'instruction « End Sub », toutes les instructions nécessaires au traitement d'un clic sur le bouton Ok ont été exécutées.

Comment voir ce que fait la fonction getInfosSession() ?

On peut à nouveau saisir le numéro de session dans le formulaire puis suivre pas à pas l'exécution de l'application (pas à pas détaillé – touche F8).

On moment d'exécuter l'alternative, on se retrouve aussitôt dans le code de la fonction getInfosSession(). En fait, nous avons suivi l'appel de cette fonction puisque le résultat retourné par cette fonction est nécessaire pour évaluer la condition de l'alternative. Nous allons maintenant suivre ses instructions.

```

(Général) | getInfosSession
' fonction getInfosSession : récupère les informations d'une se
' retour : vrai si une session existe pour ce numéro, faux si
' paramètres :
'     p_numeroSession (entrée) : reçoit le numero de la sessi
'     rsSession (sortie) : récupère le résultat de la recher
-----
Public Function getInfosSession(ByVal p_numeroSession As String

'--- contrôle si le numero de session passé en paramètre es
If estEntier(p_numeroSession) Then

'--- requête pour récupérer les informations de la sess
Dim requete As String

```

Exécuter pas à pas (principal Majuscule-F8) les instructions suivantes et répondre aux questions suivantes.

Quelle est la valeur du numéro de session qui est passé en paramètre à la fonction getInfosSession ?

Il suffit d'amener le pointeur de la souris sur le nom de cette variable pour lire la valeur contenue actuellement pas la variable. Cette valeur est 6 : c'est le numéro de session qui a été saisi dans le formulaire.

A quoi sert l'alternative « If estEntier ... » ?

A vérifier que le numéro de session saisi est une valeur entière.

Quelle est la requête SQL construite ?

"SELECT * from activite,action,session,lieu where activite.numero=action.numeroActivite and action.code = session.codeAction and lieu.id = session.idLieu and session.numero=6;"

On peut utiliser clic-droit / Ajouter un espion pour voir en permanence le valeur d'une variable. Ceci permet aussi de récupérer la totalité de la requête envoyée au SGBD.

```

'--- contrôle si le numero de session passé en paramètre est un entier ---
If estEntier(p_numeroSession) Then

'--- requête pour récupérer les informations de la session correspondan
Dim requete As String
requete = "SELECT * from activite,action,session,lieu"
requete = requete & " where activite.numero=action.numeroActivite"
requete = requete & " and action.code = session.codeAction"
requete = requete & " and lieu.id = session.idLieu"
requete = requete & " and session.numero=" & p_numeroSession & ";"
Set rsSession = CurrentDb.OpenRecordset(requete)
'--- retourne un booléen indiquant si la requete a donné ou non un résu
If rsSession.EOF Then

```

Expression	Valeur	Type	Contexte
66 requete	"SELECT * from activite,action,session,lieu where activite.numString	Routines.getInfosSession	

Quelle instruction permet de soumettre la requête SQL au SGBD ?

Set rsSession = CurrentDb.OpenRecordset(requete)

rsSession est un jeu d'enregistrement, il ne contient qu'une seule ligne : les données qui concernent la session de formation. Ce jeu d'enregistrement constitue un résultat retourné par la fonction.

CurrentDB désigne la base de données utilisée.

OpenRecordset soumet la requête au SGBD

A quoi sert l'alternative suivante ?

Il suffit de lire le commentaire associé : la fonction prend la valeur d'un booléen (variable qui prend la valeur « vrai » ou « faux ») : vrai quand on a trouvé une session portant ce numéro dans la base de données, et « faux » sinon.

Quelle valeur prend la fonction ?

Il suffit de continuer l'exécution des instructions (F8) : elle prend la valeur « vrai » car on a trouvé une session 6 dans la base de données et les données qui la concernent se trouvent dans le jeu d'enregistrements rsSession.

Après avoir exécuté la fonction getInfosSession() où se retrouve t-on ?

Après avoir appuyé sur F8 sur la ligne « End Function » on se retrouve dans la procédure btn_OK_Click, juste après l'appel de la fonction getInfosSession().

Comme getInfosSession() a pris la valeur « Vrai », les instructions qui suivent « then » sont exécutées.

Quelques questions de révision

Quelle requête SQL permet de récupérer dans la base de données les informations affichées dans le formulaire ?

Quelle instruction dans quelle fonction permet de vérifier que le numéro de session saisi par l'utilisateur existe ?

Quelle instruction dans quelle fonction permet d'afficher le libellé de l'activité dans le formulaire ? Comment interpréter cette instruction ?

Quelle instruction dans quelle fonction prévient l'utilisateur que le numéro de session n'existe pas ?

Saisir un numéro de session qui n'existe pas (par exemple 1000) et observer comment se comporte l'application.

Modifier l'application

Observons à la trace l'exécution de l'application lorsqu'on saisit par erreur un numéro composé de lettres.

A votre avis, dans quelles fonctions l'erreur peut-elle se produire ?

Peut-être dans la fonction estEntier() qui vérifie si la valeur saisie comme numéro de session est bien un entier.

Éventuellement dans la fonction getInfosSession() puisque la session n'existe pas.

Quelle est l'instruction qui provoque l'erreur d'exécution ?

Pour la connaître, il suffit d'utiliser le bouton Débogage dans la boîte de dialogue qui s'affiche quand l'erreur se produit.

Il s'agit de l'instruction « rs.Close » située à la fin de la procédure btn_OK_Click.

Quelle est le rôle de cette instruction ?

Elle ferme le jeu d'enregistrements « rs » utilisé pour stocker les données issues de la base de données concernant la session de formation choisie par l'utilisateur.

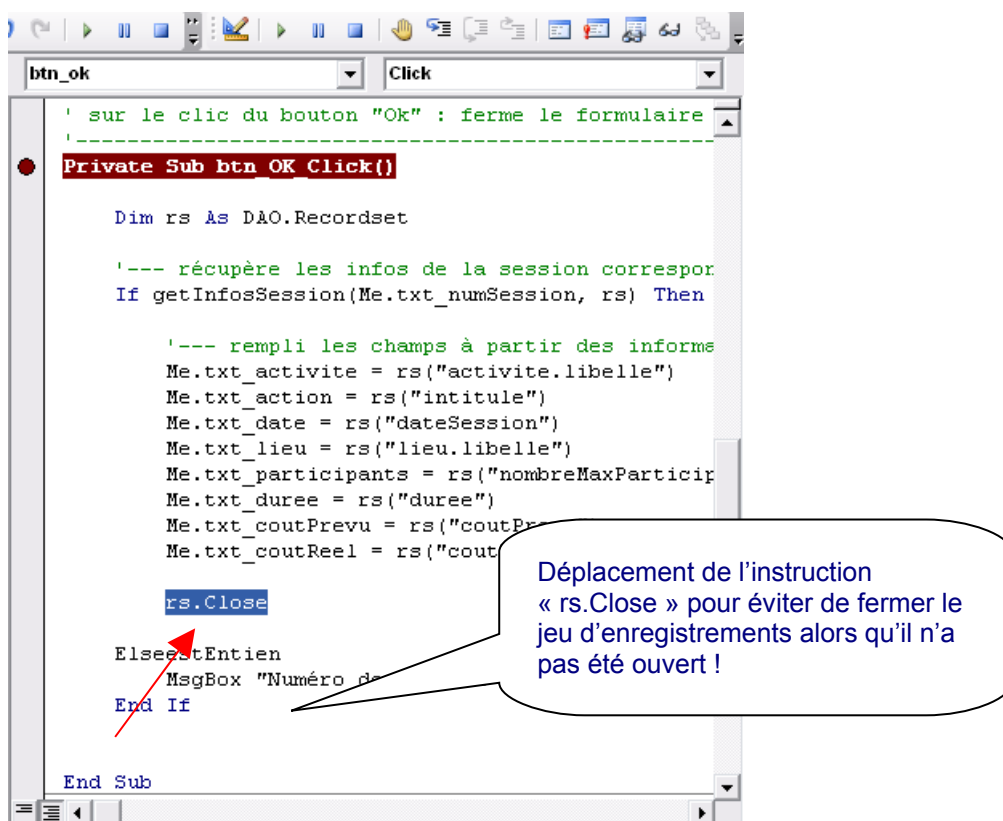
Pourquoi cette erreur se produit-elle ?

La fermeture du jeu d'enregistrement ne peut avoir lieu car il n'a jamais été ouvert. En effet quand le numéro saisi n'est pas un entier (fonction estEntier), l'instruction qui ouvre le jeu d'enregistrements n'est jamais exécutée. Il s'agit de l'instruction : CurrentDb.OpenRecordset(requete)

On peut vérifier cela en exécutant l'application en saisissant un mot comme numéro de session.

Comment modifier le programme pour faire en sorte que cette erreur ne se produise pas ?

Il ne faut exécuter l'instruction qui ferme le jeu d'enregistrements « rs » que lorsqu'on est sûr que celui-ci a été ouvert, autrement dit, lorsque qu'on a bien récupéré les informations de session de la base de données. Soit à la fin de la première partie de l'alternative.



Il faut remarquer que ce correctif n'est pas tout à fait satisfaisant puisqu'il existe maintenant des cas où le jeu d'enregistrement est ouvert et jamais fermé : quand il n'y a pas de session correspondant au numéro de session saisi. Ceci dit, dans ces cas là, le jeu d'enregistrement existe mais il est vide ! On peut aussi avantageusement se contenter de supprimer l'instruction car, par défaut, la fermeture est gérée automatiquement.

Lexique

Les définitions proposées ci-dessous ne sont pas exhaustives, elles conviennent pour l'exercice proposé dans cette publication.

Application	Unité de traitement au sein du système d'information d'une organisation prenant en charge des besoins métier.
Programmes	Ensemble des fichiers qui constituent les programmes d'une application
Code	Texte rédigé en respectant la syntaxe d'un langage de programmation.
Ligne de code	Ligne dans un programme. Une ligne de code peut contenir un commentaire et / ou une ou plusieurs instructions.
Fonctions d'une application	Rôles que doit prendre en charge une application : gérer le dialogue avec l'utilisateur, traiter les informations, accéder aux données.
Interface utilisateur	Ensemble des éléments qui permettent le dialogue entre un utilisateur et une application : fenêtres, contrôles graphiques, liens. Exemple : le formulaire de gestion des frais
Variable	Espace en mémoire vive dans lequel peuvent être enregistrées des valeurs pendant l'exécution d'un programme. Une variable a un nom, un type et une valeur. Le type d'une variable désigne le type de valeurs qu'elle permet de stocker : entier, réel, chaîne de caractères, etc. L'opération d'affectation consiste à affecter une valeur à une variable. Exemple : getInfosSession = True
Débogage	Activité qui consiste à identifier et à corriger des erreurs dans le code d'une application
Débogueur	Outil logiciel qui permet de suivre à la trace l'exécution des instructions d'un programme en positionnant des points d'arrêt, en exécutant les instructions pas à pas et en observant le contenu des variables.
Procédure	Unité de traitement dans un programme. Une procédure a un nom et admet un ensemble de paramètres. Ses instructions définissent le traitement réalisé. Exemple : MsgBox "Numéro de session incorrect !"
Fonction	Procédure pouvant être appelée dans une expression. Une fonction retourne une valeur utilisable dans une expression. Exemple : getInfosSession(...) As Boolean Exemple d'appel de fonction : If getInfosSession(Me.txt_numSession, rs) Then... Ou If estEntier(p_numeroSession) Then...
Paramètres d'une fonction ou d'une procédure	Dans la définition d'une procédure ou d'une fonction, un paramètre est une variable dont la valeur doit être fournie au moment de l'appel de la procédure ou de la fonction. getInfosSession(ByVal p_numeroSession As String, ByRef rsSession As DAO.Recordset) As Boolean ByVal : la valeur est donnée à la procédure ByRef : la valeur est donnée par la procédure

Événement	<p>Action de l'utilisateur (clic de souris ou frappe d'une touche) ou phénomène interne (erreur d'exécution) susceptible de déclencher l'exécution d'un traitement.</p> <p>Exemple : Procédure exécutée quand l'utilisateur clique sur le bouton Ok : Private Sub btn_OK_Click()</p>
Alternative	<p>Instruction qui permet d'exécuter des instructions en fonction d'une expression logique (qui prend la valeur Vrai ou Faux).</p> <p>Exemple :</p> <pre>If rsSession.EOF Then getInfosSession = False Else getInfosSession = True End If</pre>
Jeu d'enregistrements	<p>Résultat d'une requête stocké dans une variable.</p> <p>Exemple :</p> <p>Déclaration de la variable rs en tant que jeu d'enregistrements Dim rsSession As DAO.Recordset</p> <p>Exécution d'une requête et affectation du résultat dans la variable Set rsSession = CurrentDb.OpenRecordset(requete)</p> <p>Test pour vérifier si le jeu d'enregistrement est vide, s'il est vide cela signifie que la requête n'a retourné aucune ligne de résultat. If rsSession.EOF Then</p>

Comprendre ce qu'est un jeu d'enregistrements

On peut considérer qu'un jeu d'enregistrements est une variable qui contient le résultat retourné par une requête.

Prenons par exemple la requête suivante :

```
Requete = "SELECT Agent.code, Agent.nom, Agent.prenom
          FROM Agent.code = inscription.codeAgent
          WHERE inscription.numeroSession = 6"
```

On définit la variable jeu d'enregistrement:

Dim jeuEnregistrements as DAO.Recordset

On exécute la requête en affectant le résultat dans le jeu d'enregistrements :

Set jeuEnregistrements = CurrentDb.OpenRecordset(requete)

La variable jeuEnregistrements alors ceci :

	code	nom	prenom
▶	6560748Y	Gomez	Sandrine
	6851923T	Nguyen-Ngoc	Annie-Claire
	7140024W	Dageville	Sabine
	7430099V	Maxiola	Stéphane
	7803406N	Pasti	Stéphanie
	7950856K	Ginocchio	Christophe
	7967964F	Wetterwald	Dominique

La flèche montre la prochaine ligne qui peut être lue ainsi :
jeuEnregistrements.MoveNext

Après cette instruction, la ligne est accessible dans la variable jeu d'enregistrements.

	code	nom	prenom
▶	6560748Y	Gomez	Sandrine
	6851923T	Nguyen-Ngoc	Annie-Claire
	7140024W	Dageville	Sabine
	7430099V	Maxiola	Stéphane
	7803406N	Pasti	Stéphanie
	7950856K	Ginocchio	Christophe
	7967964F	Wetterwald	Dominique

Il est possible d'accéder à une valeur en écrivant par exemple :

```
codeAgent = jeuEnregistrements(« code »)
```

On exécute maintenant les deux instructions suivantes :

```
jeuEnregistrements.MoveNext
jeuEnregistrements.MoveNext
```

Quelle est la valeur de jeuEnregistrements(« nom ») ?

Réponse : « Dageville »

Maintenant, combien de fois faut-il encore exécuter l'instruction « jeuEnregistrements.MoveNext » pour qu'on ait l'égalité suivante :
jeuEnregistrements.EOF = True ?

Réponse : 5 fois, car on se situe alors après le dernier enregistrement du jeu d'enregistrement.