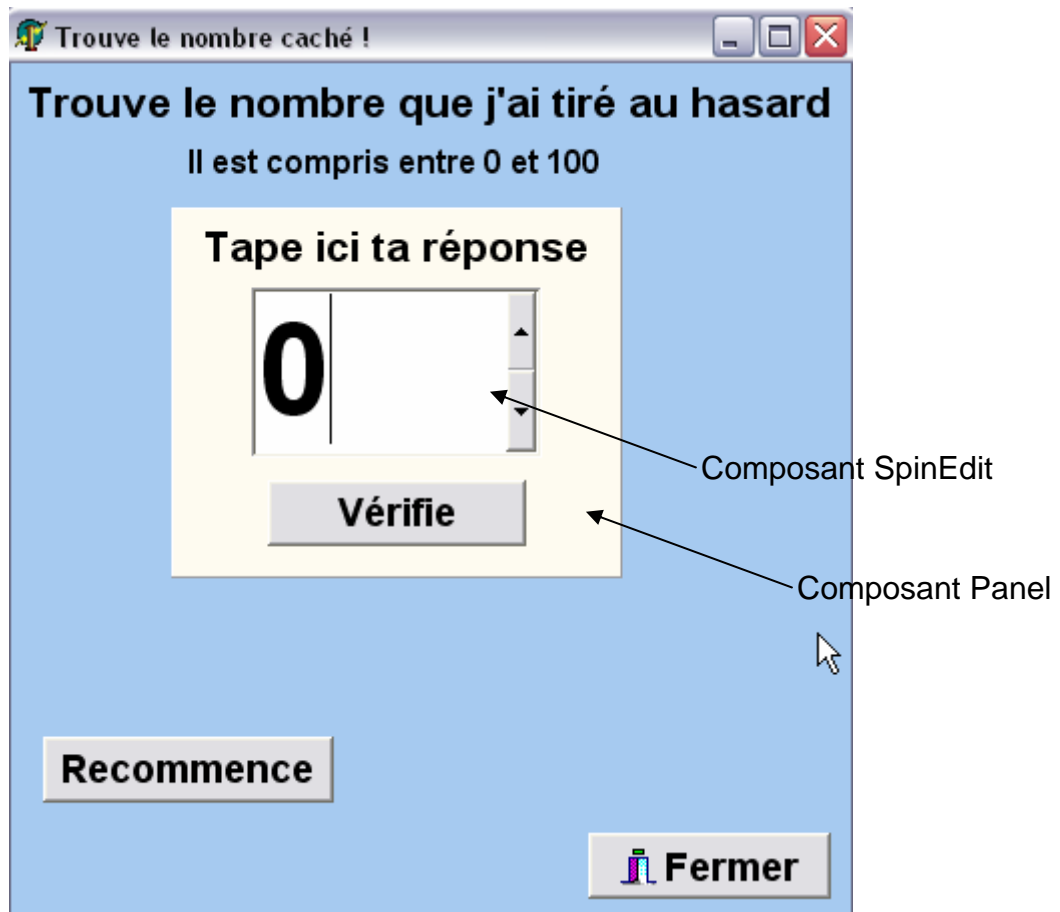


Delphi : Projet Nombre Secret

Objectif de ce projet, réaliser un programme qui tire un nombre au hasard et qui donne des infos afin de nous permettre de le retrouver.

Voici un exemple d'interface à réaliser



A l'aide des fiches précédentes, toutes présentes dans ton cahier, **réalise** tout ce que tu peux de cette interface, **dépose** et **personnalise** tous les composants. Les flèches te rendent attentif sur des composants que tu ne connais pas encore, mais que tu peux retrouver dans la barre des composants

Laisse libre court à ton imagination pour donner à l'interface l'aspect qui te convient (couleurs, police, etc...). N'oublie pas d'enregistrer rapidement et régulièrement ton projet.

Une fois que ton interface est prête, réfléchis et note sur une feuille de papier toutes les opérations que devra effectuer le programme lors du clic sur :

- 1) le bouton **Vérifie**
- 2) le bouton **Recommence**
- 3) le bouton **Fermer**

Essaye également de déterminer le nombre de Variables nécessaire.

Passons maintenant aux tâches à réaliser (réponses aux questions)

- 1) Quelles tâches doit effectuer l'interface lors du clic sur le bouton **Fermer** ?
Le programme doit :
 - a) Fermer l'interface
- 2) Quelles tâches doit effectuer le programme lors du clic sur le bouton **Vérifier** ?
Le programme doit :
 - a) Ajouter 1 au nombre des essais
 - b) Comparer la réponse donnée avec le chiffre tiré au hasard
 - c) Indiquer si la réponse donnée est plus grande, plus petite ou égale au chiffre tiré au hasard (Eventuellement changer la couleur du commentaire)
 - d) Si la réponse donnée est correcte, il doit en plus afficher le nombre d'essais
Afficher le bouton **Recommence** et cacher le bouton **Vérifie**
 - e) Sinon placer le curseur dans le composant SpinEdit (où l'on doit donner la réponse)
- 3) Quelles tâches doit effectuer le programme lors du clic sur le bouton **Recommence** ?
Le programme doit :
 - a) Tirer un nouveau numéro au hasard
 - b) Mettre à "0" le nombre des essais
 - c) Mettre le curseur dans le composant SpinEdit
 - d) Afficher le bouton **Vérifie** et cacher le bouton **Recommence**
 - e) Effacer le commentaire
- 4) Il faut aussi à l'ouverture de l'interface faire effectuer les mêmes opérations que lorsque l'on clique sur le bouton **Recommence**

Voici comment cela se traduit en programmation

Ne pas oublier de déclarer toutes les variables nécessaires, il en faut deux !
Celle contenant le nombre tiré au hasard (**Hasard** de type Integer) et celle contenant le nombre d'essai nécessaire (**Essai** de type Integer).

- 1) Le programme doit **Fermer** l'interface :
Cette fonction est documentée dans les pages du projet Inverse Texte. Le type de bouton utilisé est un peu spécial puisqu'il s'agit d'un **BitBtn** auquel on attribue à la propriété **Kind** la fonction **bkClose**.

- 2) Ces opérations se font à l'aide de tests logiques de type si ... alors ... sinon

```
inc(Increment);                                {Ajoute 1 à la variable Incrément}

if SpinEdit1.Value = Hasard then                {si la valeur donnée est égale au chiffre tiré au hasard}
begin
  Label4.Font.Color := clGreen;                {attribue la couleur verte au caractère du label 4}
  Label4.Caption := 'BRAVO !!! (tu as eu besoin de ' + IntToStr(Essais) + ' essais)';
  {Ici on assemble du texte et le contenu d'une variable numérique qui faut transformer en caractères}
  Button1.Visible := false;                    {Rend invisible le bouton 'Vérifie'}
  Button2.Visible := True;                     {Rend visible le bouton 'Recommence'}
end
else                                            {Sinon}
begin
  if SpinEdit1.Value < Hasard then              {si la valeur donnée est inférieure au chiffre tiré au hasard}
  Begin
    Label4.Font.Color := clRed;
    Label4.Caption := 'Le nombre donné est trop petit !';          {affiche la phrase dans le Label4}
    SpinEdit1.setfocus;                            {met le curseur à l'intérieur du composant SpinEdit}
  end
end
```

```

else
  Begin
    Label4.Font.Color := clRed;
    Label4.Caption := 'Le nombre donné est trop grand !';
    SpinEdit1.setFocus;      {met le curseur à l'intérieur du composant SpinEdit}
  end;
end;

```

3) et 4) il faut attribuer la même procédure au Click sur le bouton "Recommence" et à l'ouverture de l'interface, il faut donc l'écrire deux fois. La première de manière classique en double-cliquant sur le bouton "Recommence"

```

begin
  Randomize;                {initialise le générateur de nombre au hasard}
  Hasard := random(101);    {Attribue à la variable Hasard un nombre au hasard entre 0 et 101}
  Increment := 0;          {Initialise la variable Essai}
  Button1.Visible := true; {Rend visible le bouton 'Vérifie'}
  Button2.Visible := false; {Rend invisible le bouton 'Recommence'}
end;

```

Pour attribuer la procédure à l'ouverture de la fiche, il faut double cliquer à un endroit sans composant de l'interface et saisir la même procédure.

