**Création des étudiants**

Dans le code du clic du bouton :

sRépertoire est une chaîne

sRépertoire=fRepSélecteur(fRepEnCours(), "Sélectionner le répertoire des photos")

SI sRépertoire="" ALORS // la sélection n' pas été effectuée

ToastAffiche("Importation annulée",*toastLong*,*cvMilieu*,*chCentre*,*RougeFoncé*)

SINON

HCréation(Etudiant)

UnFichier, ResListeFichier sont des chaînes

i est un entier

// Liste les fichiers de type ".JPG" présents dans le répertoire désigné.

// Le parcours ne se fait pas dans les sous-répertoires et est interruptible.

ResListeFichier = fListeFichier(sRépertoire+"\\*.JPG", *frNonRécursif*+*frInterruptible*)

// Pour chaque fichier trouvé

POUR TOUTE CHAÎNE UnFichier DE ResListeFichier SEPAREE PAR *RC*

// Ajout des étudiants dont on trouve la photo

Sablier(*Vrai*)

Etudiant.NumEtu= fExtraitChemin(UnFichier,*fFichier*)

Etudiant.Nom="Etu "+i

Etudiant.Prénom="Prénom "+i

Etudiant.Groupe= Hasard(1,4)

Etudiant.Etat\_présence = (Hasard(0,1) = 1) // présent ou absent

SI PAS Etudiant.Etat\_présence ALORS

Etudiant.Excusé = (Hasard(0,1) = 1) // si absent alors excusé ou non

SINON

Etudiant.Excusé = *Faux* // si présent alors non excusé

FIN

HAttacheMémo(Etudiant,Photo,UnFichier,*hMémoImg*)

HAjoute(Etudiant)

i++

FIN

ToastAffiche(i+" étudiants ajoutés",*toastLong*,*cvMilieu*,*chCentre*,*VertFoncé*)

MoiMême..Etat=*Grisé* //grise le bouton

FIN

**Inscription des étudiants**(exercice 6)**:**

Dans une des fenêtres, créez le bouton « Inscriptions » et y coller le code suivant :

nbActivités\_disponibles est un entier = HNbEnr(Activité)

SI nbActivités\_disponibles < 2 ALORS

Info("Inscriptions impossibles, pas assez d'activités disponibles")

RETOUR

FIN

TA est un tableau de nbActivités\_disponibles chaînes // on met les activités dans un tableau pour pouvoir

i est un entier // tirer 2 indices au hasard

POUR TOUT Activité

i++

TA[i]=Activité.Nom\_Activité

FIN

nNBActivités\_Pour\_Cet\_Etudiant est un entier

nIndice\_Activité1, nIndice\_Activité2 sont des entiers

HCréation(Etudiant\_Activité) // On écrase les anciennes données

POUR TOUT Etudiant

nNBActivités\_Pour\_Cet\_Etudiant= Hasard(0,2) // 0 à 2 activités à attribuer par étudiant

nIndice\_Activité1= Hasard(1,nbActivités\_disponibles) // un indice du tableau des activités au hasard

nIndice\_Activité2=nIndice\_Activité1

TANTQUE nIndice\_Activité2=nIndice\_Activité1

nIndice\_Activité2= Hasard(1,nbActivités\_disponibles) // un indice différent du tableau des activités

FIN

Etudiant\_Activité.NumEtu=Etudiant.NumEtu // étudiant concerné par l'activité

SI nNBActivités\_Pour\_Cet\_Etudiant>=1 ALORS

Etudiant\_Activité.Nom\_Activité=TA[nIndice\_Activité1] // la première activité

Etudiant\_Activité.NB\_heures=Hasard(1,6)

HAjoute(Etudiant\_Activité) // ajout #1

FIN

SI nNBActivités\_Pour\_Cet\_Etudiant=2 ALORS

Etudiant\_Activité.Nom\_Activité=TA[nIndice\_Activité2] // la seconde activité

Etudiant\_Activité.NB\_heures=Hasard(1,6)

HAjoute(Etudiant\_Activité) // ajout #2

FIN

FIN

ToastAffiche("Activités ajoutées")