Prénom :....

TP SIN

Introduction langage C++

Support : ordinateur

Pré requis (l'élève doit savoir):

- Utiliser un ordinateur

Objectif terminale :

L'élève doit être capable de réaliser une page html5.

<u>Matériel</u>

- Ordinateur
- Logiciel C++ builder

Compétences :

Nom :.....

Prénom :....

1. Présentation



On peut trouver sur cette page plusieurs parties :

- • La classique barre de menu
- La barre d'outils qui se décompose en 2 grandes parties :
 - La palette d'outils permettant d'effectuer les opérations les plus courantes (sauvegarde, ouverture de fenêtres, etc.)
 - o Les palettes de composants disposées accessibles par des onglets
- L'inspecteur d'objets qui permet de manipuler les propriétés des composants et d'associer du code à leurs événements
- Les fenêtres d'interface créées par l'utilisateur. Ce sont les fenêtres de l'application en cours de création, elles portent ici le nom de fiche (form en anglais). Certaines peuvent être cachées, d'autres présentes à l'écran. On verra que dans la plupart des cas, leur position (ainsi que leurs autres caractéristiques géométriques) à l'exécution est le reflet exact de ce qu'elles étaient lors de la conception
- L'éditeur de code. A chaque fiche ouverte correspond deux fichiers source (un fichier .h et un fichier .cpp) qui sont éditables dans cette fenêtre avec mise en évidence de la syntaxe.

2. <u>Réalisation d'une application</u>

Les éléments automatiquement créés sont les suivants : une fiche nommée Form1 ainsi que les fichiers associés Unit1.cpp et Unit1.h. Les fonctionnalités ainsi créées sont toujours renommables après coup.

• Ouvrir un nouveau projet

6	RAI	O Studio XE5						-		-			
	Fich	ier Edition	Chercher	Voir	Refactori	ng	Projet	Exécuter	Composant	Outils	Fenêtre	Aide	Disposition par o
		Nouveau			۱.		Applic	ation Fich	es VCL - C++B	uilder			- 🧇
	\$	Ouvrir				-10	Applic	ation VCL	Metropolis UI	- C++Bu	Ilder		6
Ľ	2	Ouvrir un pro	ojet	C	trl+F11		Applic	ation de b	ureau FireMon	key - C+	+Builder		
		Ouvrir depuis	s le contrôle	de ver	sion		Applic	ation Fire	Monkey Metro	polis UI -	C++Build	er	
	•	Rouvrir			•	Ô	Packa	ge - C++B	uilder				
		Enregistrer			Ctrl+S		Fiche	VCL - C++	Builder				
i -	<u>)</u>	Enregistrer so	ous			8	Fiche	FireMonke	y - C++Builder	r			cumentation
	4 4	Enregistrer le	projet sous			-	Unité	- C++Build	der				
		Tout enregist	trer	Maj	+ Ctrl+S	93	Autre.						
	8 40	Fermer											-
	1 44	Tout fermer					Persor	nnaliser					1 =
	٦	Utiliser l'unit	é		Alt+F11		1	Savi	- ez qu'il est i	nécessa	aire de c	lispos	er d'un Mac
	•	Imprimer						Mac	conformén	nent au	ıx instru	ctions	du tutoriel
		Quitter											
	🧧 In	specteur d'obje	ets		P 23]		C	onfigurer	votre	e Mac		

• Enregistrer le projet et les différents fichiers dans un même dossier



Nom :.....

Prénom :....

• Sélectionner Form1 et changer son nom



• Changer le texte de présentation de la page



• Changer l'icône (voir dossier ressource)



• Changer l'icône du fichier exécutable sous Windows





Options de projet pour Project1.exe	(Win32 - Debug)
Compilateur C++ Avancées Avertissements Compatbilité Débogage En-têtes précompilés Optimisations Répertoires et Conditions Compilateur Delphi Compilateun Delphi Compilateun Delphi Compilateun Delphi Compilateur Delphi Compilateur C++ Avertissements Sortie Compilateur de ressources Compilateur de construction Application Application	Gible : Debug configuration - Windows 32 bits plate-forme · Appliquer Enregistrer Paramètres d'icône d'application Icône : Charger une icône 2 Par défaut 2 Thèmes d'exécution • Manifeste personnalisé • Paramètres de sortie • Extension du fichier cible : •
	OK Annuler Aide

• Changer la plateforme d'exécution si nécessaire : 32 bits -> 64 bits



• Compiler pour essayer



Le fichier exécutable se trouve sur la racine du dossier d'enregistrement, dans Win32, Debug pour une application Windows 32 Bits



- Exercice d'application :
 - Cahier des charges :
 - On veut changer le contenu d'un texte en appuyant sur un bouton (bonjour devient salut)
- Ajoute sur la page un Label depuis la palette d'outils

Project1.cb 🗄 Vue Modèle	Rexplorateur
🕂 Palette d'outils	# 8
🗈 👻 🔓 🔍 label	8
🗆 Standard	
📧 TLabel	
🗆 Supplément	
🔠 TLinkLabel	

Sélectionnez Tlabel avec la souris et faites le glisser sur la page

Page d'accueil	Unit1.cpp	Déploiement Project 1	
Formation st	i2d C++		
	<mark>.</mark>		
a di anti a di anti a di anti	el 🖸 🗄 👘 👘		
	P' 🔍		
	<mark>.</mark>		
	······································		
		· · · · · · · · · · · · · · · · · · ·	
		· · · · · · · · · · · · · · · · · · ·	

Nom :.....

Prénom :.....

• Faire la même chose avec un bouton



ł		1	I	F	br	m	at	tio	or	13	sti	2	d	C	+	÷																										ŀ	C	_			e)		S	3
•									1	1	1	1	÷	÷	÷	÷	1					1	1	1	1	÷	÷	÷	1	1	•		÷	÷	1	•		1	÷	÷	÷			1	-	-		-	-	1	1
•															1	•												•	1		•		•	1							•		• •				1	• •	1.1	1	•
•	•			•						•			•			•	•	•	• •					•	•			•		•	• •		•			•	• •			•	•		• •			•		• •			•
•											1				1														1		•			1	1				1			1					1	• •	1	1	
1													1	_	_	_	_	_	_	_		1.1							1		•											1						1.1	1	1	•
1	•									•					F	lu i	tte	on	1			•						•		•	• •		•			•	• •			•	•		• •			•		• •		1	•
1													l			~			-												•		•													•				1	•
Ċ.	1							- 1	1	-	1											- 1	1	1		1			÷.	÷.				÷.	÷.	1		- 1	1		÷.	1		- 1		1	1			0	÷.
[÷.											1	1							1					1	1						1				1	1
									1		1				÷		2						-	-		1			2					1	2	1		-	1		1					1					1
																																																		1	
													1	al	be	11																					 														
													-7																								 														
÷																																																	÷		
÷	•						•			•			·		÷	•	•	•	• •		•			•	•			•		•	• •	•	•		•	•	 • •			•	•	• •	• •			•	•	• •	• •		•
÷															÷	•												•			• •					•							• •				•		÷ •		•
•																•		•	•									•	۰.	•	•					•					•						•	• •	•		٠.
•	•						•			•			•			•	•	•	• •		•			•	•		•	•	•	•	• •	•	•		•	•	• •			•	•	• •	• •			٠.	•	• •	•		•
r.	•									•			•		1	•	•	•								1		•	1	•	• •		•		1	•	• •		1		•		• •					• •			•
1													•			•													1		•			1								1					1	• •	1.1	1	•
•	•									•			•			•	•	•	• •						•			•		•	• •		•			•	• •			•	•		• •			•		• •		1	•
1															1														1		• •								1											1	
[1										1	1	1	1	1	1	1	1				1	1	1	1	1	1	1	1	1			1	1	1				1	1	1	1.1		1			1				1
[1	Ĵ.											1	1							1					1	1						1				1	1
Ē.								- 1	- 1	- 1	- 1				0	2	2					- 1						2	2	2				0	2	1		- 1			2			- 1		1				0	1

• Changer la propriété Caption du bouton



• Que remarquez-vous au niveau du bouton ?

• Changer la propriété Caption du label

₩ 2 ↑ ↓		
page1	Formation sti2d C++	
Button1		
Label1		
	·····	
	• • • • • • • • • • • • • • • • • • • •	
	•••••••••••••••••••••••••••••••••••••••	
	• • • • • • • • • • • • • • • • • • • •	
	• • • • • • • • • • • • • • • • • • • •	
	•••	
📲 Inspecteur d'objets 🛛 🕂 😂		
Label1 TLabel	/	
Propriétés Evénements		
Alignment taLeftJustify		
AlignWithMargins False		
Anchors [akLeft,akTop]		
AutoSize ITrue		
≫ BiDiMode bdLeftToRiaht ▼		
Caption boniour		

- Que remarquez-vous au niveau du Label ?
- Maintenant, on va modifier la propriété de l'évènement Onclick du bouton

1	🦉 Inspecteur d'o	bjets 🕂 🕄
Bı	utton1 TButton	
F	Propriétés Evéneme	ents
	Images	A
Ð	LiveBindings	LiveBindings
»	OnClick	clicbouton 💌 😑
	OnContextPopup	
	OnDragDrop	
	OnDragOver	

- Que se passe-t-il au niveau de la page unit1.cpp ?
- Ecrire dans la page

void __fastcall Tpage1::clicbouton(TObject *Sender)

{

this->Label1->Caption="salut";

}



- Compiler l'application.
- Appuyer sur le bouton que remarquez-vous ?
- Réaliser une nouvelle application ci-dessous

🖥 Forma	tion sti2d C++		
		hut honigur	
	Sd	iut Donjour	
			· · · · · · · · · · · · · · · · · · ·
	boniour		

Nom :..... Pré

Prénom :....

• Cahier des charges

- Quand on appuie sur le bouton salut : le texte salut doit s'écrire
- Quand on appuie sur le bouton bonjour : le texte bonjour doit s'écrire

• Correction :



Nom :..... Prénom :.... 🔐 Page d'accueil 📓 Unit1.cpp 😰 Déploiement Project1 #pragma resource "*.dfm" Tpage1 *page1; //----fastcall Tpage1::Tpage1(TComponent* Owner) — : TForm(Owner) { L } 11---void __fastcall Tpage1::clicbouton(TObject *Sender) { this->Label1->Caption="salut"; L } //---void __fastcall Tpage1::Button2Click(TObject *Sender) { this->Label1->Caption="bonjour"; L} L

- Deuxième exercice d'application :
 - o On veut créer une application qui change les Euros en Dollars US

Cours de l'euro et du dollar US :

http://www.daily-bourse.fr/forex-EURO-DOLLAR-US-parite-EUR-USD.php

• Pour l'application suivante on va devoir utiliser deux variables pour la valeur en Euro et en dollar.

Il existe plusieurs types de variables, et chacun correspond à une taille maximale de nombre, ou un caractère, ou encore une vérité. Plus le type peut contenir de gros nombres, plus il utilise de place en mémoire.

Nom :.....

...... Prénom :.....

• bool

Utilise généralement 1 octet en mémoire, valeurs : true ou false.

• char

Utilise généralement 1 octet en mémoire, permet de stocker un caractère, valeurs : 256 caractères.

• unsigned short int

Utilise généralement 2 octets en mémoire, valeurs : de 0 à 65 535.

• short int

Utilise généralement 2 octets en mémoire, valeurs : de -32768 à 32 767.

• unsigned long int

Utilise généralement 4 octets en mémoire, valeurs : de 0 à 4 294 967 295.

long int

Utilise généralement 4 octets en mémoire, valeurs : de -2 147 483 648 à 2 147 483 647.

• int (16 bits)

Utilise généralement 2 octets en mémoire, valeurs : de -32 768 à 32 767.

int (32 bits)

Utilise généralement 4 octets en mémoire, valeurs : de -2 147 483 648 à 2 147 483 647.

• unsigned int (16 bits)

Utilise généralement 2 octets en mémoire, valeurs : de 0 à 65 535.

• unsigned int (32 bits) Utilise généralement 2 octets en mémoire, valeurs : de 0 à 4 294 967 295.

• double

Utilise généralement 8 octets en mémoire, valeurs : de 2.2e-308 à 3.4e-38.

• float

Utilise généralement 4 octets en mémoire, valeurs : de 1.2e-308 à 3.4e-38.



• Ecrire le programme suivant



- Tester le programme
- Mettre un mot à la place d'un chiffre dans l'Euros et valider. Que se passe-t-il ?
- A quoi correspond « StrToFloat » ?
- A quoi correspond « FloatToStr » ?
- Dans la définition des variables, que veut dire une variable « double » ?
- Le souci, c'est que l'utilisateur peut rentrer une lettre ou un chiffre. Donc maintenant nous allons contrôler le type de touche appuyé.

Pour cela nous allons utiliser l'évènement Onkeypress dans Edit1, qui va permet de récupérer la valeur de la touche appuyée dans la variable « Key ».

Remarque :

Key paramètre dans l'OnKeyPress gestionnaire d'événements est de type Char, par conséquent, la **OnKeyPress** événement enregistre le caractère ASCII de la touche pressée. Les touches qui ne correspondent pas à un ASCII Char valeur (MAJ ou F1, par exemple) ne génèrent pas un OnKeyPress événement. Les combinaisons de touches (comme SHIFT + A) génére un seul **OnKeyPress** événement (pour cet exemple, SHIFT + A entraîne une clé valeur de "A" si le verrouillage des majuscules est éteint). Pour répondre à des touches non-ASCII ou combinaisons de touches, utilisez le OnKeyDown ou OnKeyUp gestionnaire d'événements.

Les touches de 0 à 9, ont des valeurs comprises de 48 à 57.

Nous allons donc contrôler d'abord, si la valeur Key à une valeur supérieure à 48 et inférieure à 57. Dans le cas contraire on attribuera la valeur 0 et on affichera le message « touche incorrecte ».

• Ecrire le programme suivant.

-			÷.		void	fastcall	TForm1::Edit1Ke	yPress (TObject	*Sender,	System::WideChar	&Key)
	🖷 Inspecteur d'obj	ets	# 23		{						
E	dit1 TEdit		-		if	(Key < 48	Key > 57)				
	Propriétés Evéneme	ents				{					
	OnGesture			11.		Show	Message ("touche	incorrecte");			
	OnKeyDown			32							
≫	OnKeyPress	Edit1KeyPress				Key = 0;					
1	OnKeyUp		_	-		}					
	OnMouseActivate		-	_							
	OnMouseDown		=		3						
	OnMouseEnter				11						
	OnMouseLeave		-		//						
		1									
Lie	er visuellement										

Rappel :

• Les opérateurs de conditions

Comme leur nom l'indique, ces opérateurs vont permettre de comparer diverses valeurs entre elles. En tout, ils sont au nombre de huit, les voici :

Opérateur	Signification
==	égal à
<u>=</u>	différent de
===	contenu et type égal à
!==	contenu ou type différent de
>	supérieur à
>=	supérieur ou égal à
<	inférieur à
<=	inférieur ou égal à

• Les opérateurs logiques

Ils fonctionnent sur le même principe qu'une table de vérité en électronique. Ils sont nombre de trois :

Opérateur	Type de logique	Utilisation
&&	ET	valeur1 && valeur2
	OU	valeur1 valeur2
!	NON	!valeur

- Que se passe-t-il lorsqu'on tape un chiffre ou une lettre ?
- Exercice d'application
 - On veut déplacer une boule dans un carré par l'intermédiaire des flèches du clavier et afficher la valeur de la touche.
 - Contraintes :
 - La page doit faire 602x823
 - Le carré 400x400 doit être centré sur la page
 - La boule 100x100
 - L'image des touches 150x100 doit être centrée sur la page
 - Utiliser un Tpanel pour positionner les éléments







Nom :	Prénom :	
Exercice3		×
Valeur de la touche appuyée	Arelier	
	Appuyer sur une des flèches du clavier pour déplacer la boule	

• Pour les images utiliser l'objet Timage



- Si l'utilisateur n'appuie pas sur une des flèches, afficher le message « touche incorrecte »
- Pour contrôler la valeur de la touche, on utilisera l'évènement OnKeydown

Utilisez l'OnKeyDown gestionnaire d'événements pour spécifier un traitement spécial à se produire quand une touche est enfoncée. Le **OnKeyDown** gestionnaire peut répondre aux touches du clavier, y compris les touches de fonction et touches combinées avec le MAJ, ALT et CTRL boutons de la souris enfoncé les touches, et. **Remarque:** TCheckBox ne se déclenche pas **OnKeyDown** événements pour les touches fléchées.

• Pour tester la touche on utilisera la fonction SWITCH

<u>Syntaxe</u>

```
switch ( <variable switch> ) {
case <expression constante> : <instruction>;
break:
  .
default: <instruction>;
}
Exemple :
using namespace std;
int main(int argc, char * argv[])
{
 char ch;
 cout << "APPUYEZ sur a, b OU c. TOUT AUTRE CHOIX METTRA FIN A CE PROGRAMME." << endl;
 for ( /* TOUJOURS */; cin >> ch; )
  switch (ch)
  {
   case 'a' : /* LE CHOIX DE a A SA PROPRE ACTION. */
    cout << endl << "Option a sélectionnée." << endl;
    break;
   case 'b': /* b ET c DONNENT LES MEMES RESULTATS. */
   case 'c':
    cout << endl << "Option b ou c sélectionnée." << endl;
    break;
   default:
    cout << endl << "CHOIX INCORRECT ! Au revoir ..." << endl;
    return(-1);
  }
}
```

• Pour contrôler la position de la boule, utiliser la propriété position X et Y

```
(this->boule->Position->X);
t(this->boule->Position->Y);
```

Prénom :....

• Créer l'application

😅 RAD Studio 10 Seattle

