

Universidad Nacional “San Luis Gonzaga” de Ica

Facultad de Ingeniería Mecánica y Eléctrica

Escuela de Ingeniería Electrónica

**TUTORIAL, USO DEL COMANDO RATIO BUTTOM**

Crear una ventana basada en diálogo utilizando el MFC AppWizard en la que se pida el ingreso de 2 números y se pueda seleccionar una de las tres operaciones (COBRE, PLATA, ORO) utilizando radio butons. Deberá almacenar la aplicación con el nombre de **TAYPEHUERTAS.**

<<File/New/MFC AppWizard(exe)/Proyect Name=castillonjorge/

Location = BUTOON CASTILLON/Ok>>



***Seguidamente…***

<<Step1=Dialog Based/Next>>

<<Step2=3D controls/ActiveX controls/Next>>

<<Step3=MFC Standard/Yes Pleace/As a shared DLL/Next>>

<<Step4=Finish/Ok>>

Seleccione los controles creados por el asistente y elimínelos.

Ubique los controles indicados:

* 4 etiquetas.
* 3 cajas de texto.
* 3botones de opción.
* 1 caja de grupo.
* 1 Botón de comando.

|  |  |  |
| --- | --- | --- |
| **Control** | **Propiedades** | **Valor** |
| IDD\_TAYPEHUERTAS\_DIALOGIDC\_STATICIDC\_STATIC (2)IDC\_STATIC (3)IDC\_STATIC (4)IDC\_STATIC (frame)IDC\_EDIT1IDC\_EDIT2IDC\_EDIT3IDC\_RADIO1IDC\_RADIO2IDC\_RADIO3IDC\_BUTTON1 | CaptionCaption CaptionCaptionCaptionCaptionIDNumberIDNumberIDRead-onlyNumberIDGroupCaptionIDCaptionIDCaptionIDCaptionDefault button | Conductancia de un ConductorResistencia de un ConductorLongitud en metrosSeccion del conductorResultado“”IDC\_CONDUCTORTrueIDC\_SECCIONTrueIDC\_RESULTADOTrueTrueIDC\_ALUMINIOTrue&CobreIDC\_LATON&PlataIDC\_PLOMO&OroIDC\_CALCULAR&CalcularTrue |

**Nota**: Las propiedades Number, defaul button y Read-only de las cajas de texto se encuentran en la ficha Styles de la ventana de propiedades.

Modifique la ventana en tiempo de diseño, la apariencia de la ventana deberá ser similar a esta:



Ahora vamos a proceder a crear variables miembros para las cajas de texto de la siguiente manera, invoque al Class Wizard y en la ficha **Member Variables**:

<<Ctrl.+W / IDC\_CONDUCTOR / Add Variable>>

|  |  |  |
| --- | --- | --- |
| **Control IDs** | **Tipo** | **Nombre Variable** |
| IDC\_CONDUCTORIDC\_SECCIONIDC\_RESULTADOIDC\_COBRE | doubledoubledoubleint | m\_Conductorm\_Seccionm\_Resultadom\_Operacion |

Ahora si procedemos a escribir el código para que nuestra aplicación funcione. Todo código irá en el botón Calcular. Cree la función miembro para el botón llamada OnCalcular() dando doble clic al botón de comando en el diálogo,



Ahora digite la siguiente codificación

//Actualizamos los datos de las variables

 UpdateData (true);

 //Elegimos la operación de acuerdo a lo que el usuario

 //haya seleccionado desde los radio buttons

 switch (m\_Operacion)

 {

 case 0: m\_Resultado = 0.028 \*(m\_Longitud/m\_Seccion); break;

 case 1: m\_Resultado = 0.07 \*(m\_Longitud/m\_Seccion); break;

 case 2: m\_Resultado = 0.46 \*(m\_Longitud/m\_Seccion); break;

 }

 UpdateData(false);

}

Ahora para hacer que la operación predeterminada sea cobre, escribiremos unas líneas de código en la función **OnInitDialog** de la clase **TAYPEHUERTASDlg**:

Después de ubicarse en la función haga doble clic y modifique el contenido predeterminado a lo siguiente:

CDialog::OnInitDialog();

 //Set the icon for this dialog. The framework does this

 //automatically

 // when the application´s main window is not a dialog

 SetIcon(m\_hIcon, TRUE); // Set big icon

 SetIcon(m\_hIcon, FALSE); // Set small icon

 // Enviamos el foco a la primera operación (Suma)

 m\_Operacion = 0;

 UpdateData(false);

 return false;

}

// If you add a minimize button to your dialog, you will need the code below

// to draw the icon. For MFC applications using the document/view model,

// this is automatically done for you by the framework.

Presione las Teclas ctrl. + F5 para ejecutar la aplicación y verá una ventana similar a esta: