

INGENIERÍA MECATRÓNICA



DI_CERO

DIEGO CERVANTES RODRÍGUEZ

INSTRUMENTACIÓN VIRTUAL

NI LABVIEW 2020 (32-BIT)

Crear Tablas en un Archivo de
Lectura de Datos en LabVIEW

Contenido

Introducción Teórica de LabVIEW:.....	2
Introducción al Entorno de LabVIEW:.....	2
Front Panel: Ventana Gris con la Interfaz del Programa	4
Block Diagram: Ventana Blanca con la Lógica del Programa (Bloques)	4
Front Panel o Block Diagram - Show Context Help: Descripción de Bloques	5
Front Panel y Block Diagram: Navegar de una Ventana a Otra	6
Block Diagram - Cambiar Nombre a los Bloques: Nombre de los elementos en el Front Panel	7
Block Diagram - Highlight Execution: Correr Más Lento el Programa.....	8
Coertion dot: Conversión Automática de Datos por Parte de LabVIEW	8
Block Diagram - Clean Up Diagram: Organizar Automáticamente los Bloques del VI	8
Programa: Leer Datos y Darle Formato a su Archivo	9
Desarrollo del Programa: Dar Formato a un Archivo de Lectura de Datos	9
Block Diagram - Bucle For: Iteraciones Finitas para la Creación de Tablas.....	9
Block Diagram - Format Into String: Conversión de Cualquier Tipo de Dato a un String.....	11
Block Diagram - Random Number: Creación de Número Aleatorio.....	13
Block Diagram - Table: Creación de una Tabla con Strings	16
Ejecución del Programa: Crear una Tabla de Datos en LabVIEW	17
Block Diagram - Property Node: Propiedad Específica de Cualquier Bloque.....	18
Block Diagram - True Constant: Constante Booleana con Valor Verdadero.....	20
Block Diagram - Array Constant: Constante Tipo Array Vacía de 1 Posición	21
Block Diagram - String Constant: Constante Tipo String.....	22
Ejecución del Programa: Introducir Datos a una Tabla en LabVIEW	24



Introducción Teórica de LabVIEW:

LabView sirve para poder usar la computadora como instrumento de medición, monitoreo, control y análisis de procesos y operaciones, esto se hace a través de una frecuencia de muestreo que se relaciona con mediciones de los dispositivos digitales y tiene que ver con la señal de reloj de la tarjeta de desarrollo, indicando cada cuánto tiempo se hará un muestreo de cualquier señal del mundo real.

La diferencia entre los instrumentos virtuales de medición y los reales es más que nada el precio, ya que un osciloscopio cuesta alrededor de \$10,000 y se puede hacer la misma función con LabView y un Arduino, que cuesta alrededor de \$170, además de que es modular, esto implica que se pueden agregar o quitar funcionalidades. La mejor tarjeta de desarrollo para hacer esto es la de NI Instruments, que es la creadora de LabVIEW.

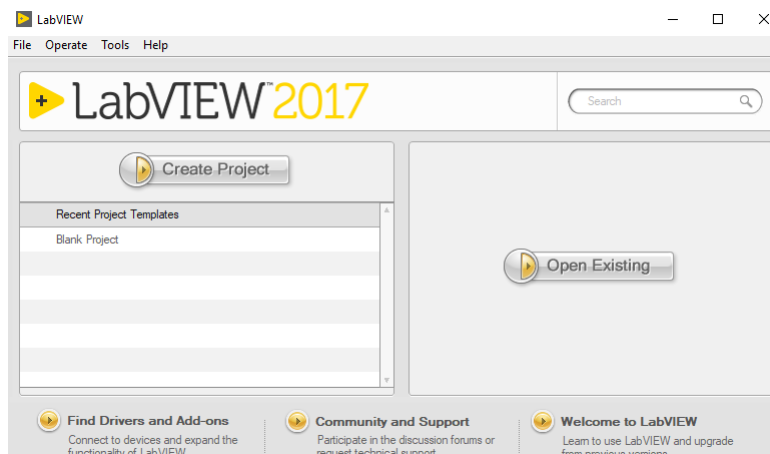
- **Instrumentación Tradicional:** El hardware es más usado, como por ejemplo con los circuitos integrados de un osciloscopio.
- **Instrumentación Virtual:** El software es el más utilizado y sus funciones son modulares, como lo es en una tarjeta de desarrollo de National Instruments.

La instrumentación virtual es empleada para la gestión de sistemas industriales y muy utilizado en compañías como: Ford, SpaceX, Accenture, Bosch, etc.

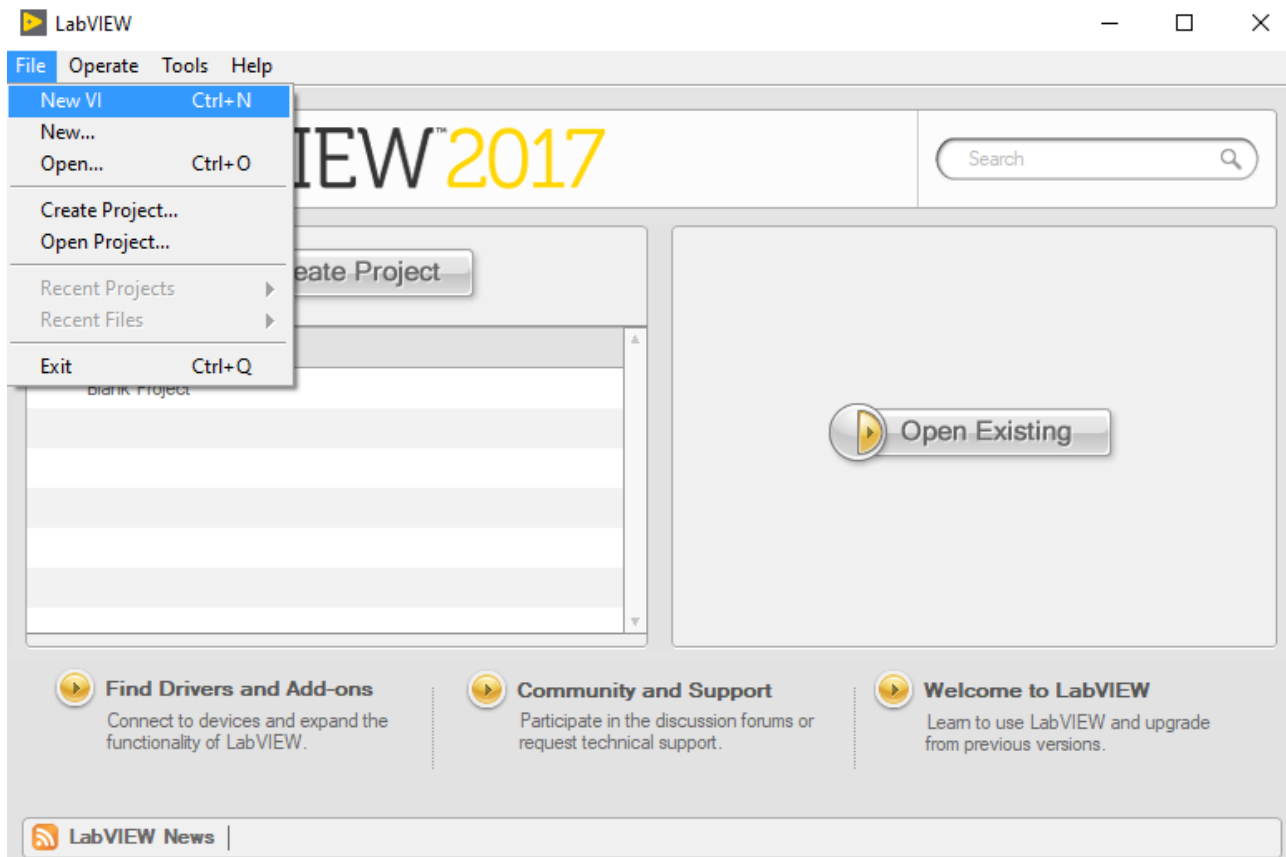


Introducción al Entorno de LabVIEW:

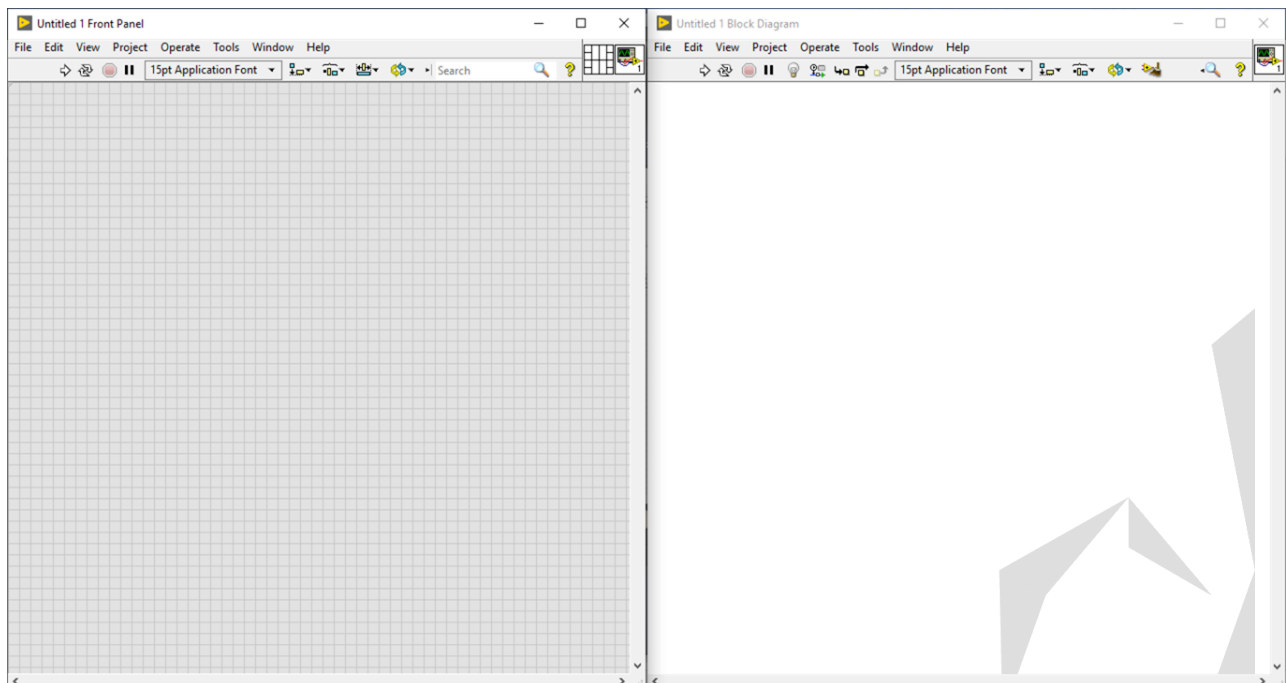
Un nuevo proyecto de LabView se abre por medio del botón de Create project que aparece inmediatamente cuando abra el programa.



VI se refiere a Virtual Instrument.

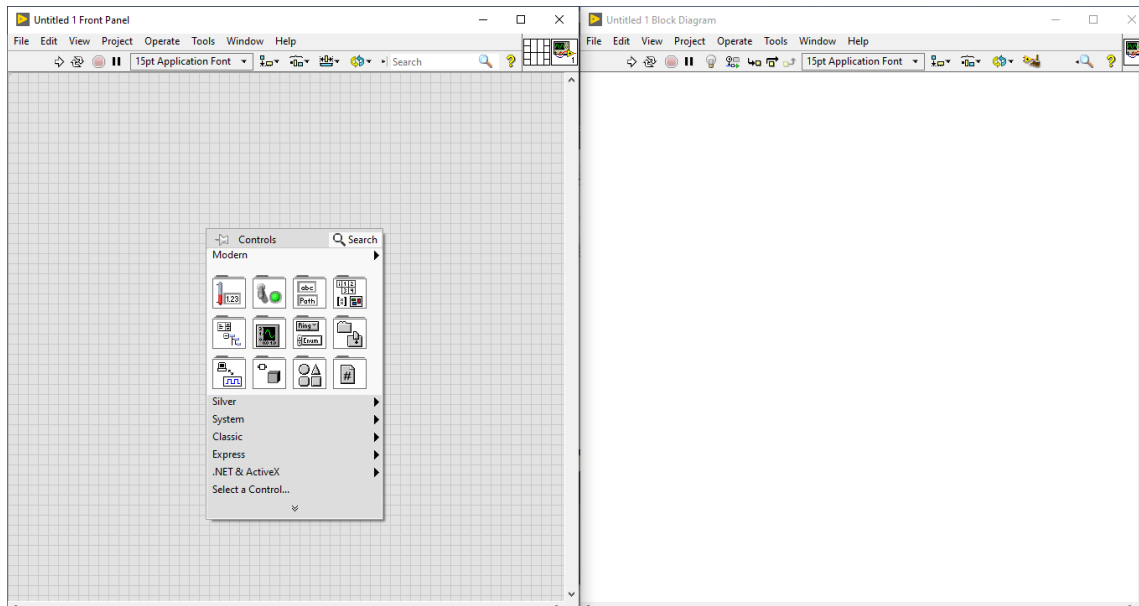


Al hacerlo me abrirá estas dos ventanas, en una de ellas se creará el programa con bloques (Ventana Block Diagram) y en la otra se verá la interfaz (Ventana Front Panel).



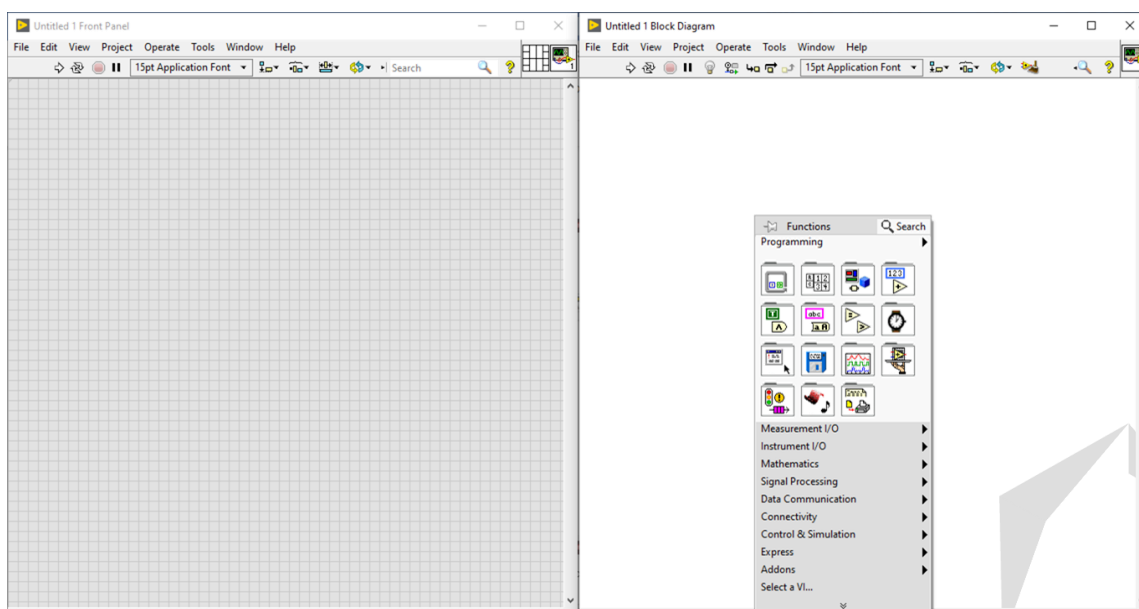
Front Panel: Ventana Gris con la Interfaz del Programa

En la ventana gris llamada **Front Panel**, es donde se observa la interfaz del Programa y se cuenta con el **control palette** que sirve para poder añadir elementos gráficos a la interfaz y aparece dando clic derecho en la pantalla gris. Si no aparece la otra ventana (blanca) por default, se debe seleccionar la opción **Window → Show Block Diagram** y con ello aparecerá.



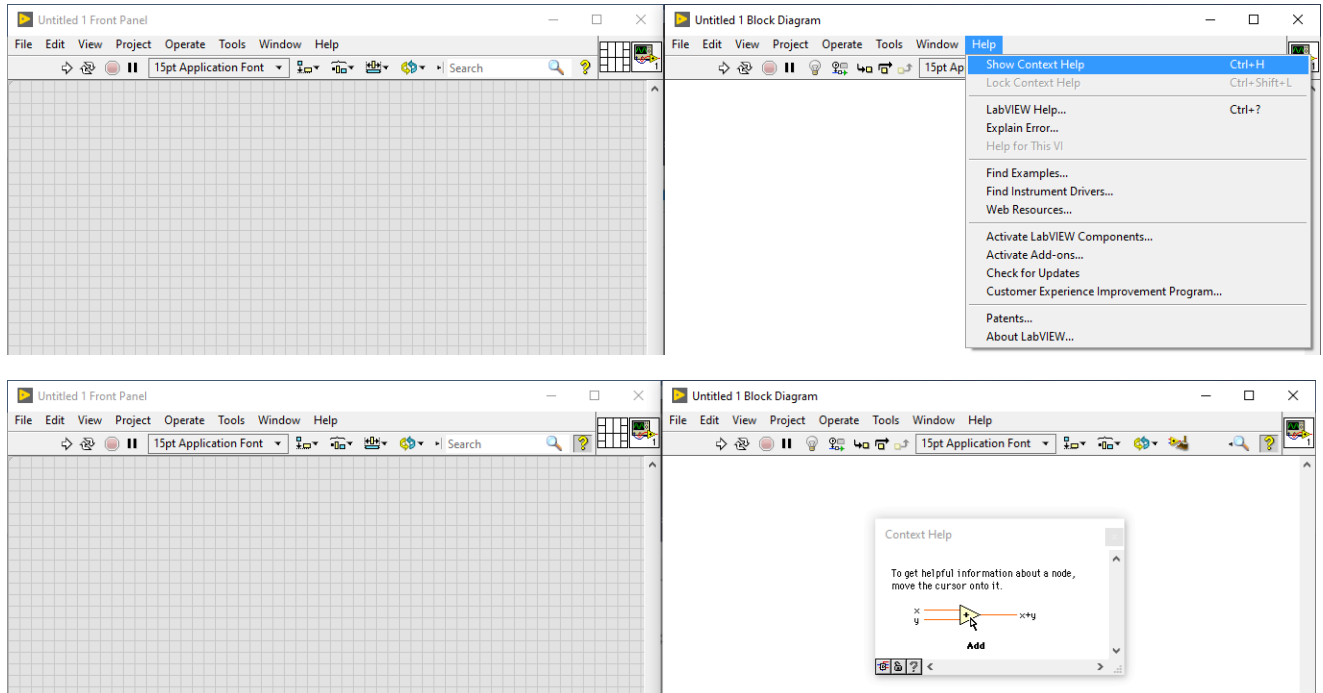
Block Diagram: Ventana Blanca con la Lógica del Programa (Bloques)

En la ventana blanca llamada **Block Diagram** aparece la **paleta de funciones** que sirve para introducir los elementos de programación en forma de bloques que se conectarán entre ellos y describirán la función del programa, aparece dando clic derecho en la pantalla gris. Si no aparece la ventana gris se debe seleccionar la opción **Windows → Show Front Panel** y con ello aparecerá.



Front Panel o Block Diagram - Show Context Help: Descripción de Bloques

Seleccionando la opción de Help → Show Context Help, aparecerá una ventana emergente que explicará las propiedades de los bloques que se puede seleccionar, mostrando una descripción de su función, imágenes explicativas y significado de sus pines de entrada y salida.



Las funciones o subrutinas son los elementos más básicos que pueden existir en LabView, dentro de ellas existe un código de bloque propio que describe sus funciones, pero además se cuenta con otros elementos:

VIs Express, VIs y Funciones

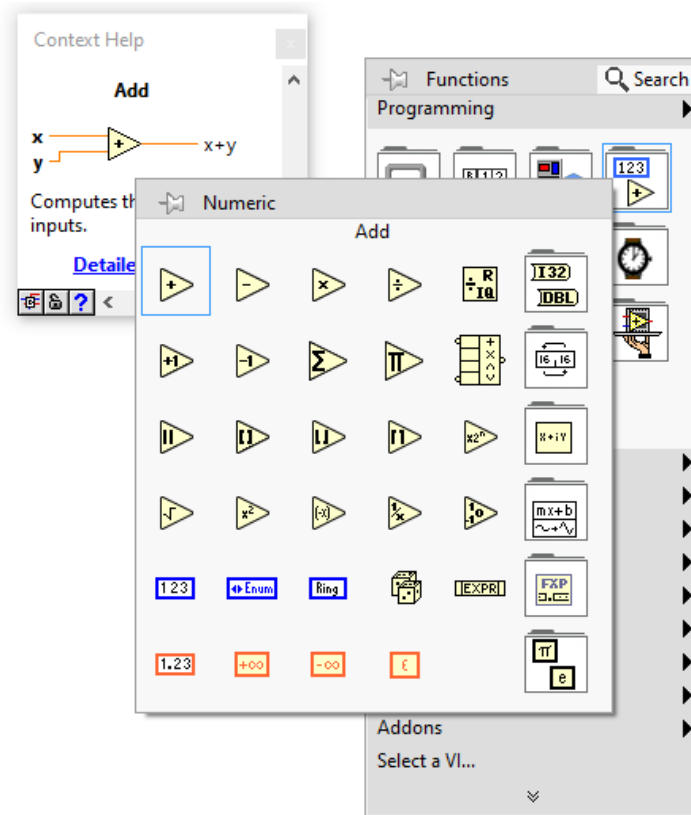


- **VIs Expreso:** VIs interactivos con pagina de dialogo configurable
- **VIs estándar:** VIs modulares y personalizables mediante cableado
- **Funciones:** Elementos fundamentales de operación de LabVIEW; no contiene panel frontal o diagrama de bloque



Función

En un bloque de código, las **terminales que aparezcan en negritas** son las que a fuerza deben estar **conectadas a algo**, las que no estén en negritas no deben estar conectadas a nada forzosamente.

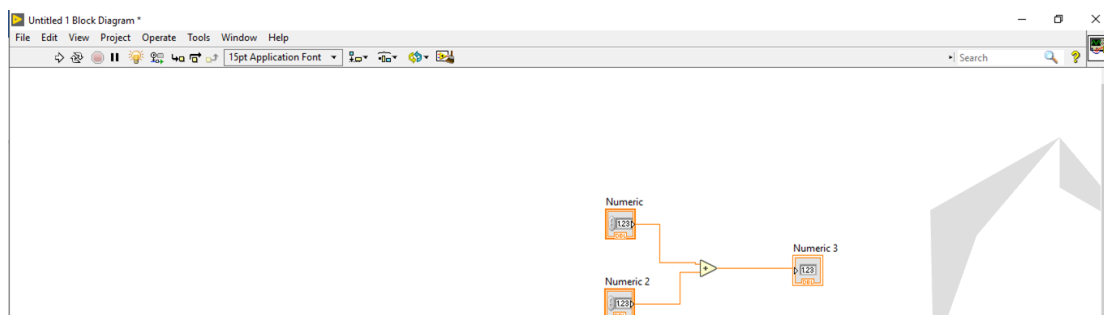


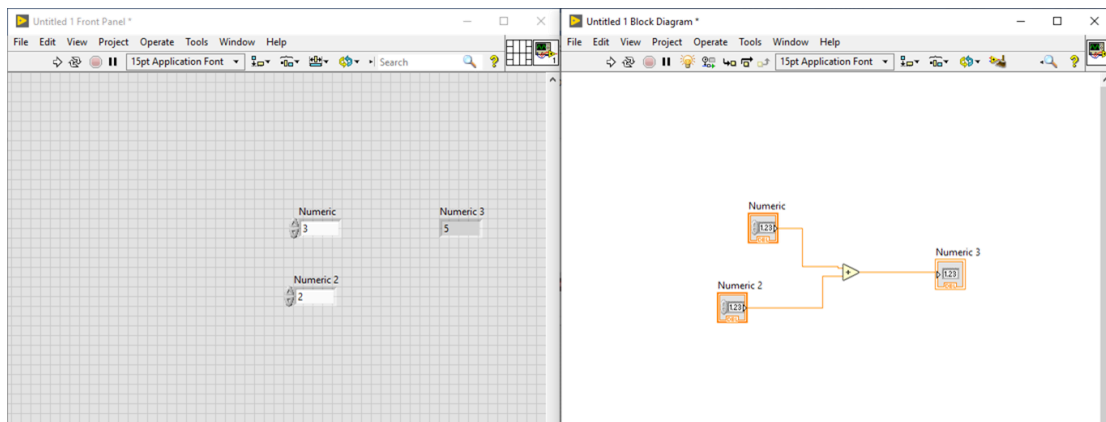
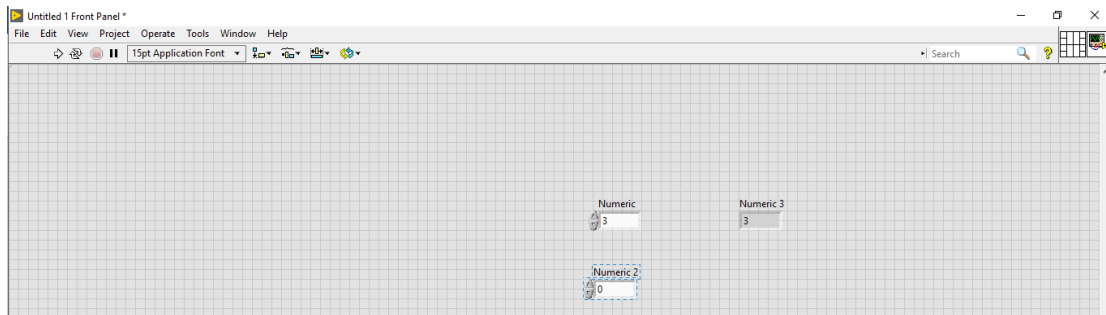
El programa es autocompilable, es decir que se corre por sí solo, por lo que si la flechita aparece rota es porque hay un error en el programa.



Front Panel y Block Diagram: Navegar de una Ventana a Otra

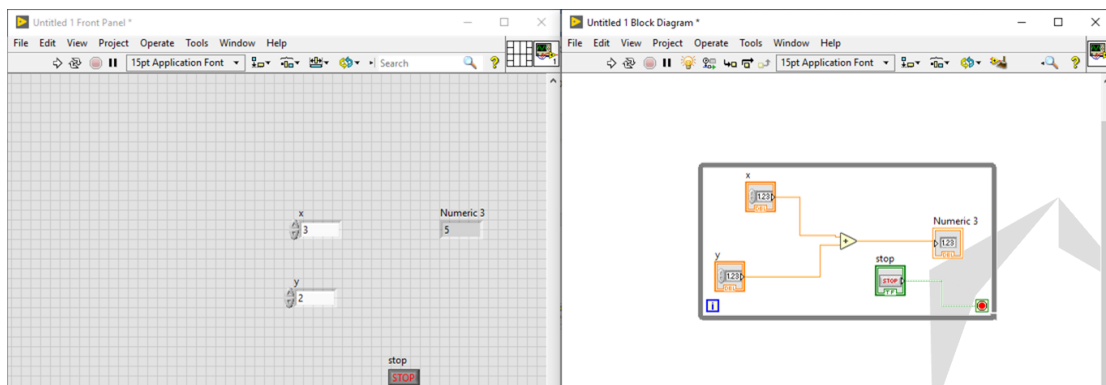
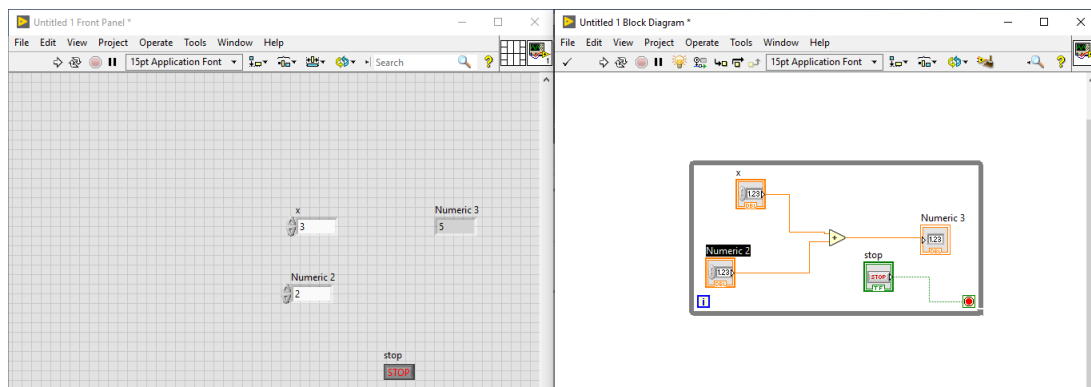
Al dar doble clic en el bloque de la pantalla blanca, me llevará al punto donde se encuentra el mismo bloque, pero en la pantalla gris.

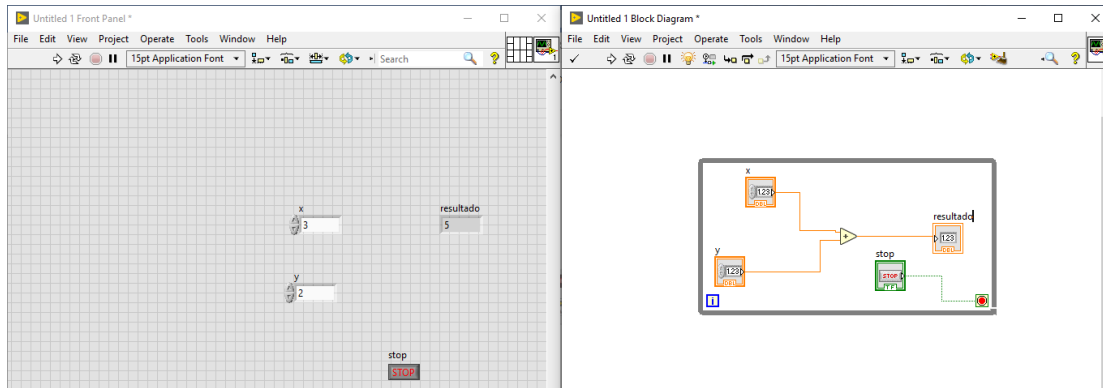




Block Diagram - Cambiar Nombre a los Bloques: Nombre de los elementos en el Front Panel

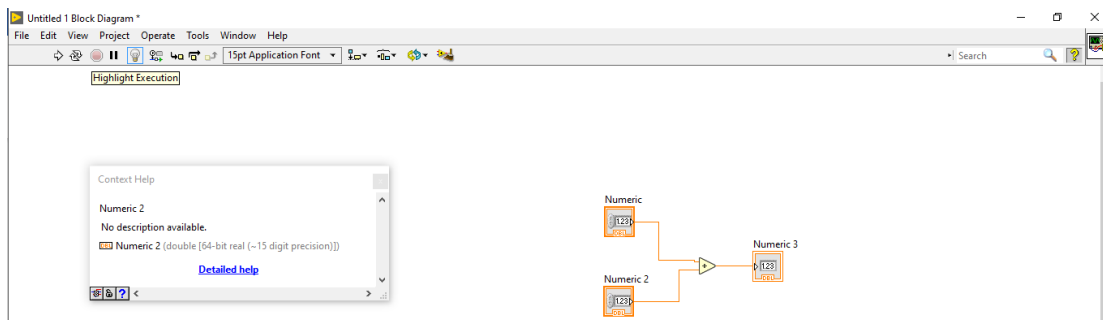
El nombre de los elementos de las interfaces se puede cambiar desde el Block Diagram, cambiándole literal el nombre a los bloques.





Block Diagram - Highlight Execution: Correr Más Lento el Programa

Podemos presionar el foquito del menú superior para ver el funcionamiento de programa de manera más lenta.

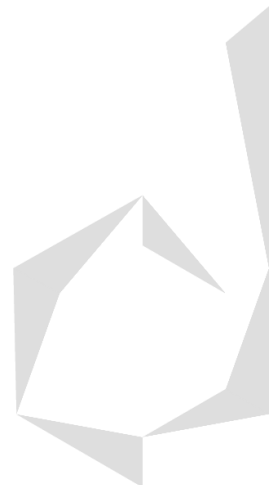


Coertion dot: Conversión Automática de Datos por Parte de LabVIEW

Aparece un punto rojo en la terminal del bloque llamado coercion dot, este lo que me dice es que los tipos de datos en la conexión son distintos, por lo que LabVIEW está forzando una conversión de un tipo de dato a otro, el problema es que en este tipo de conversión yo no sé si se están perdiendo datos, por eso debemos evitar el uso de coercion dots porque usa direcciones de memoria o recursos de la computadora sin que yo tenga control de ellos.

Block Diagram - Clean Up Diagram: Organizar Automáticamente los Bloques del VI

Con el botón de Clean Up Diagram que se encuentra en la parte superior derecha del Block Diagram se organizan mejor y de forma automática mis elementos.



Programa: Leer Datos y Darle Formato a su Archivo

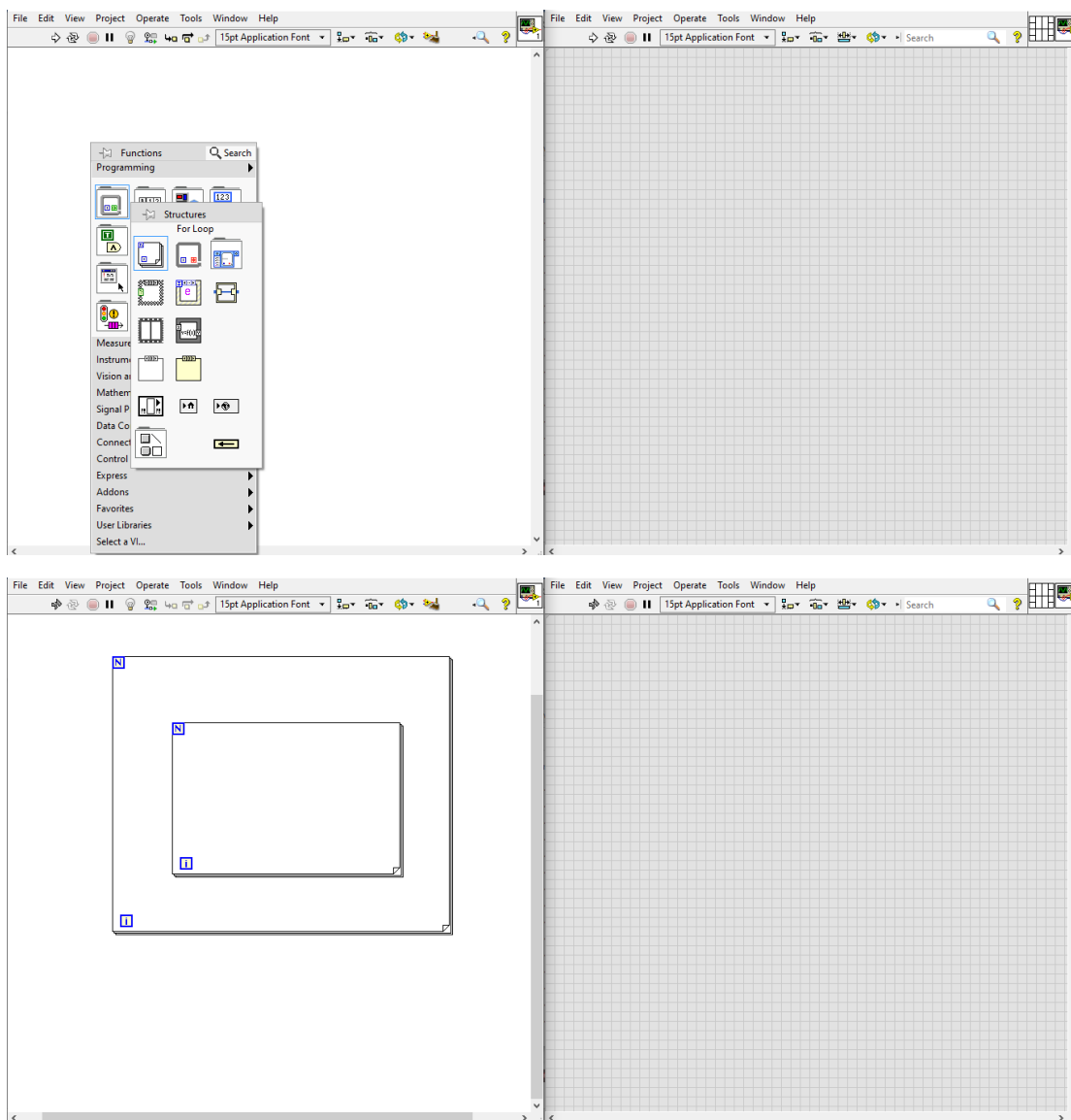
Dar formato a un archivo de lectura de datos en LabVIEW, estos datos pueden provenir de una señal o cualquier otra fuente.

Desarrollo del Programa: Dar Formato a un Archivo de Lectura de Datos

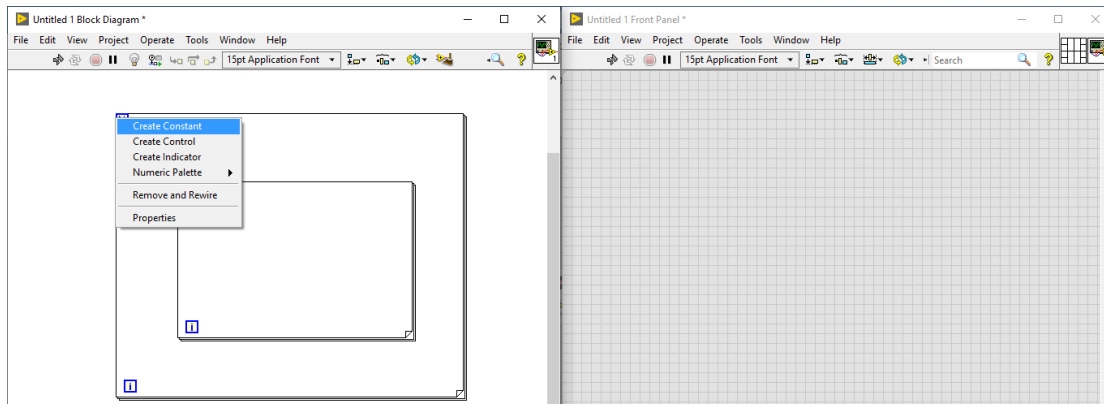
Block Diagram - Bucle For: Iteraciones Finitas para la Creación de Tablas

El ciclo for hace que el programa se ejecute un número finito de veces, indicado por las variables N e i, en el Bucle la variable N nos dice cuántas iteraciones se van a hacer y la i es una variable que indica el paso del conteo, en otras palabras, nos dice de cuanto en cuanto vamos contando hasta llegar a N.

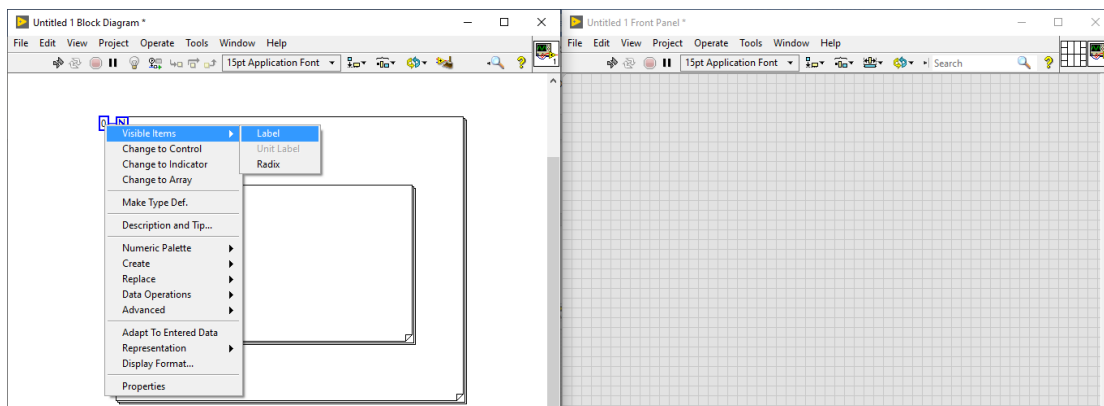
Se encapsulan dos bucles for para la creación de tablas, en uno se manejan las filas y en el otro las columnas.



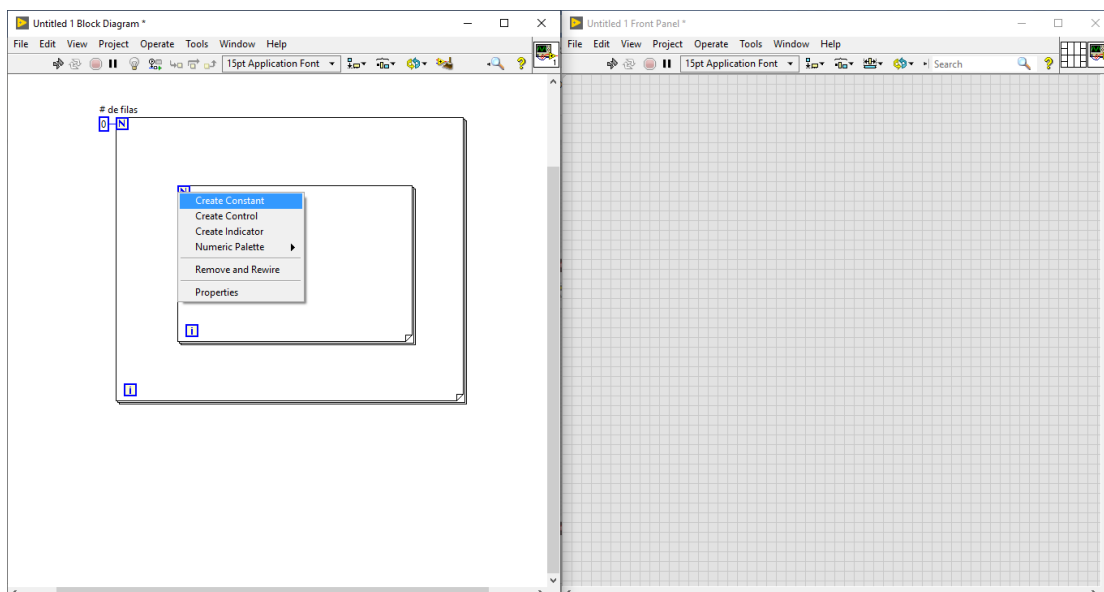
Crear una Constante para un Bloque: Clic derecho en la terminal del bloque de interés → Create → Constant. Esta constante debe ser igual a $N = \text{Número de Filas}$.

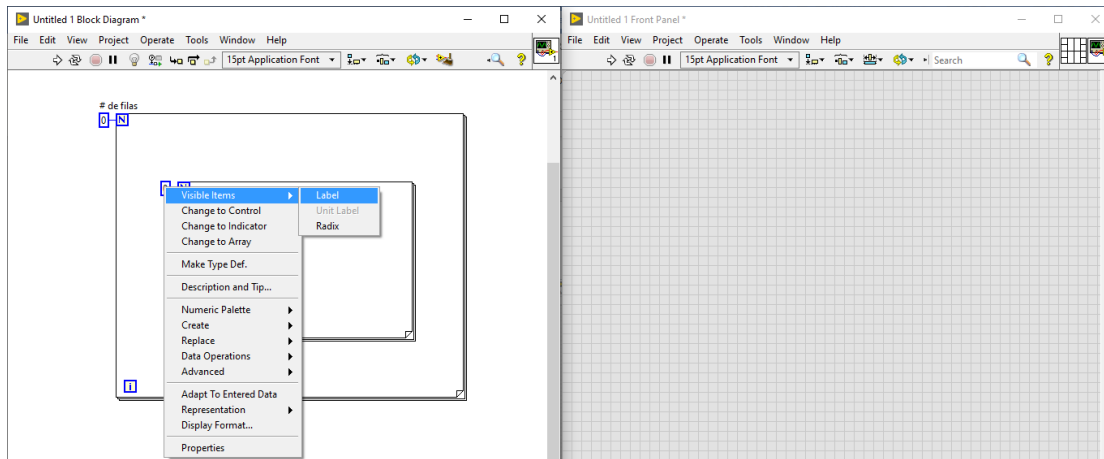


Mostrar nombre del bloque: Clic derecho → Visible Items → Label.



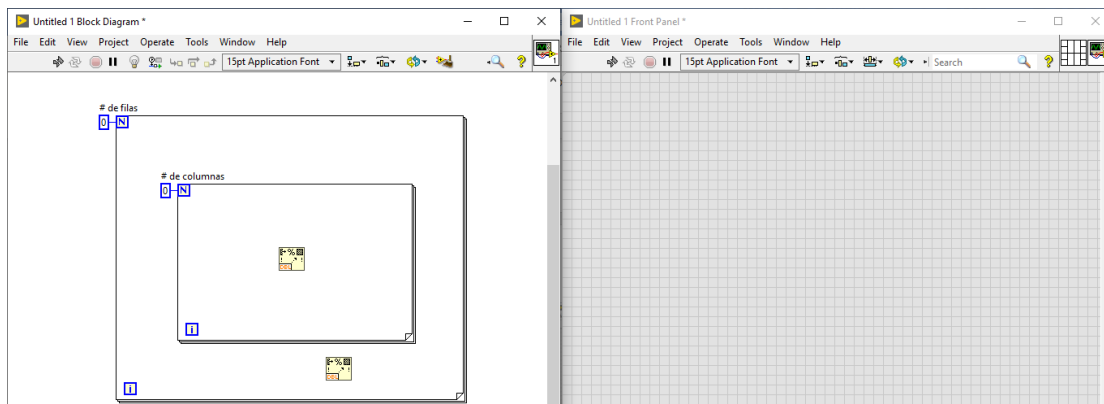
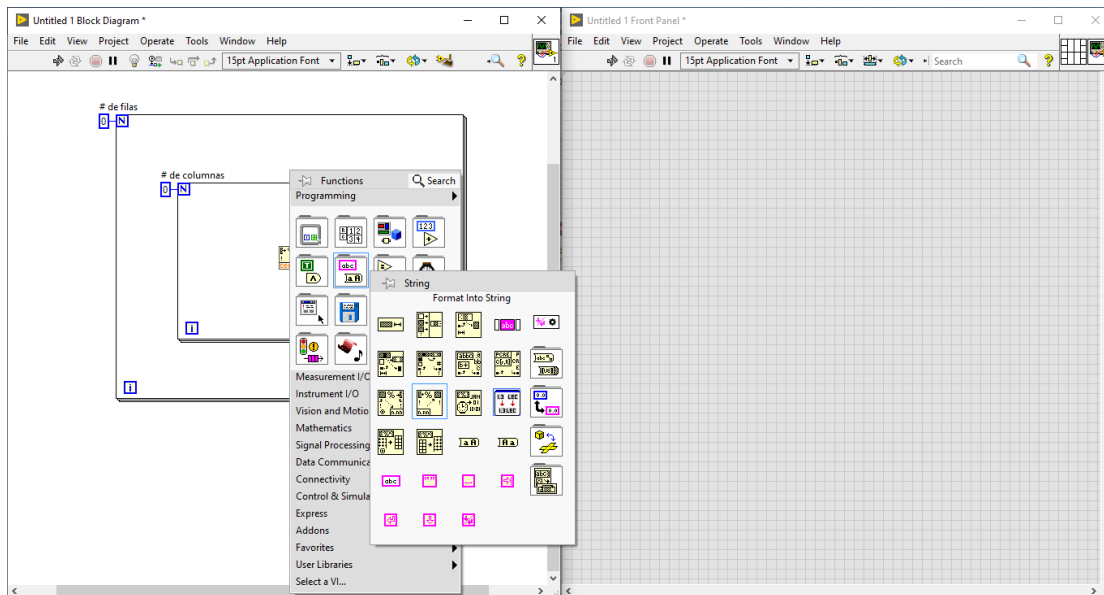
Crear una Constante para un Bloque: Clic derecho en la terminal del bloque de interés → Create → Constant. Esta constante debe ser igual a $N = \text{Número de Columnas}$.



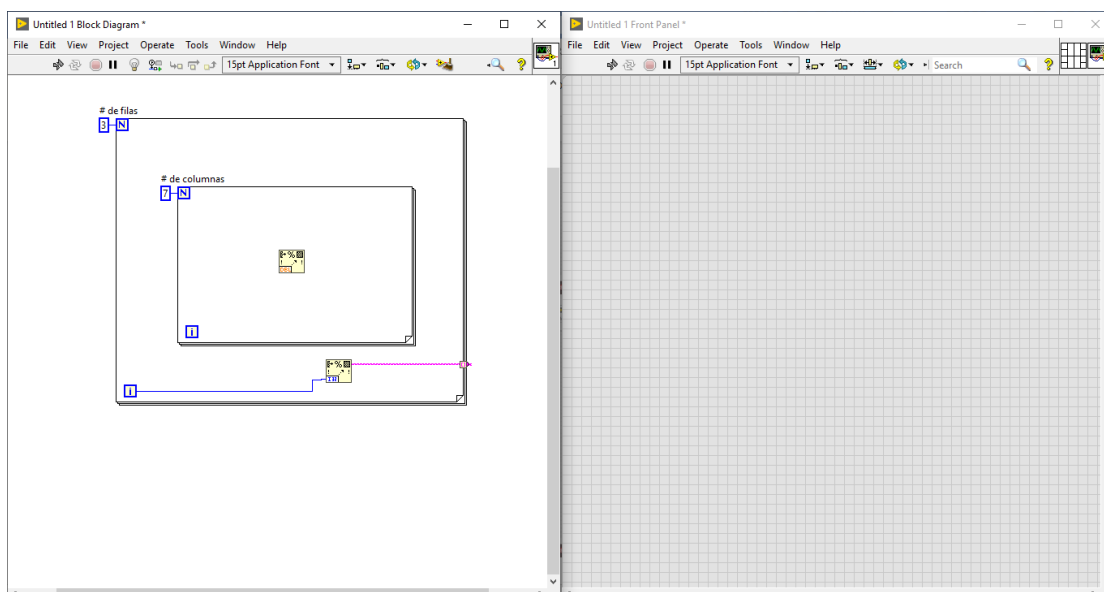
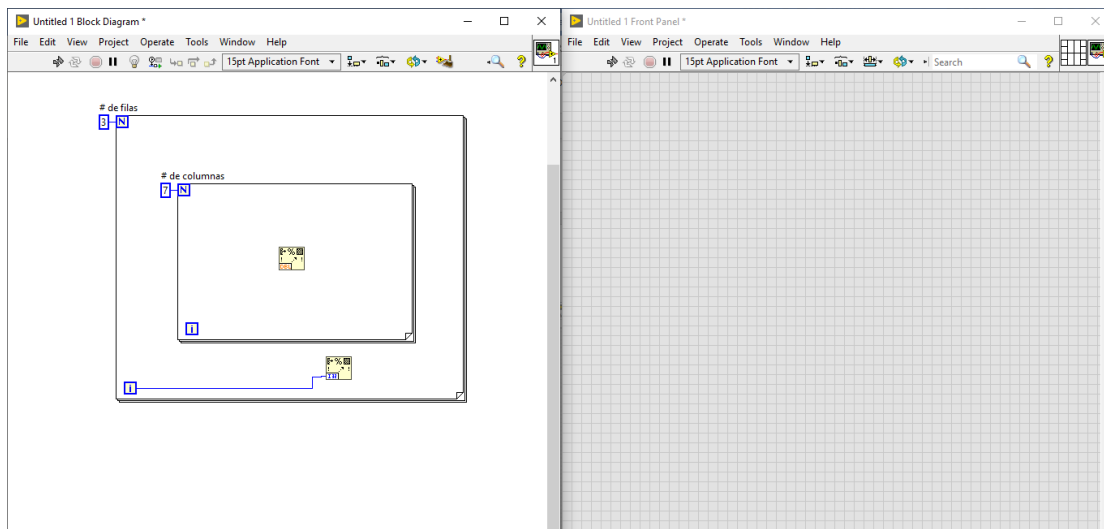
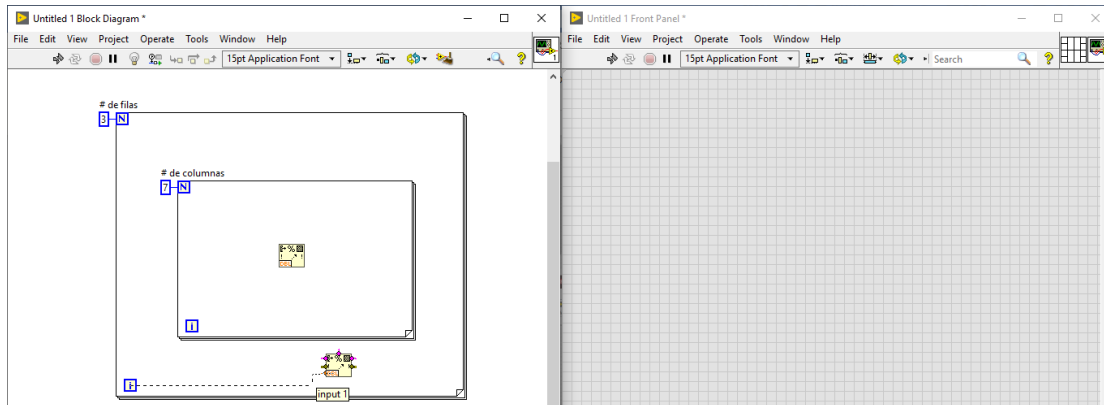


Block Diagram - Format Into String: Conversión de Cualquier Tipo de Dato a un String

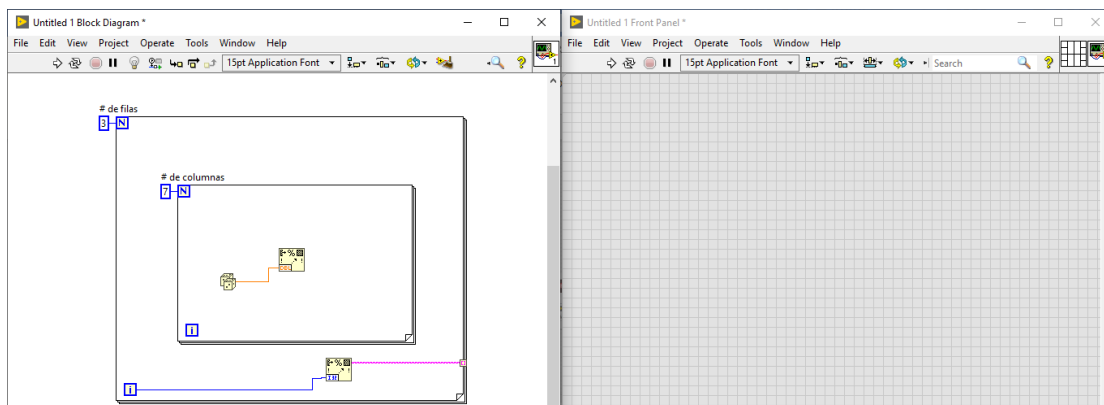
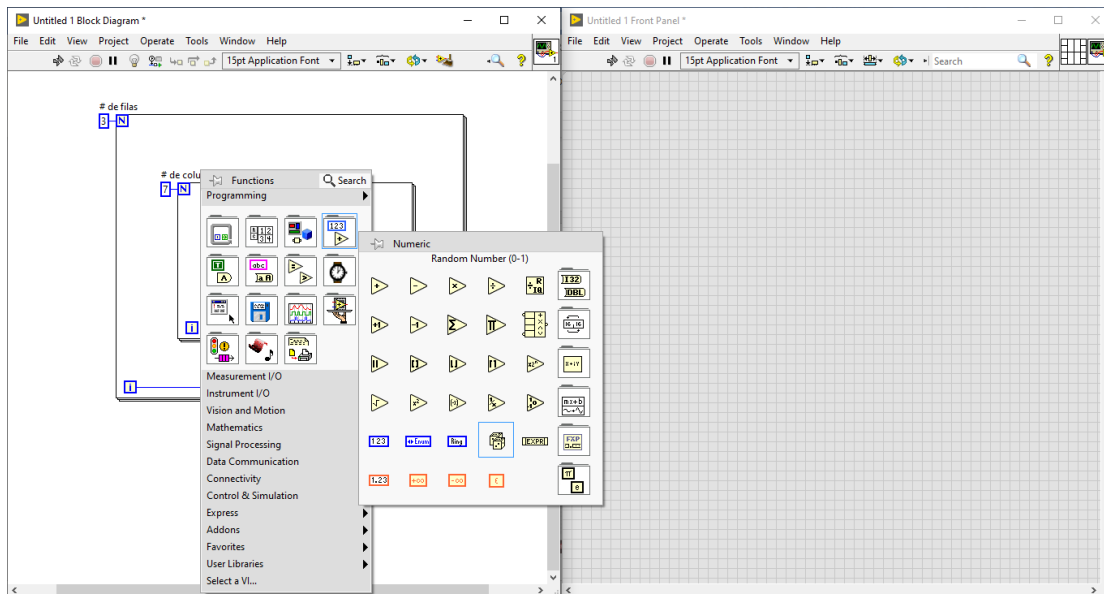
El bloque Format into string sirve para convertir cualquier tipo de dato a un String, ya sea numérico, booleano, dinámico, etc.



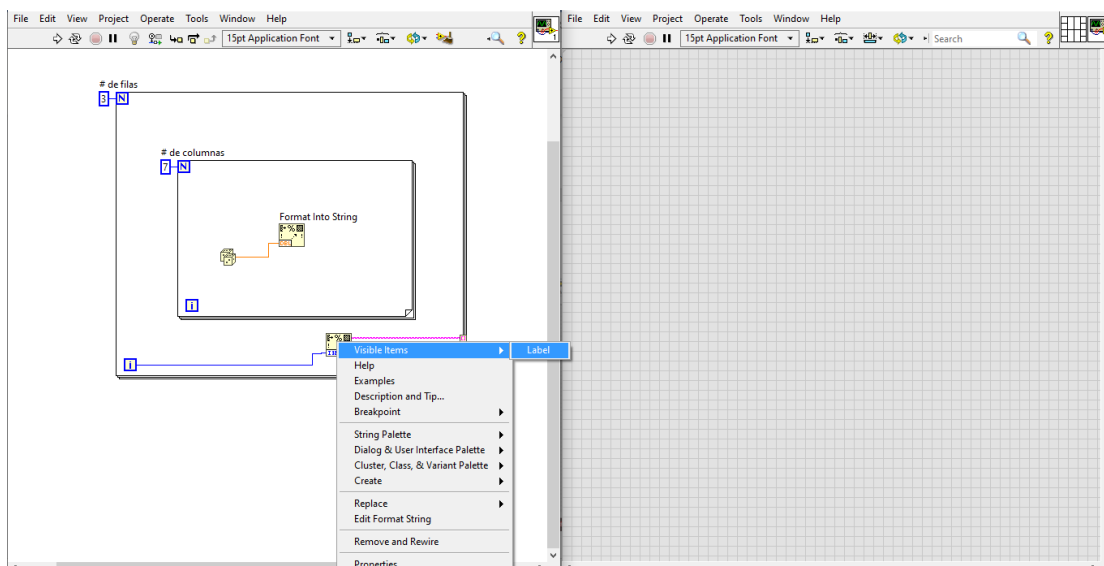
Se conecta la entrada del bloque Format Into String a la variable i del bucle for para que reciba esos números y los mande hacia fuera del bucle por medio de un túnel que todavía no está indexado, osea que no crea un vector de los datos que va recopilando.

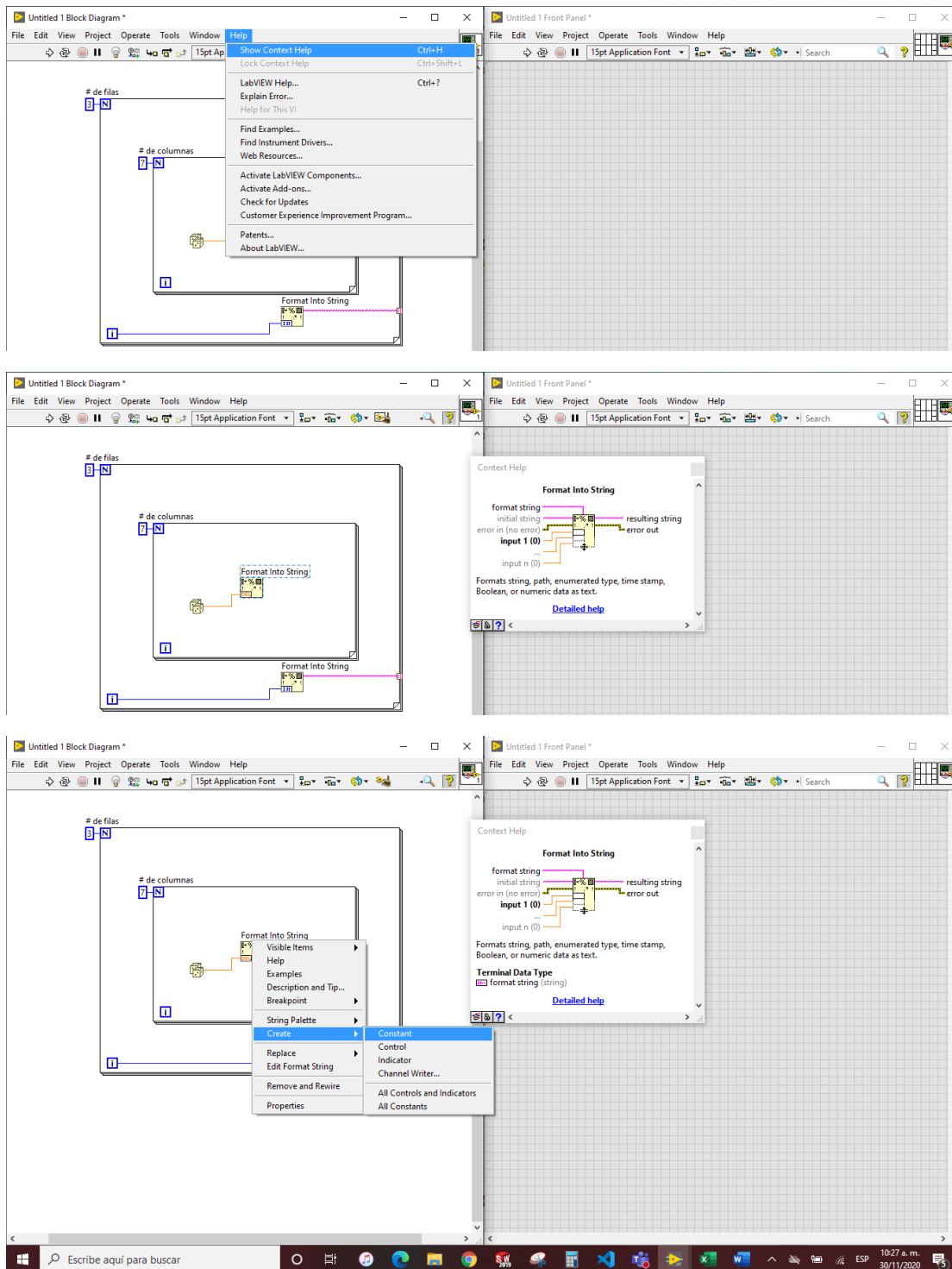


Block Diagram - Random Number: Creación de Número Aleatorio



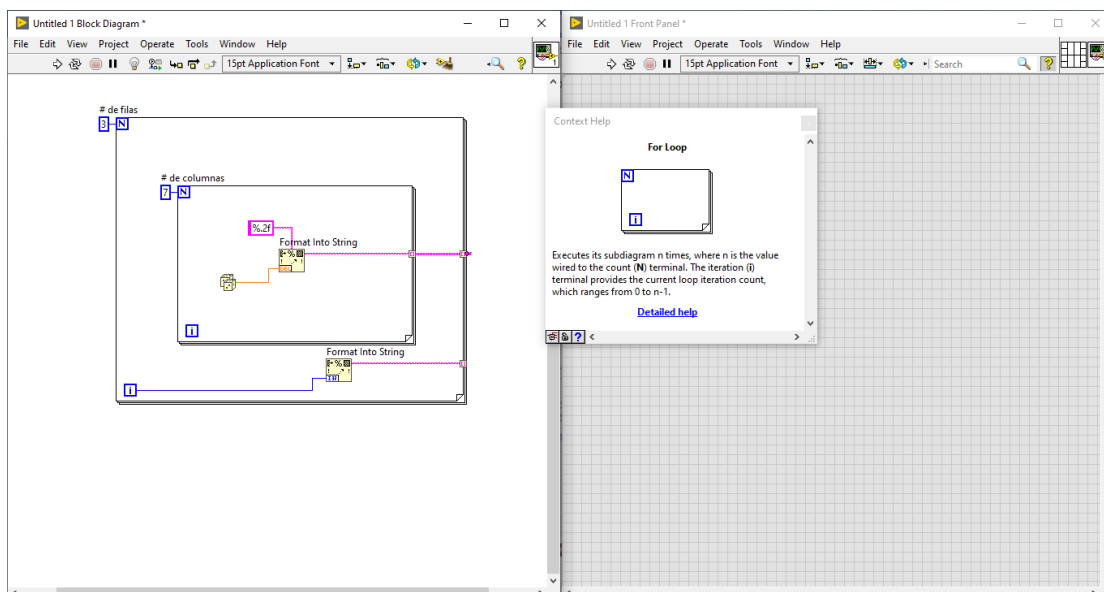
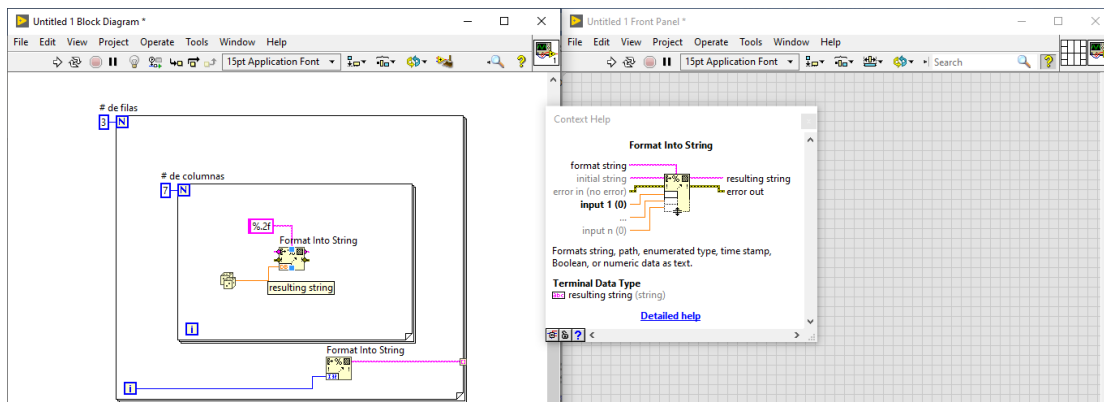
Mostrar nombre del bloque: Clic derecho → Visible Items → Label.



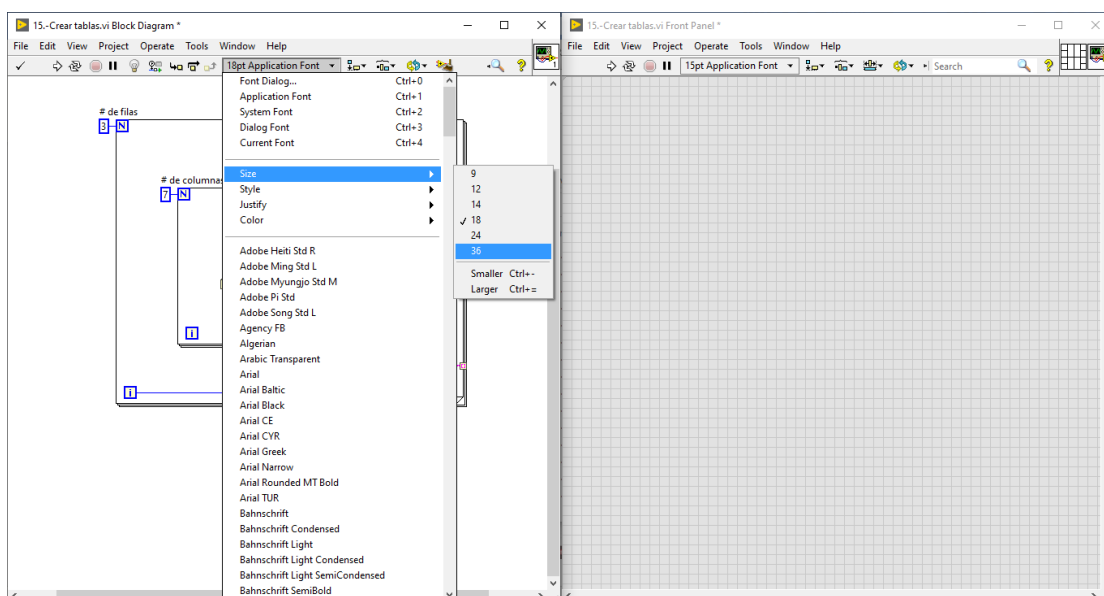


En la terminal de format string, que es la primera debemos indicar al bloque por medio de un código el formato que le va a dar al texto incluido (el número del bloque Random Number):

- Vamos a poner el código **%2f**:
 - **%2f es porque le voy a dar 2 cifras decimales a los números** de ese tipo.



Cambiar Tamaño de Letra: Menú Superior → Application Font → Size → Tamaño de Letra.

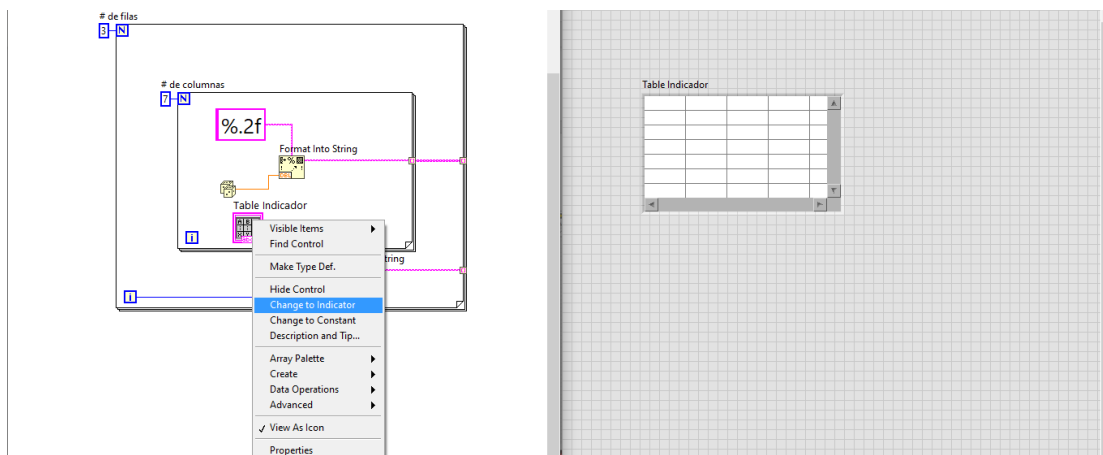


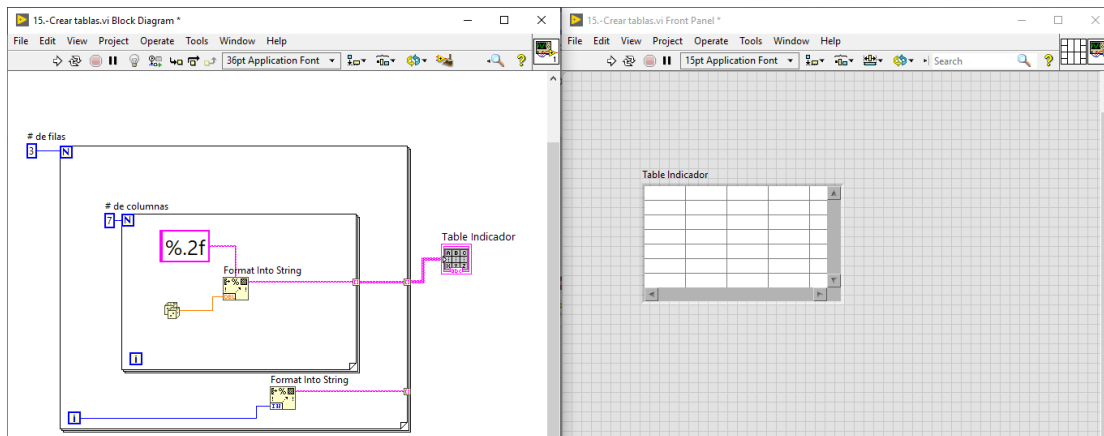
Block Diagram - Table: Creación de una Tabla con Strings

El bloque Table sirve para crear una tabla que contenga varios datos de tipo String, que además deben estar organizados y creados dentro de dos bucles for, uno que indique su número de filas y el otro el número de columnas, donde además se cuenta con la creación y/o manejo de datos de tipo String.



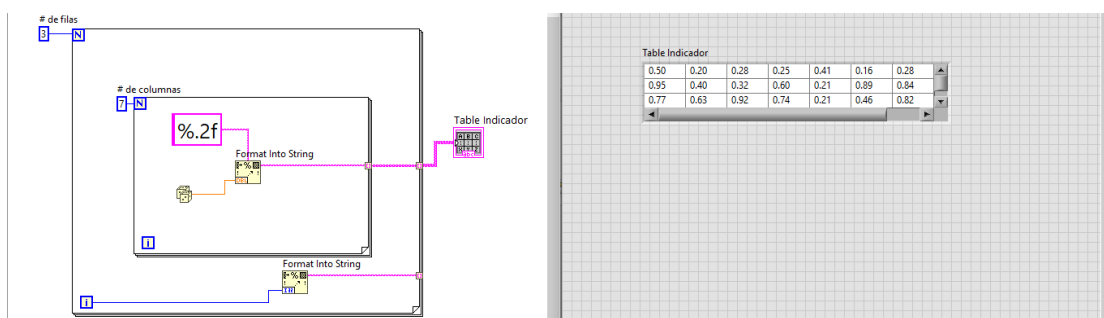
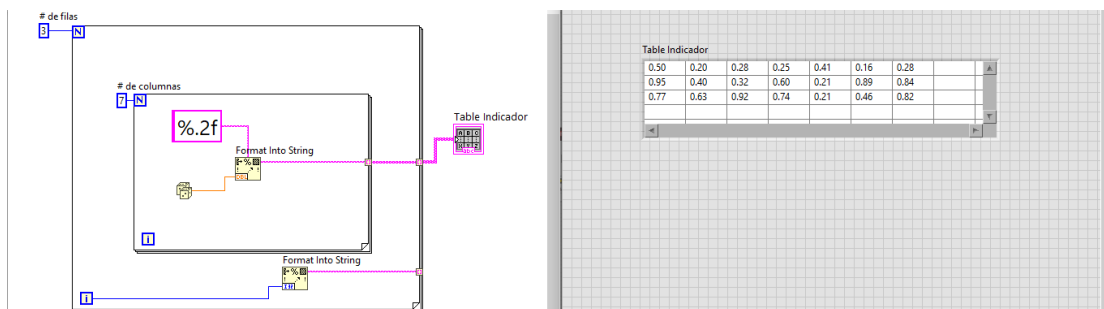
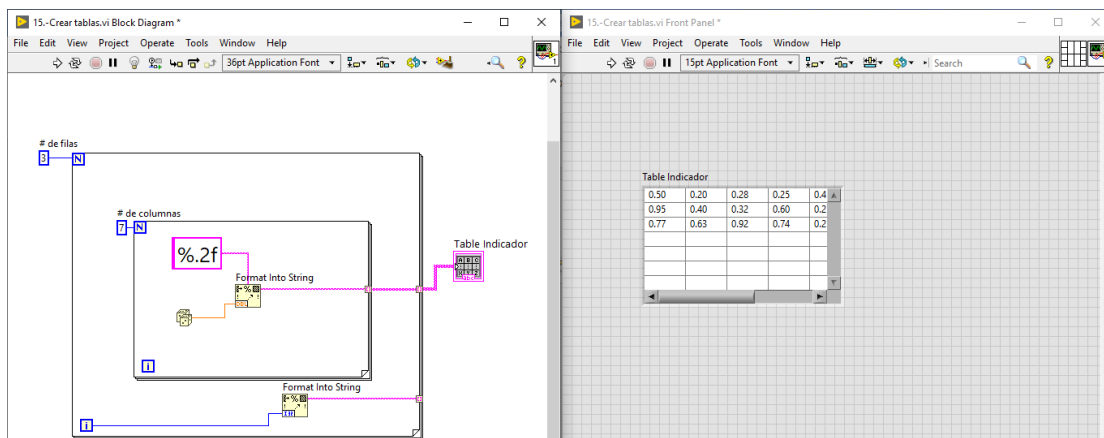
Cambiar un control a que sea Indicador: Clic derecho en el bloque → Change to Indicator.





Ejecución del Programa: Crear una Tabla de Datos en LabVIEW

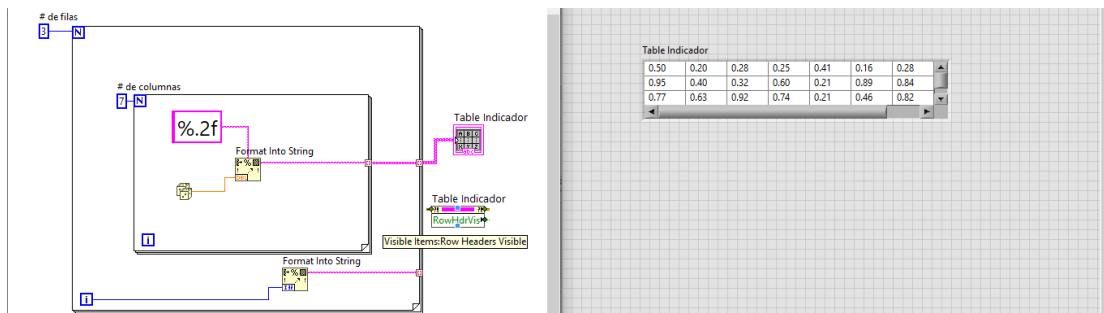
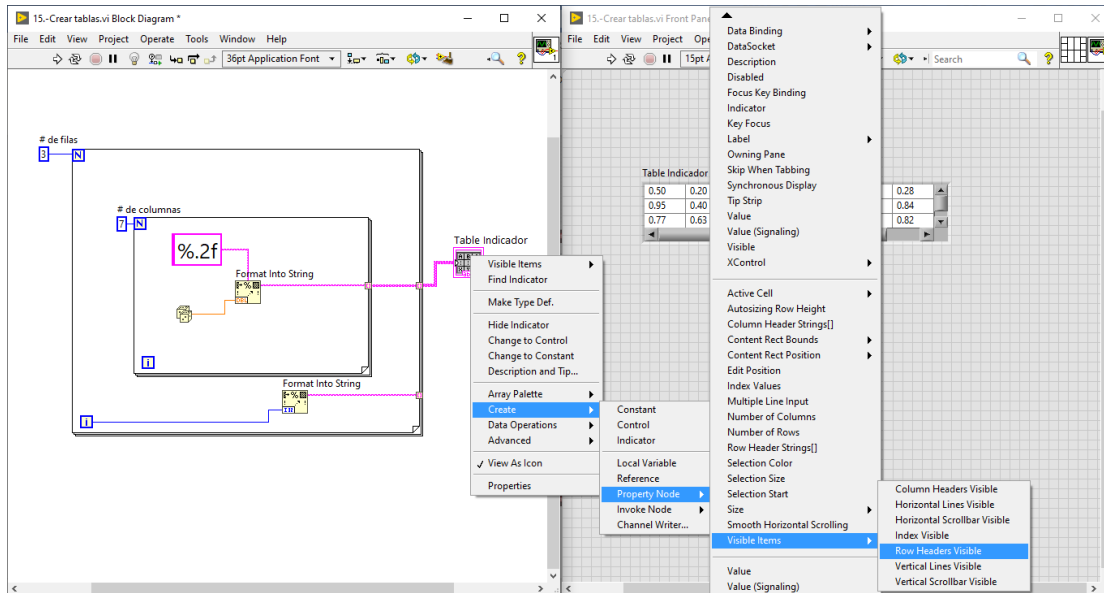
Al correr el programa saldrán los números aleatorios creados por el bloque Random Number en la tabla.



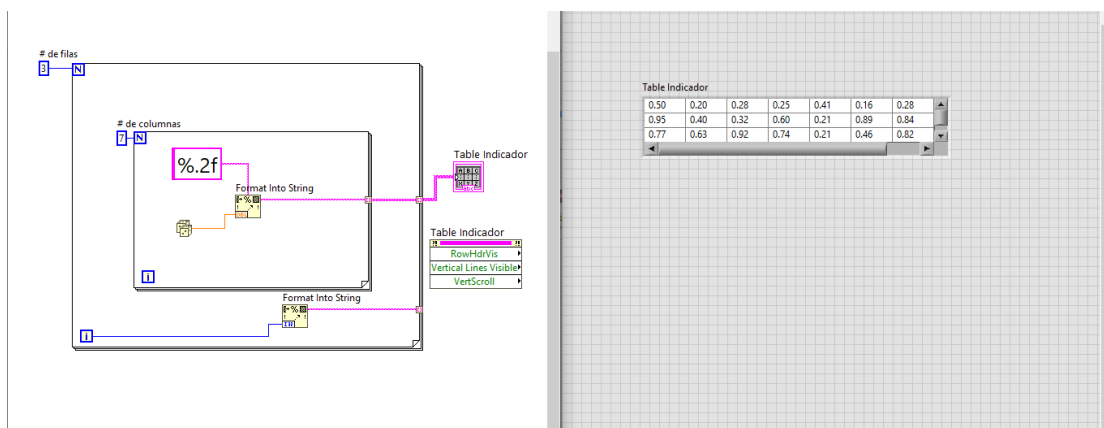
Block Diagram - Property Node: Propiedad Específica de Cualquier Bloque

Ahora lo que voy a hacer es acceder a las propiedades de la tabla dando clic derecho en el bloque de la siguiente manera y seleccionando la opción de Create → Property Node → Visible Items → Row Headers Visible. Con esto editaré el nombre de las filas en la tabla:

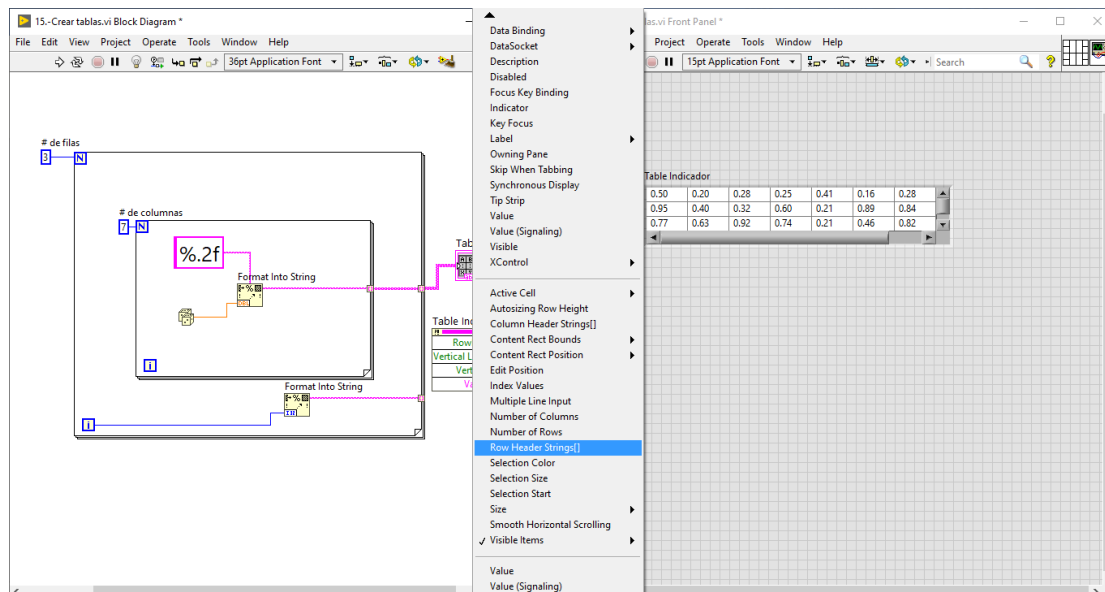
Recordemos que un Property Node es un elemento que accede a cualquier aspecto del bloque indicado, puede ser el historial, color, valor, **nombre de columnas o filas en una tabla**, etc.



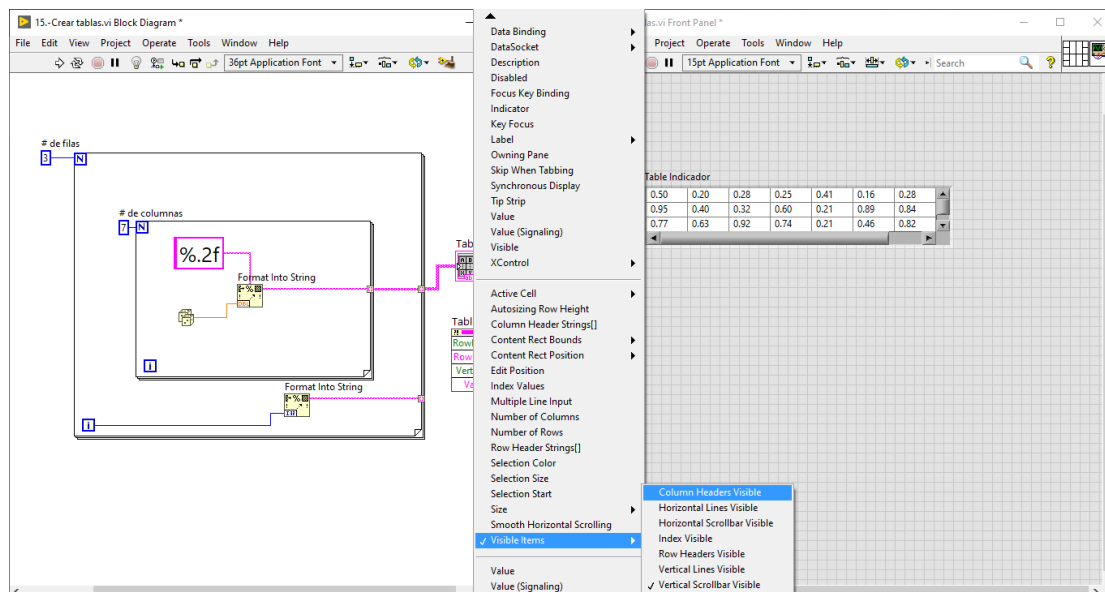
Se debe expandir el bloque para que se pueda acceder a las propiedades del Property Node.



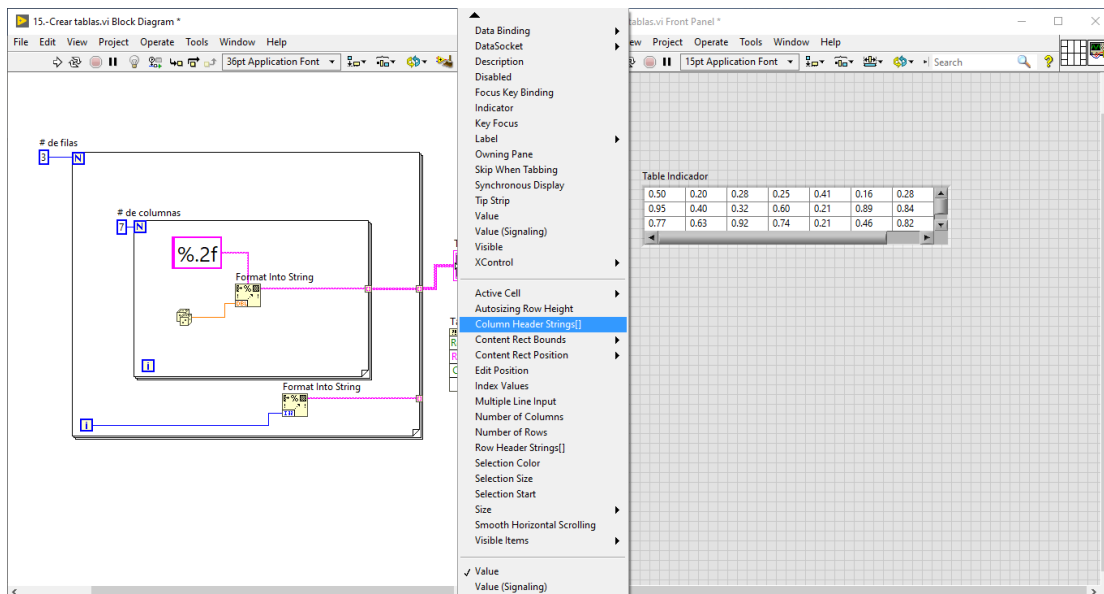
Para poder editar el nombre de las filas en la tabla le doy clic izquierdo a la parte que dice Vertical Lines Visible en el Property Node y voy a buscar la opción de Row Header Strings [].



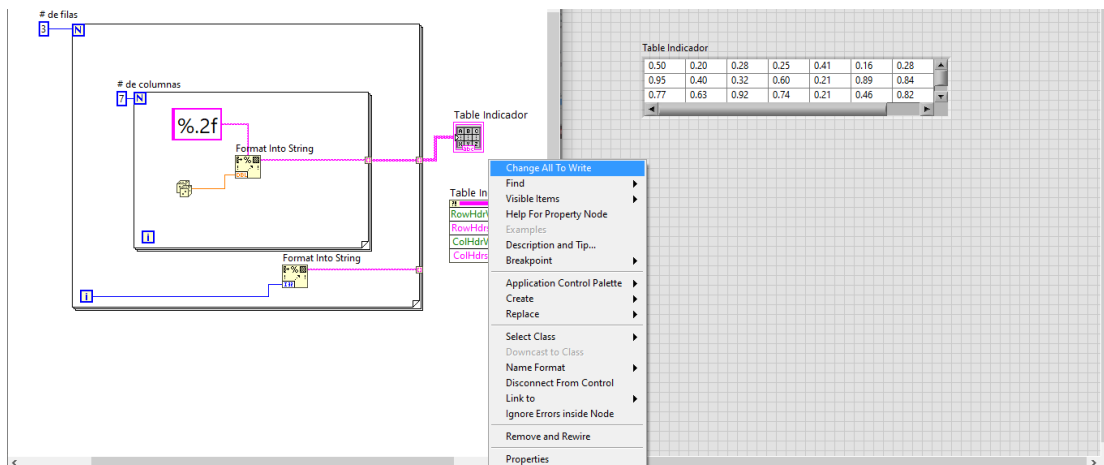
Se realiza lo mismo con el de abajo, pero ahora con la opción Column Headers Visible → Column Header Strings [] para poder editar los nombres de las columnas.



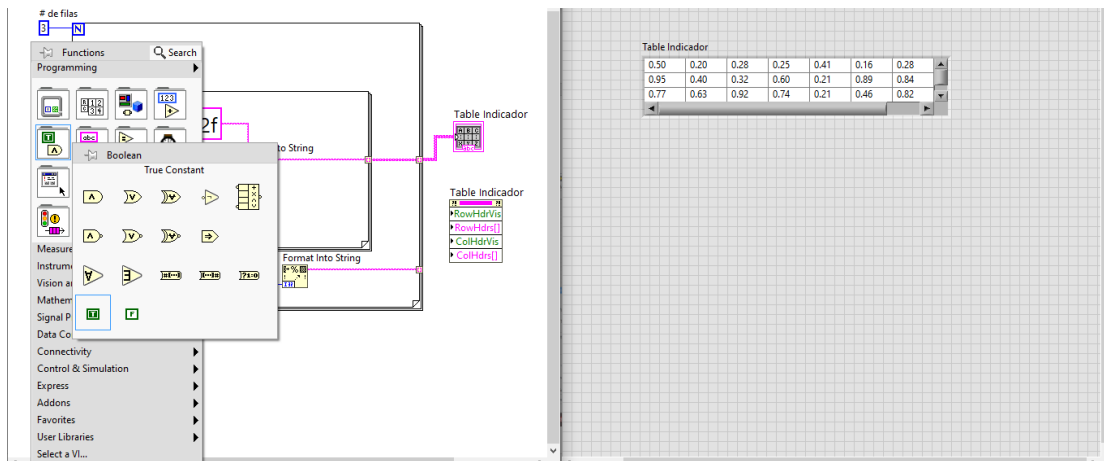
Esto se realiza para dar nombres a las filas y columnas de la tabla, dichos nombres van a ser extraídos de los bucles for anidados de las filas y columnas, para que de esta manera todas las posiciones de datos de la tabla tengan una numeración.

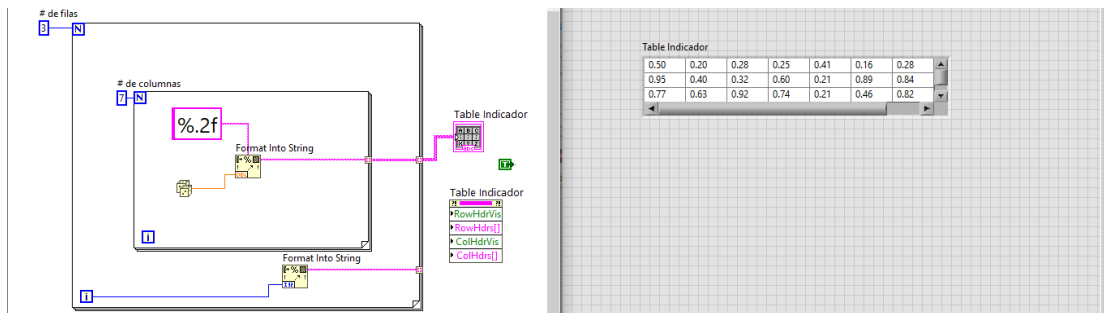


Ahora daré clic derecho en la parte superior rosa del Property Node y seleccionaré la opción de Change All To Write:

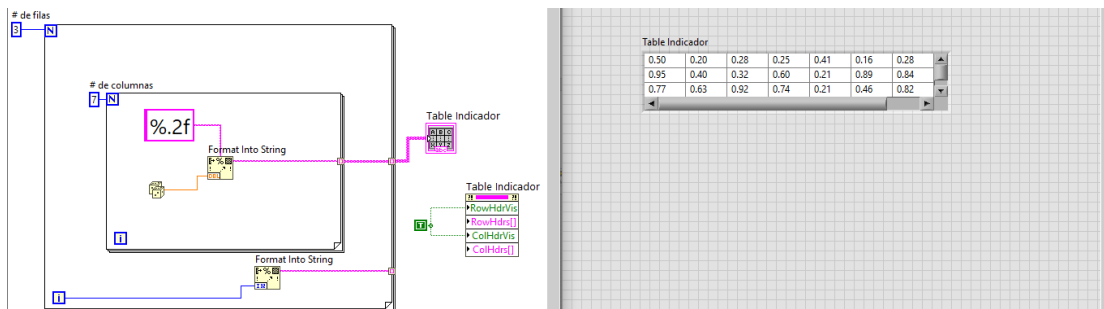


Block Diagram - True Constant: Constante Booleana con Valor Verdadero

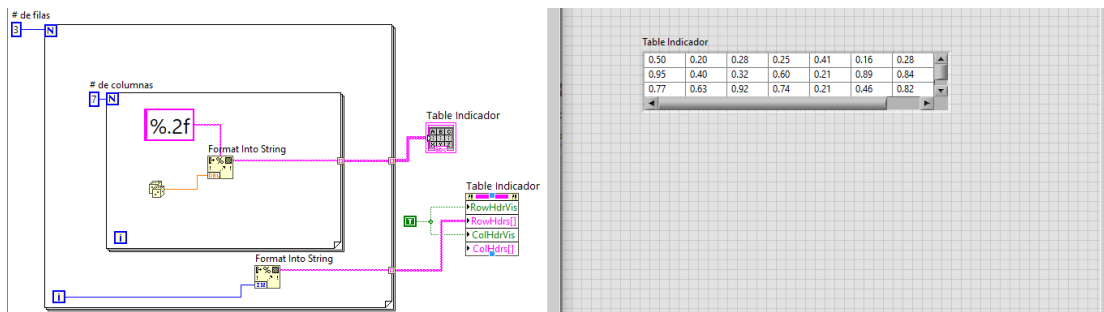




Este valor de True lo que hace es habilitar los encabezados para que aparezcan en la tabla.

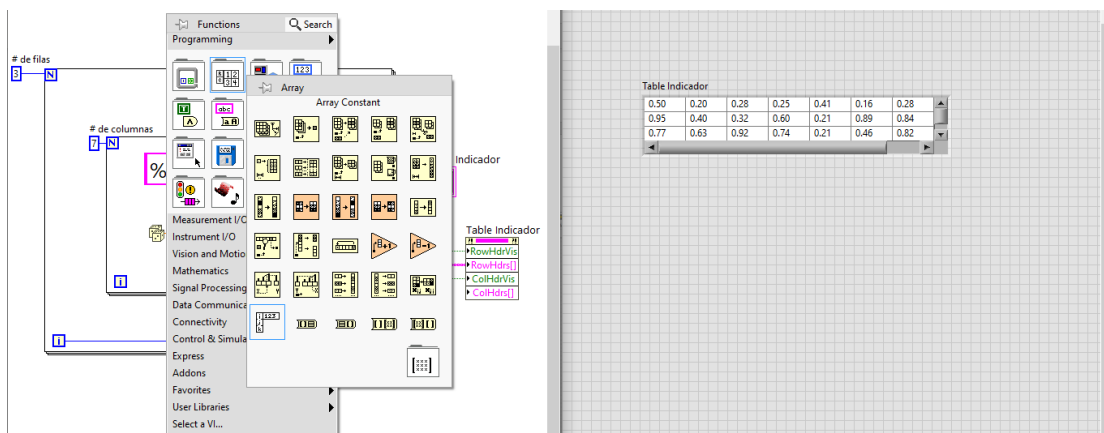


Ahora vamos a conectar los encabezados al bucle for que le corresponda, el encabezado de las filas se conecta de forma directa.



Pero para el encabezado de las columnas vamos a crear un arreglo o array intermedio de tipo String.

Block Diagram - Array Constant: Constante Tipo Array Vacía de 1 Posición



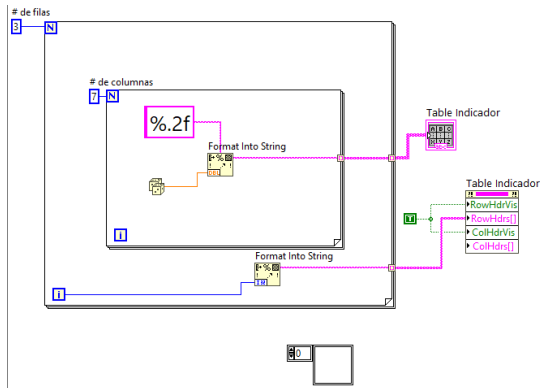


Table Indicador						
0.50	0.20	0.28	0.25	0.41	0.16	0.28
0.95	0.40	0.32	0.60	0.21	0.89	0.84
0.77	0.63	0.92	0.74	0.21	0.46	0.82

Block Diagram - String Constant: Constante Tipo String

Se usan en conjunto los bloques de Array Constant y String constant para que se cree una constante de tipo Array String, para ello se debe arrastrar el bloque de String Constant dentro del Array Constant.

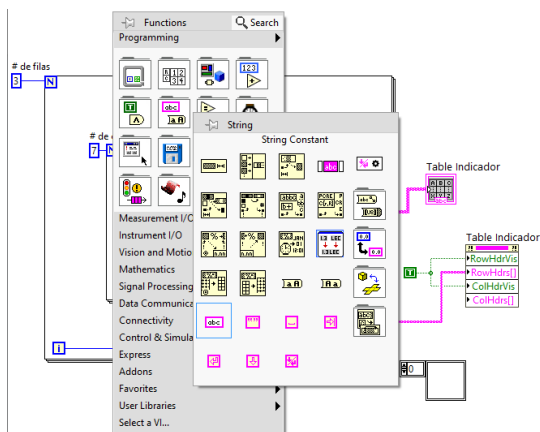


Table Indicador						
0.50	0.20	0.28	0.25	0.41	0.16	0.28
0.95	0.40	0.32	0.60	0.21	0.89	0.84
0.77	0.63	0.92	0.74	0.21	0.46	0.82

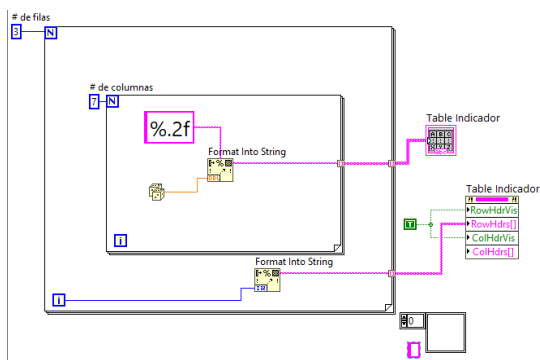


Table Indicador						
0.50	0.20	0.28	0.25	0.41	0.16	0.28
0.95	0.40	0.32	0.60	0.21	0.89	0.84
0.77	0.63	0.92	0.74	0.21	0.46	0.82

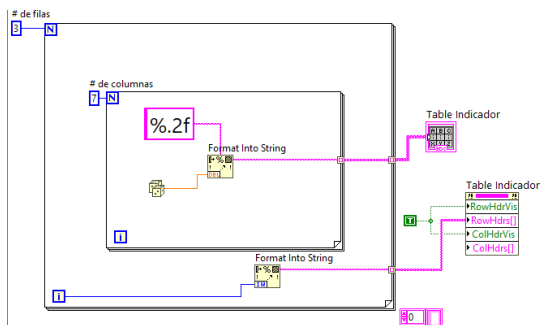
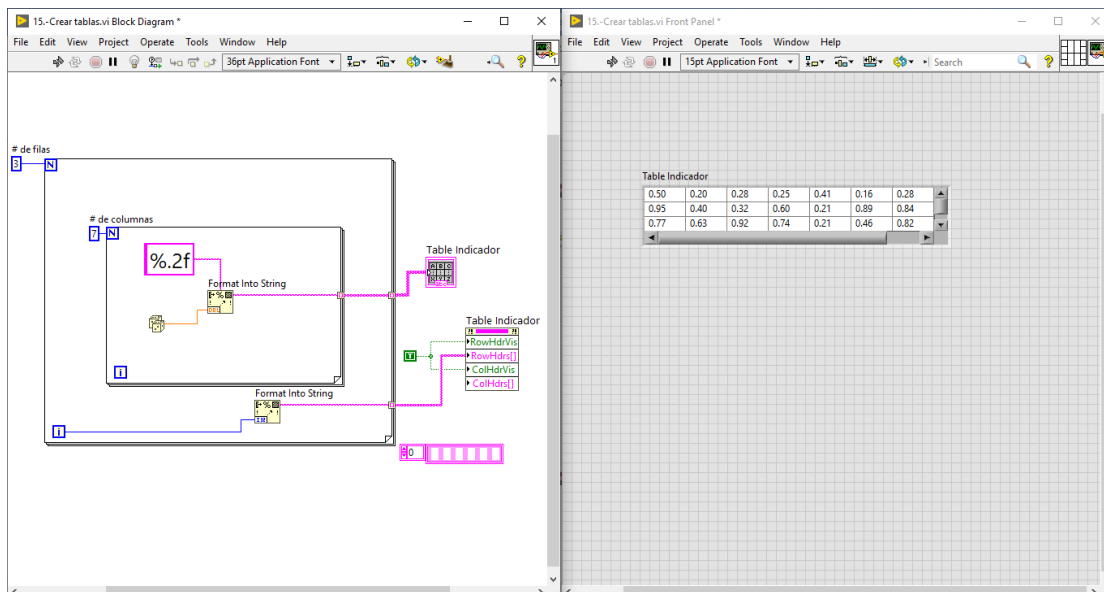
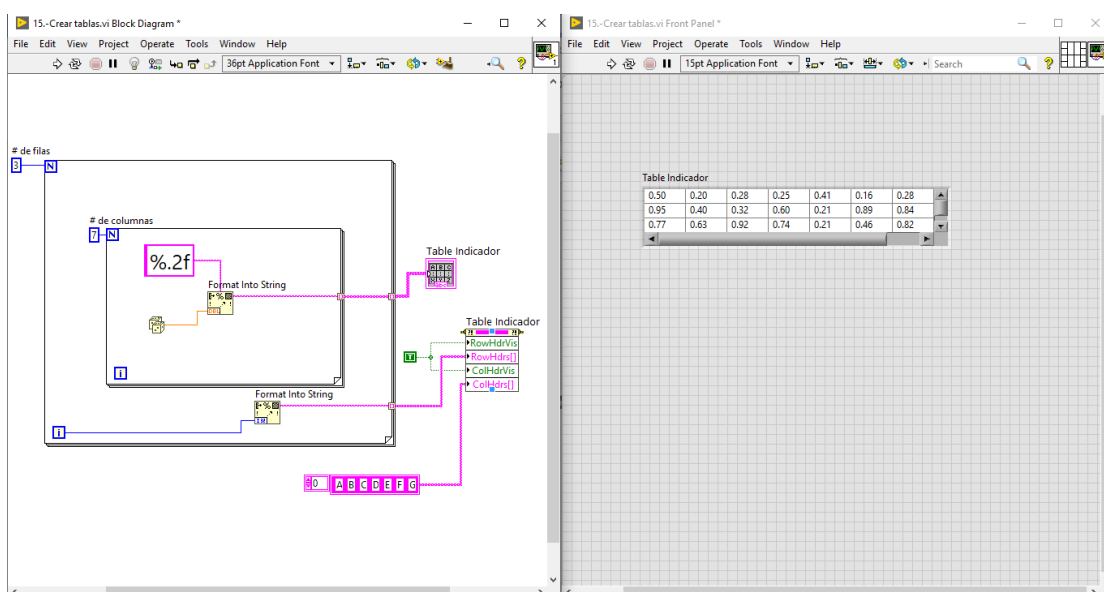
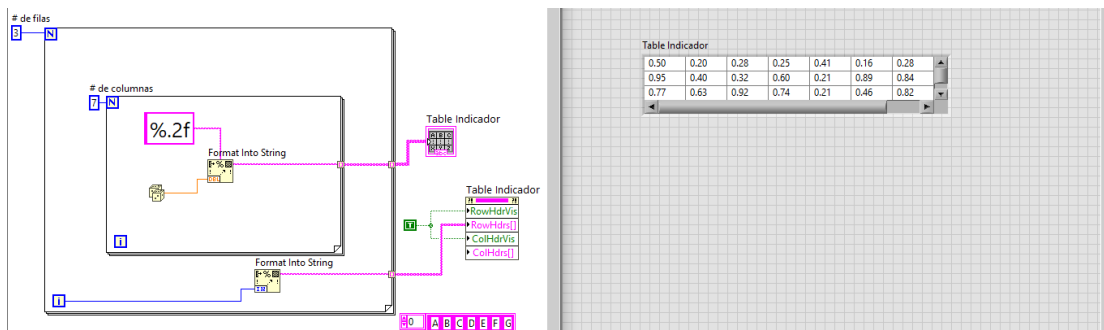


Table Indicador						
0.50	0.20	0.28	0.25	0.41	0.16	0.28
0.95	0.40	0.32	0.60	0.21	0.89	0.84
0.77	0.63	0.92	0.74	0.21	0.46	0.82

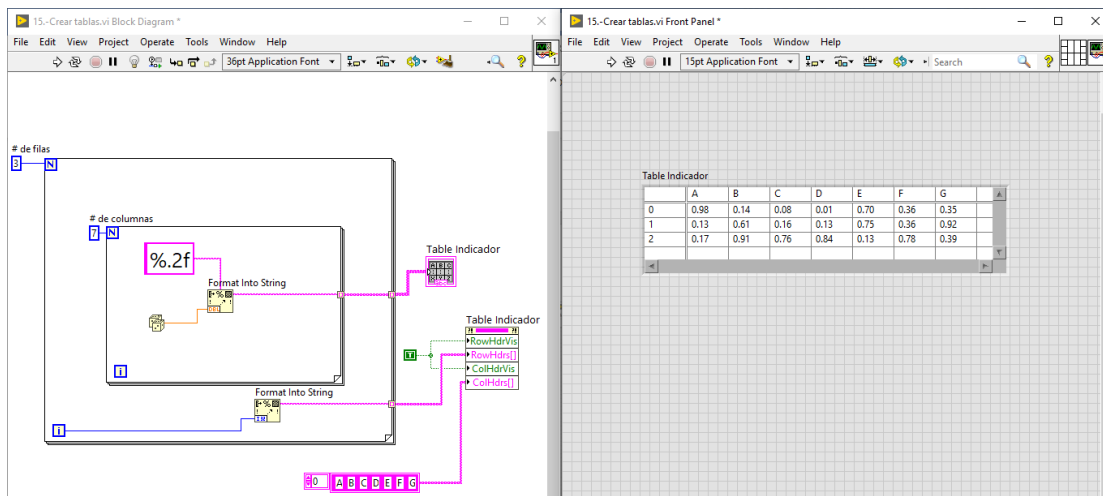
Ahora vamos a expandir esto en 7 lugares, ya que la tabla cuenta con 7 columnas, como se puede ver en la variable N del bucle for anidado.



Ahora le voy a dar nombre a cada casilla, esto corresponderá al nombre de cada columna de la tabla.



Ejecución del Programa: Introducir Datos a una Tabla en LabVIEW



Si cambio el valor de la constante booleana a False en los Property Nodes, se borrarán los encabezados.

