

فصل چهارم

نرم افزار Labview

۴-۱-آشنایی با بسته نرم افزاری Labview

Labview، مخفف عبارت Laboratory Virtual Instrument Engineering Workbench،

نام نرم افزاری قدرتمند است که از آن برای جمع آوری و تحلیل داده ها، بررسی و شبیه سازی محیط های آزمایشگاهی و سیستم های اندازه گیری و... استفاده می شود. این نرم افزار که برای ساده کردن برنامه نویسی از زبان گرافیکی مخصوصی موسوم به «G» بهره می برد، در صنایع، تحصیلات، آموزش و تحقیقات آزمایشگاهی کاربرد ویژه ای یافته است. به کمک این نرم افزار می توان داده ها را از محیط جمع آوری کرد، پردازش های مختلفی روی آن انجام داد و در نهایت اطلاعات حاصل را با امکانات مختلفی از جمله نمایش نموداری، ذخیره فایل و... ارائه داد. برخی ویژگی های این نرم افزار به شرح زیر است:

- زبان برنامه نویسی گرافیکی : برنامه نویسی در Labview با روش های برنامه نویسی متنی متداول متفاوت است. در این روش برنامه نویسی که اصطلاحاً به آن «G» گفته می شود، دستورات به صورت بلوک هایی در برنامه قرار می گیرند و برای جریان داده بین آنها، این بلوک ها به هم سیم کشی می شوند. کسانی که با قسمت Simulink نرم افزار متلب کار کرده باشند، بیشتر متوجه این مزیت می شوند. کار کردن با این نرم افزار ساده است و این در حالی است که دستورات قدرتمند آن طیف وسیعی از کاربردها را در بر می گیرد.
- محیط کاربر پسند : محیط برنامه نویسی در Labview، شامل دو بخش صفحه ی پانل (Front Panel) و صفحه ی نمودار بلوکی می شود. صفحه ی پانل برای ساده کردن ارتباط کاربر با برنامه ایجاد شده است. دستوراتی که در این صفحه وارد می شوند، به دو بخش «کنترلها» و «نشان دهنده ها» تقسیم می شوند. کنترل ها محلی برای نگهداری داده های ورودی، و نشان دهنده ها برای نمایش مقادیر حاصل از پردازش به کار می روند. این دستورات در صفحه ی نمودار بلوکی نمادی مخصوص خود دارند. نمودار بلوکی، بخش اجرایی برنامه است که در آن از دستوراتی مثل حلقه ها یا ساختارهای تصمیم گیری استفاده می شود. همه ی دستورات نمودار بلوکی لزوماً دارای نمادی در صفحه ی پانل نیستند.
- وجود مثال های متعدد و پشتیبانی قوی : در بسته ی نرم افزاری Labview، مثال های بسیاری از نحوه ی کار کردن با دستورات آن وجود دارد. این مثال ها، و همچنین برنامه های ایجاد شده توسط کاربر را می توان به صورت یک دستور در برنامه ی دیگری به کار برد؛ بدون آنکه احتیاج به تبدیل برنامه به فایل کتابخانه ای خاصی، به طوری که در زبان های برنامه نویسی متداول انجام می شود،

باشد. امکان جستجوی یک دستور خاص در Labview وجود دارد. این نرم افزار شامل بخش Help قدرتمندی می شود. علاوه بر آن سایت اینترنتی تولید کنندگان این بسته نرم افزاری، پشتیبانی از آن را بر عهده دارد.

- امکانات متنوع : علاوه بر وجود دستورات قوی در Labview، بسته های نرم افزاری دیگری برای اهداف خاصی تولید شده و به همراه این نرم افزار به کار برده می شوند. بسته ی نرم افزاری بینایی ماشین و پردازش تصاویر دیجیتال از جمله این امکانات است. این امکانات، به کاربر توانایی ویژه ای جهت کنترل سخت افزار کامپیوتر می دهند.
- اصل جریان داده : همان طور که می دانیم، در زبان های برنامه نویسی متنی، برنامه خط به خط اجرا می شود تا به پایان برسد. استفاده از حلقه های (while) تودرتو مجاز نیست و در هر لحظه تنها در یک قسمت برنامه هستیم. اما در Labview اجرای برنامه به صورت «موازی» انجام می شوند؛ به این معنی که در یک لحظه می تواند چندین قسمت برنامه و یا مثلاً دو حلقه while کاملاً مجزا اجرا شود. این در حالی است که به کمک بعضی دستورات می توان برخی، و یا همه ی قسمت های برنامه را به صورت ترتیبی انجام داد.

۴-۲- دستورات کاربردی Labview

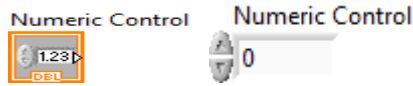
در ادامه به توضیح مختصر دستورات به کار رفته در پروژه پرداخته می شود. این دستورات در جعبه ابزارهایی که بر روی صفحه ی پانل و نمودار بلوکی (پالت کنترل و پالت توابع) ظاهر می شوند، موجودند. آیکن هایی که به همراه توضیحات می آیند، نماد دستور هستند. اگر دستوری در هر دو صفحه ی پانل و نمودار بلوکی دارای نماد باشد نماد سمت راست نشان دهنده ی آیکن صفحه ی پانل و نماد سمت چپ نشان دهنده ی آیکن نمودار بلوکی است. در غیر این صورت آیکن مربوط به نمودار بلوکی آورده شده است. اسم دستور در بالای آن به صورت برجسته آمده است. در بخش نرم افزار پروژه در مورد کار با Labview بیشتر توضیح داده خواهد شد.

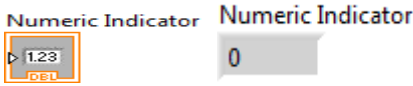
۴-۲-۱- متغیرها و ثابت ها

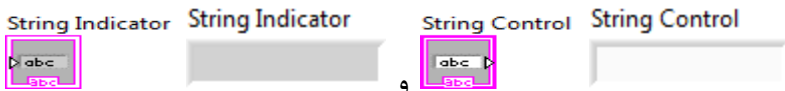
همانند سایر زبان های برنامه نویسی، متغیر به مکانی از حافظه که داده ای در آن ذخیره می شود، گفته می شود. سه نوع اصلی متغیرها در زبان G، متغیرهای دودویی، عددی و رشته ای هستند. تقسیم بندی

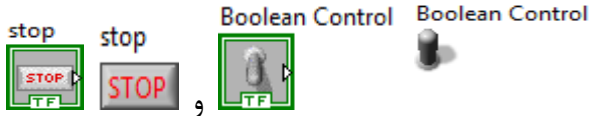
متغیرهای عددی در Labview با زبان های برنامه نویسی دیگر مشابه است. متغیرهای رشته ای در این زبان به صورت کاراکتر وارد شده نمایش داده می شوند ولی در محاسبات مقدار کد اسکی آن در نظر گرفته می شود.

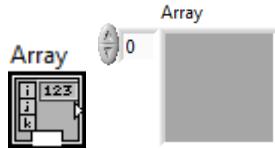
هر سه نوع داده می تواند به صورت آرایه باشد. این داده ها می توانند به طور دستی وارد شوند و یا از طریق محاسبات تولید شوند. مثلاً با استفاده از ساختارهای while و for و ویژگی «اندیس گذاری خودکار» می توان آرایه ها را تولید کرد.

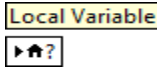
 : بوسیله این دستور می توان یک مقدار عددی را وارد کرد. این مقدار می تواند یکی از انواع صحیح علامت دار و بی علامت، اعشاری و... باشد. با راست کلیک روی آیکن دستور و انتخاب گزینه Representation می توان نوع آن را تغییر داد.


 : این دستور یک مقدار عددی را نمایش می دهد.

 : به ترتیب برای وارد کردن و نشان دادن متغیر رشته ای به کار می روند.

 : کلید هستند و برای وارد کردن مقادیر صحیح یا غلط (صفر و یک) به کار می روند. برای نمایش مقادیر دودویی از LED استفاده می شود.

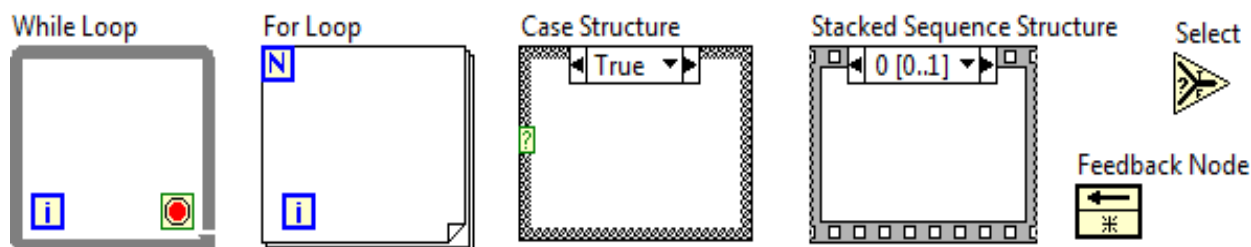
 : آرایه ای را تعریف می کند. این دستور در حالت اولیه خالی است و با قرار دادن یکی از نوع داده های فوق در آن می توان به آرایه ای از آن نوع دست یافت.

 : وقتی می خواهیم از قسمت های مختلف برنامه به مقدار به روز شده ی یک متغیر دسترسی داشته باشیم، از دستور متغیر محلی استفاده می شود. مثلاً به کمک این دستور می توان از یکی از دو حلقه ی while که به طور موازی اجرا می شوند، دیگری را کنترل کرد.

 : به ترتیب ثابتهای دودویی، عددی و رشته ای هستند. ثابت عددی دارای مقدار اولیه صفر است ولی مقدار ثابت رشته ای می تواند تهی باشد. مقدار ثابت دودویی با کلیک روی آن قابل تغییر است.

۴-۲-۲- ساختارهای کنترل جریان داده

این ساختار ها عبارتند از: حلقه while و for، ساختار تصمیم گیری Case، ساختار Sequence، دستور Select و گره ی فیدبکی. اگر سیمی از درون چهار مورد اول به بیرون ساختار کشیده شود، در محل تقاطع سیم با ساختار یک علامت مربع ظاهر می شود. به این عمل اصطلاحاً «تونل زنی» گفته می شود و برای ارسال داده به بیرون آن ساختار استفاده می شود. این ساختارها در شکل زیر مشاهده می شوند.



While loop : اگر شرطی که به المان گوشه پایین سمت راست در داخل حلقه متصل است درست باشد، حلقه متوقف می شود. با کلیک روی المان می توان آن را به شرطی برای اجرا شدن حلقه تبدیل کرد. المان گوشه پایین سمت چپ شمارنده ی حلقه است و تعداد تکرار آن را نشان می دهد.

For loop : دستورات قرارداده شده در این ساختار به تعداد عدد متصل به المان گوشه بالا اجرا می شوند. المان گوشه پایین سمت چپ شمارنده ی حلقه است و تعداد تکرار آن را نشان می دهد. اگر داده ای از درون این حلقه به بیرون سیم کشی شود، آرایه ای تولید می شود. پس از هر بار تکرار حلقه، اندیس آرایه به طور خودکار یک واحد افزایش یافته و داده ی جدید به آن اندیس منتقل می شود. برای غیر فعال کردن ویژگی اندیس گذاری خودکار ساختار، باید بر مربع موجود بر لبه ی ساختار راست کلیک کرده و گزینه ی **Disable Indexing** را انتخاب کرد. لازم به ذکر است که با حلقه ی **while** هم می توان آرایه ایجاد کرد ولی ویژگی اندیس گذاری خودکار در آن به طور پیش فرض غیر فعال است. در این حالت، مقدار نهایی متغیر در هنگام خروج از حلقه به بیرون راه می یابد.

Case Structure : بر حسب شرط یا مقداری که به علامت سؤالی که روی ساختار قرار دارد متصل شده است، تصمیم گیری می کند و دستورات موجود در یک فریم را انجام می دهد. اگر یک مقدار دودویی به علامت شرط متصل باشد، ساختار مانند دستور **if-else** در زبان C عمل می کند. مقادیر عددی و رشته ای را هم می توان به عنوان شرط به کار برد که در این حالت بر حسب مقادیر این متغیر تصمیم گیری می شود و ساختار می تواند بیشتر از دو فریم داشته باشد. فریم ها با کلیک در بالای ساختار قابل تغییر هستند. توجه شود که مقدار متغیرهای خروجی از این ساختار باید در همه ی حالت ها مشخص باشد وگرنه علامت مربع محل خروج سیم از ساختار سفید رنگ می شود و برنامه اجرا نمی شود.


Stacked Sequence Structure : از این ساختار برای ایجاد ترتیب در اجرای دستورات استفاده می شود. هنگام اجرای آن، ابتدا دستوراتی که در فریم اول قرار دارند اجرا می شوند. سپس دستورات فریم دوم اجرا شده و به همین ترتیب تا اجرای ساختار تکمیل شود. تا زمانی که دستورات یک فریم به طور کامل انجام نشود، به مرحله ی بعد وارد نمی شویم. با راست کلیک در لبه ی سمت راست ساختار و انتخاب گزینه ی **add frame** می توان فریم بعدی را اضافه نمود. توجه شود که دو نوع ساختار توالی وجود دارد. در نوع اول، همه ی فریم ها مانند فریم های یک فیلم در یک صفحه و پشت سر هم می آیند (**Flat**). در این نوع دستور (**Stacked**)، فریم ها در صفحات پشت سر هم می آیند. استفاده از نوع دوم جای کمتری از فضای برنامه را اشغال می کند ولی کاربرد دستور نوع اول ساده تر و قابل فهم تر است.

Select : بوسیله ی این دستور می توان بر حسب صحیح یا غلط بودن یک شرط یکی از دو مقدار که به سرهای صحیح و غلط دستور متصل شده اند را انتخاب کرد.


Feedback Node : از طریق این دستور می توان به مقدار قبلی متغیر دست یافت. ترکیب این دستور با عملی مانند جمع با یک، و اعمال آن روی یک متغیر عددی مثل نوشتن دستور **"i++;"** در زبان C است.

۴-۲-۳- دستورات محاسباتی و منطقی


این دستورات عملیاتی نظیر جمع، تفریق، ضرب، عملیات منطقی مثل مقایسه و... را بر روی داده ها انجام می دهند. برای متغیرهای رشته ای، مقدار کد اسکی کاراکتر در محاسبات منظور می شود.

Add : دستور جمع. مقدار دو متغیر را با هم جمع می کند و در خروجی قرار می دهد. 


Increment

 : مقدار متغیر وصل شده به آن را یک واحد افزایش می دهد.


Divide

 : عملیات تقسیم را بر روی دو متغیر انجام داده و حاصل را در خروجی قرار می دهد.


Equal?

 : مقدار دو متغیر را با هم مقایسه می کند و اگر مساوی بودند مقدار صحیح (یک) را بر می گرداند.


Greater Or Equal?

 : مقدار دو متغیر را با هم مقایسه می کند و اگر بزرگتر و مساوی بودند مقدار صحیح (یک) را بر می گرداند.


Less Or Equal?

 : مقدار دو متغیر را با هم مقایسه می کند و اگر کوچکتر و مساوی بودند مقدار صحیح (یک) را بر می گرداند.


Not Equal To 0?

 : اگر مقدار متغیر با صفر برابر نباشد، مقدار درست (یک منطقی) را بر می گرداند.

And

 : اگر هر دو متغیر بولی متصل شده به دستور مقدار درست داشته باشند، مقدار درست را بر می گرداند.

Or

 : اگر حداقل یکی از دو متغیر بولی متصل شده به دستور مقدار درست داشته باشند، مقدار درست را بر می گرداند.

۴-۲-۴-دستورات کار با آرایه ها و رشته ها

به کمک این دستورات می توان داده های رشته ای را تولید و دستکاری کرده و روی آنها عملیات متنوعی انجام داد. نرم افزار Labview دستورات بسیار قدرتمندی برای دستکاری داده های رشته ای در اختیار کاربر می گذارد.

الف) دستورات کار با رشته ها :

Concatenate Strings



دو یا چند داده ی رشته ای را دریافت می کند و آن ها را با هم ترکیب کرده تا یک رشته واحد ایجاد شود.

String Length



طول یک متغیر رشته ای را به صورت یک عدد صحیح بر می گرداند.

String Subset



از مقدار وصل شده در سر **offset** شروع می کند و به اندازه مقدار وصل شده در سر **Length** از داده رشته ای ورودی می خواند و در خروجی قرار می دهد.

Search and Replace String



در رشته ی ورودی، عبارت وصل شده به سر **search** را جستجو می کند و در صورت یافتن، عبارت وصل شده در سر **replace** را جایگزین آن می کند و حاصل را در خروجی قرار می دهد. اگر در سر **replace all?** یک ثابت بولی با مقدار درست وصل شود، همه ی عبارات یافته شده جایگزین می شوند وگرنه فقط عبارت اول جایگزین می شود.

Number To Decimal String



یک مقدار عددی را به یک رشته تبدیل می کند.

Decimal String To Number



یک رشته حاوی کاراکترهای عددی را به یک عدد صحیح تبدیل می کند. این دستور از ابتدای رشته جستجو می کند و اگر کاراکتر عددی یافته نشد، هیچ مقداری برگردانده نمی شود.

Format Date/Time String



تاریخ و ساعت سیستم را در لحظه ی اجرای دستور با فرمت خاصی به رشته تبدیل می کند و در خروجی قرار می دهد. تعیین فرمت توسط کاربر با مقداردهی مناسب در سر **time format** **string** قابل انجام است. مثلاً با اتصال رشته **"%C"** به این سر، تاریخ و ساعت سیستم عیناً کپی می شود. در مورد این دستور در بخش های بعدی بیشتر توضیح داده می شود.

(ب) دستورات کار با آرایه ها

Index Array



: توسط این دستور می توان عنصری با اندیس مشخص از آرایه ورودی را خواند و در خروجی قرار داد.

Array Size



: اندازه ی آرایه ورودی به دستور را به صورت عدد صحیح بر می گرداند.

۴-۲-۵-دستورات کار با مسیرها و فایل ها

یک مسیر، عبارتی استاندارد برای مشخص کردن محل ذخیره ی یک فایل است.

D:\Projects\

: ثابت مسیر. با استفاده از این دستور می توان مسیر ثابتی را مشخص کرد.

Path



Path



: نشان دهنده ی مسیر. این دستور مسیر تولید شده توسط دستورات دیگر را نشان می دهد.

File Dialog



: با اجرای این دستور مسیری برای ذخیره ی یک فایل از کاربر پرسیده می شود.

Build Path



: این دستور یک مسیر را با یک رشته ترکیب و مسیر جدیدی تولید می کند. این رشته نام فایل را

تعیین می کند.

Write to Text File



: داده های رشته ای را در یک فایل متنی می نویسد.

Read from Text File



: داده های رشته ای را از یک فایل متنی می خواند.

۴-۲-۶-دستورات کار با پورت سریال کامپیوتر

در فصل قبل ارتباط سریال بررسی شد. نرم افزار Labview شامل دستوراتی برای کار با پورت سریال

کامپیوتر است. این دستورات که کاربردها بسیار ساده است، برای پیکره بندی، خواندن و نوشتن در پورت سریال به کار می روند.

VISA Configure Serial Port



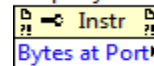
: این دستور برای پیکره بندی پورت سریال به کار می رود. اسم پورت (COMx)، نرخ ارسال، وجود بیت توازن، تعداد بیت پایان و تعداد بیت های داده از جمله مواردی هستند که باید تعیین شوند. حالت پیش فرض، COM1، با نرخ ۹۶۰۰، بدون بیت توازن، ۸ بیت داده و یک بیت توقف است.

VISA resource name VISA resource name



: یک منوی کرکره ای است که اسم پورت سریال را مشخص می کند. اگر پورت سریالی روی سیستم شناخته شود، اسم آن در این منو ظاهر می شود. اسم پورت سریال قبل از اجرای برنامه باید انتخاب شود. اگر برنامه این اسم را نشناسد، خطایی رخ می دهد.

Property Node



: این دستور اسم پورت سریال را دریافت می کند و مشخصاتی از آن را به نمایش می گذارد. یکی از این مشخصات، تعداد بایت روی پورت است. تعیین این مشخصه برای خواندن پورت الزامی است.

VISA Read



: اسم پورت را دریافت می کند و به تعداد بایت موجود در پورت که توسط Property Node مشخص می شود، داده از پورت می خواند.

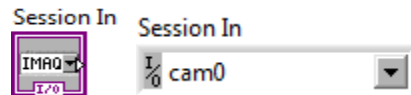
VISA Close



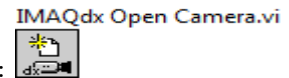
: دستور VISA Configure Serial Port پورت سریال را باز می کند. برای بستن آن از این دستور استفاده می شود.

۴-۲-۷-دستورات تصویر برداری و کار با فایل ویدئویی

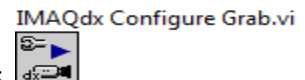
برای استفاده از این دستورات، باید بسته های نرم افزاری بینایی ماشین که به همراه نرم افزار Labview می آیند، به طور کامل به روی سیستم نصب شده باشند. در این صورت، این دستورات در بخش مخصوص خود در جعبه دستورات نمایش داده می شوند و می توان از آنها استفاده کرد. این بسته های نرم افزاری عبارتند از: NI Vision Acquisition Software، NI Vision Builder for Automated Inspection و NI Vision Development Module. در زیر تعدادی از این دستورات به طور مختصر توضیح داده می شوند.



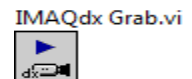
یک منوی کرکره ای برای انتخاب اسم دوربین باز می کند. اگر دوربین با موفقیت روی سیستم نصب شده باشد، توسط نرم افزار اسمی به آن اختصاص داده شده و مورد استفاده قرار می گیرد. برای مشاهده و یا تغییر این اسم می توان از برنامه Measurement & Automation که به همراه Labview نصب می شود، استفاده کرد.



دوربینی که اسم آن به دستور وارد شده است را باز، و آن را برای استفاده کاربر آماده می کند.



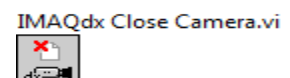
دوربین را برای استفاده از دستور Grab و تصویر برداری سریع آماده و پیکره بندی می کند.



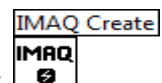
سریع ترین دستور تصویر برداری در Labview است. با استفاده از این دستور می توان به آخرین فریم موجود در بافر دست یافت. برای استفاده از این دستور، حتماً باید قبلاً از دستور IMAQdx Configure Grab استفاده شده باشد.



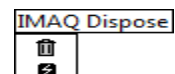
یک پنجره برای نمایش تصویر گرفته شده باز می کند و آن را نمایش می دهد.



دوربین باز شده را می بندد. اگر دوربینی توسط دستور IMAQdx Open Camera باز شود، پس از پایان عملیات باید حتماً با این دستور بسته شود.



یک حافظه ی موقت برای جا دادن اطلاعات تصویر گرفته شده ایجاد می کند. باید یک رشته که اسم تصویر را بیان می کند به عنوان ورودی به دستور وصل شود.



حافظه ی موقت ایجاد شده توسط IMAQ Create را آزاد می کند. اگر پس از اتمام کار از این دستور استفاده نشود، قسمتی از حافظه غیر قابل استفاده باقی می ماند.

IMAQ AVI Create



: یک فایل خالی ویدئویی AVI برای ساخت ویدئو در حافظه ایجاد می کند. مسیر ذخیره فایل باید در این دستور معلوم شود. در این دستور می توان مواردی مانند سرعت پخش فریم در ثانیه، کیفیت ویدئو و... را نیز معلوم کرد.

IMAQ AVI Open



: یک فایل AVI که قبلاً ایجاد شده است را از مسیر مشخص شده باز می کند.

IMAQ AVI Get Info



: از فایل AVI مشخص شده اطلاعاتی را به دست می آورد. یکی از این اطلاعات تعداد فریم موجود در فایل است.

IMAQ AVI Close



: فایل AVI باز شده و یا ایجاد شده را می بندد. مسیر فایل باید مشخص باشد.

IMAQ AVI Write Frame



: در فایل AVI باز شده یا ایجاد شده یک فریم از تصویر را می نویسد.

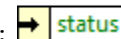
IMAQ AVI Read Frame



: از فایل AVI ایجاد شده یا باز شده یک فریم می خواند. علاوه بر مسیر فایل AVI و مرجع تصویر، باید تعداد فریم موجود در فایل هم معلوم باشد.

۴-۲-۸- سایر دستورات

Unbundle By Name



: برای دسترسی به زیر مجموعه ای از یک کلاستر به کار می رود. کلاستر ساختاری از داده هاست و می تواند شامل انواع داده (آرایه، رشته، متغیر بولی و...) باشد.

Simple Error Handler.vi



: کلاستر خطا را بررسی می کند و اطلاعاتی مانند کد خطاهای موجود را می دهد.

Two Button Dialog



: با اجرای این دستور، یک پنجره ی پیغام باز می شود و بر حسب این که بر روی دکمه ی تأیید یا رد کلیک شود، یک مقدار بولی را بر می گرداند.