

Programmer's Lab

August 8 2011

[illegible]

**Copyright
(C) 2011
Nicole Miller**

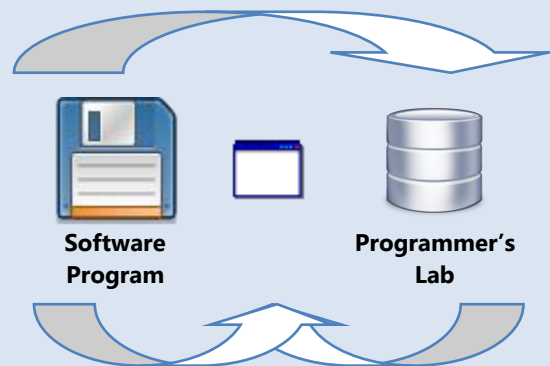
Programmer's Lab User Manual

Introduction

Welcome and thank you for downloading the Programmer's Lab. The Programmer's Lab presents a compiled collection of over 500 functions that empower limited programming languages and simplify complex code at the same time. It accomplishes these two seemingly different things by acting like a [dynamic link library](#) (DLL).

Like a dynamic link library, the Programmer's Lab receives a request for a specific function from a software program. This software program communicates with the Programmer's Lab through the common "shell," "execute," or "run" command found in popular programming languages like Visual Basic, WSH (Windows Script Host), and even the DOS batch file.

A software program requests a function from the Programmer's Lab.



The Programmer's Lab sends the result of that function to a format the software program can use and manipulate.

The Programmer's Lab then uses common programming functions (date, time, file, folder, math, system, media, networking, and string commands) to send the results of those functions to the software program that requested it. Results are stored in a textbox, a small file, or the end-user's clipboard for the taking.

Functions and Features

Functions

The Programmer's Lab provides over 500 functions in nine distinct areas:

Date – From adding dates to grabbing the current year.

File – From opening a specific Excel worksheet to grabbing the current version of Word.

Math – From getting a number's absolute value to grabbing a two-tailed p-value of a z-test.

Media - From getting the blue component of a color to grabbing a window's current state.

Networking - From disconnecting a network drive to uploading a file to an FTP server.

String - From working with ASCII code to stripping all white space from a string of text.

System - From activating a window to grabbing the current/active working directory.

All of these functions (and even more in-between) will help you transform an elementary development program into a very sophisticated piece of work.

Features

None of the functions above would be very helpful if they weren't easy to use. That's why the Programmer's Lab was designed to simplify what can seem like programming chaos. The Programmer's Lab:

- Reduces anywhere from 5 to 20 lines of code to just one!
- Comes with an easy-to-use interface for safe testing and experimentation.
- Takes up under 1MB of disk space. (That's tiny!)
- Automatically updates without any effort on your part.
- Makes the complex simple and the inconceivable believable.
- Takes advantage of Microsoft products like Excel, Access, and Word.
- Costs absolutely nothing to use or distribute. The Programmer's Lab is 100% free!

If you've been looking for an easier way to code, and a free way to boot, you've downloaded the right software. Read on to get started.

Table of Contents

Introduction	2
Functions and Features.....	2
Functions.....	2
Features	3
File List	6
PLAB.EXE	6
PL.EXE.....	6
Important!.....	6
The Interface	7
Table of Functions.....	7
A-Z Index	9
Demo & Test Troughs	10
Finding Functions to Work With	13
The Functions.....	14
Function List.....	14
Shell, Execute, and Run.....	27
Shelling Out via Visual Basic.....	28
Shelling Out via WSH (Windows Script Host).....	28
Shelling Out via MS-DOS Batch.....	28
Shelling Out via [___]	29
Special Notes.....	30
Parameters.....	30
Double Quotes	30
Microsoft Office	30
Branding.....	30
Payoneer Interface.....	30
Program Information	32
Technical Support	32
Distribution Policy.....	32
Disclaimer.....	32

File List

Before we get started, it's important to understand the difference in the provided files. The Programmer's Lab contains two important files.

PLAB.EXE

The first file, PLAB.EXE, is the Programmer's Laboratory. This file gives you the platform in which to properly experiment and test the functions inside the second file, PL.EXE. The Programmer's Laboratory will not work without the PL.EXE file being in the same folder, so make sure both files exist in the same directory.

PL.EXE

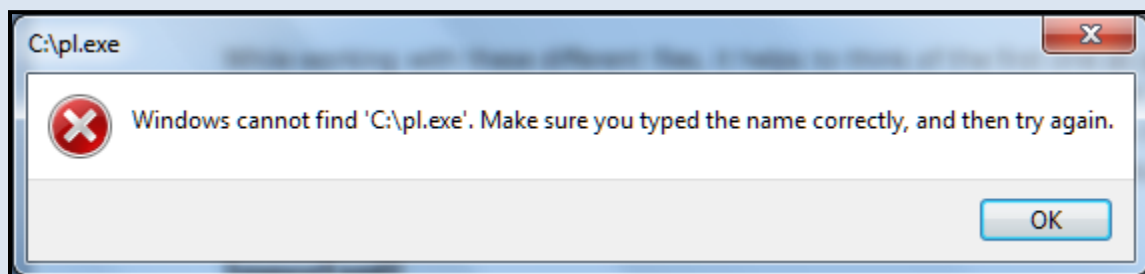
The second file, PL.EXE, stores all 500+ functions that both the laboratory and software development projects use for various purposes.

While working with these different files, it helps to think of the first one as your *personal scientific laboratory*, and the second one as your *chemistry kit*. You can even think of your software development project as your *experiment*. With these two files, you can use the various functions available in the chemistry kit (PL.EXE) to strengthen your project (or your experiment) while working in the laboratory (PLAB.EXE).

Important!

As you'll see in the examples below, software projects that use functions stored in the PL.EXE file require its access every time they're used. The PL.EXE file must, therefore, be distributed alongside finished software products so that end-users can enjoy a function-rich computing experience.

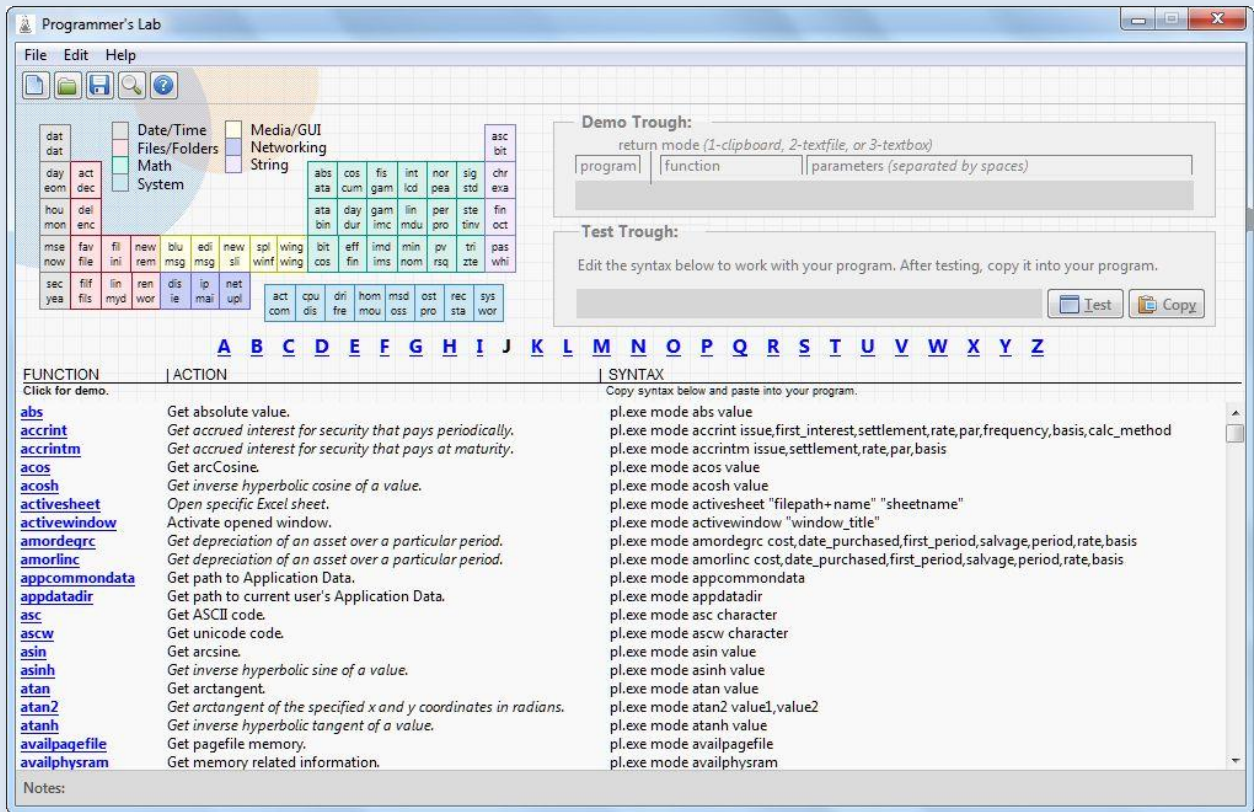
Failing to distribute the PL.EXE file with the products that use its functions will show your end-users an error message like the one below.



Since you don't want that to happen, always remember to include the PL.EXE file with every project that uses its functions!

The Interface

It's much easier to work with the Programmer's Lab through the laboratory interface (PLAB.EXE) than with the PL.EXE file alone. Here, the laboratory is explained below in all its wonderful glory.



The Programmer's Lab Interface

There are three main parts of the Programmer's Lab interface: (1) The Table of Functions, (2) The Demo & Test Troughs, and (3) the A-Z Index.

Table of Functions

[illegible]

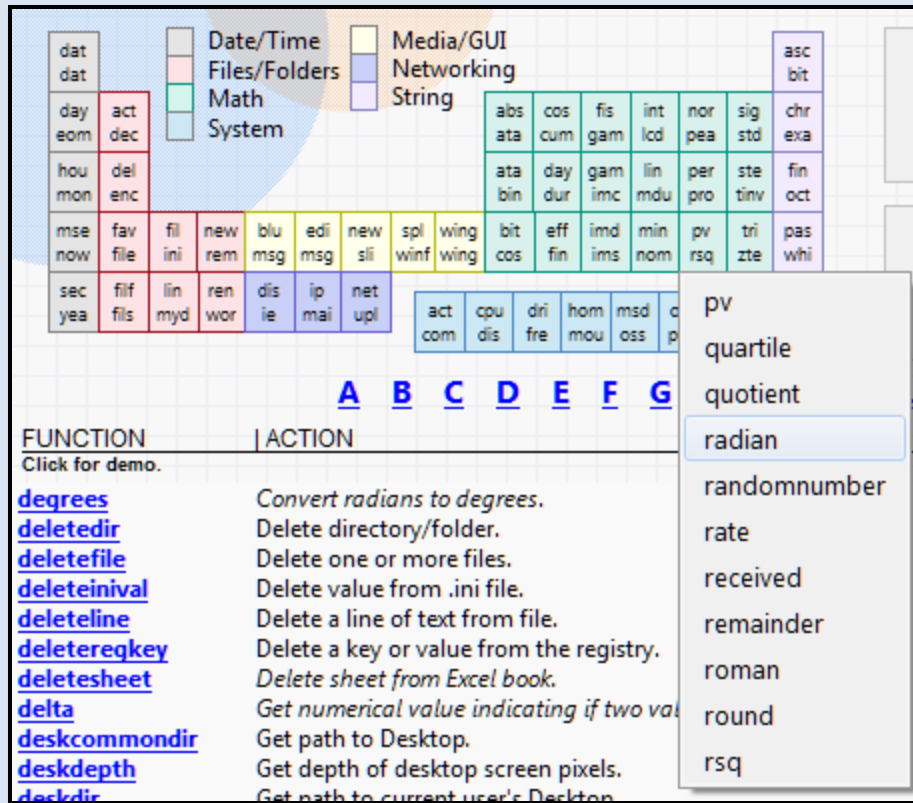
Table of Functions

The **Table of Functions** is organized much like the Periodic Table of Elements. Each function is organized into its own family: (date, time, file, folder, math, system, media, networking, and string commands).

The **ISOWEekNUMBER** Function in the Table of Elements

Clicking any one of the 'elements' will present a menu of functions available within an alphabetically ordered container. If, for example, you wanted your software to get the week number of a given date, you'd click the **hou-mon** element of the **Date/Time** family to locate the *isoweeknumber* function.

If you wanted your software to convert a degree to a radian, you'd click the **pv-rsq** element of the **Math** family to locate the *radian* function. Every function in the PL.EXE file is organized like this in the **Table of Functions**.

The *radian* Function in the Table of Elements

Functions that work with days, weeks, months, hour, minutes, and seconds are alphabetically arranged within the **Date/Time** family. Functions that work with the desktop, memory, the mouse, and hardware are alphabetically arranged within the **System** family. And so on and so forth. Each family is color coded, so locating a function is easy when you know its name and you know what family it belongs to.

After you select a function, the bottom part of the screen puts that function at the top of the **A-Z Index**.

A-Z Index

The **A-Z Index** lists every available function in alphabetical order. You can jump to a set of functions by clicking one of the letters at the top of the index. Clicking *P*, for example, will place the first function starting with the letter *P* at the top of the index. Clicking *H* will place the first function starting with the letter *H* at the top of the index. The A-Z letter set at the top of the index provides a convenient way to quickly locate functions by name.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
FUNCTION	ACTION																									
Click for demo.	SYNTAX																									
abs	Get absolute value.																									
accrint	Get accrued interest for security that pays periodically.																									
accrintm	Get accrued interest for security that pays at maturity.																									
acos	Get arcCosine.																									
acosh	Get inverse hyperbolic cosine of a value.																									
activesheet	Open specific Excel sheet.																									
activewindow	Activate opened window.																									
amordegrc	Get depreciation of an asset over a particular period.																									
amortinc	Get depreciation of an asset over a particular period.																									
appcommondata	Get path to Application Data.																									
appdatadir	Get path to current user's Application Data.																									
asc	Get ASCII code.																									
ascw	Get unicode code.																									
asin	Get arcsine.																									
asinh	Get inverse hyperbolic sine of a value.																									
atan	Get arctangent.																									
atan2	Get arctangent of the specified x and y coordinates in radians.																									
atanh	Get inverse hyperbolic tangent of a value.																									
availpagefile	Get pagefile memory.																									
availphysram	Get memory related information.																									

The A-Z Index

The first column of the index (the **Function** column) stores functions by name. The second column (the **Action** column) describes what each function can do, and the third column (the **Syntax** column) illustrates how these functions must be coded into your software program.

Clicking a function in the first column places its syntax into the **Demo and Test Troughs** for illustration and experimentation.

Demo & Test Troughs

The **Demo & Test Troughs** are where you'll spend most of your time tweaking a selected function to perfection.

Demo Trough:

return mode (1-clipboard, 2-textfile, or 3-textbox)

program	function	parameters (separated by spaces)
pl.exe	3 inputbox	"My Input Box" "Good" "Day" ""

Test Trough:

Edit the syntax below to work with your program. After testing, copy it into your program.

pl.exe 3 inputbox "Enter Password" "It's required!" "" ""	Test	Copy
---	------	------

The Demo & Test Troughs

Demo Trough

In the illustration above, you can see the syntax of the *inputbox* function placed into each trough. All that stuff is automatically placed into the **Demo Trough** whenever you select a function within the **A-Z Index**. The **Demo Trough** splits the function's syntax into its necessary parts. It exists to simply demonstrate how every function must be formatted before it can work with the pl.exe file.

Every function must contain 3 - 4 bits of information:

- **Program Name** - The name of the function container (pl.exe).
- **Mode** - A number between 0 and 4 to indicate how data will be returned.*
- **Function Name** - The name of the function you want to run.
- **Parameters** - Any extra options if necessary.

**(The pl.exe file will return data through the clipboard, in a file, or in a textbox. Your software can grab it from any one of these places.)*

Returning to the illustration above, you can see "pl.exe" as the program, "3" as the mode, "inputbox" as the function name, and "My Input Box" "Good" "Day" and "" as the function's parameters. The *inputbox* function requires each of these 'parts' to function correctly, even though the actual content of these 'parts' may differ according to your needs.

Test Trough

Here's a working example to put things into perspective. Let's say you wanted your software to display an input box to your end-users. If you clicked the *inputbox* function on the **A-Z Index**, the Programmer's Lab would automatically place the following text into the **Demo Trough** and the **Test Trough**:

```
PL.EXE 3 INPUTBOX "MY INPUT BOX" "GOOD" "DAY" ""
```

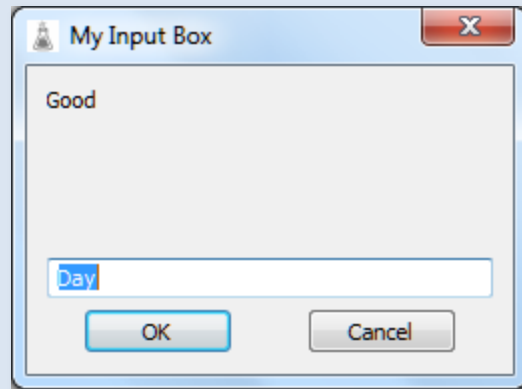
If you looked in the Syntax column of the A-Z Index, you'd see that the Programmer's Lab expects the *inputbox* function to follow this format:

```
PL.EXE MODE INPUTBOX "TITLE" "PROMPT" "DEFAULT" PASSWORD_CHARACTER
```

Taking a look at the **Demo Trough** again, you'd see the same information, only with slightly different parameters:

The name of the program required to display an input box is "pl.exe." (That part never changes.) The mode is "3," which tells pl.exe to display whatever the user types in the input box into a text box. "Good" is the name of the input box's window. "Day" is the input box's prompt text, and both the default and password_character is purposely left blank (as "") so that the input box won't prompt the user for a password.

If you clicked the *Test* button, or pressed CTRL + T, you'd see the following.

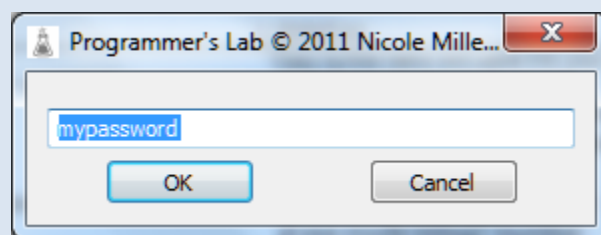


Let's see what would happen if you changed this generic input box to a fancy password request. If you (1) clicked down into the **Test Trough** and changed the text "My Input Box" to something more descriptive, like "Enter Password," then (2) replaced "Good" with "Your password is required each time you use Program XYW". Your password is assigned to you when you first receive your login information...", (3) left the default text blank, and finally (4) placed an asterisk into the last set of quotes, you'd get the following after clicking the *Test* button:



If someone entered his or her password into this box, as illustrated above, they'd only see a set of stars. After the OK button was pressed, the PLEXE file would send that password to a small textbox since we used mode 3. (If we used mode 2, that password would have been saved to a small file named, "tmp.tmp." If we used mode 1, the password would have been sent straight to the user's clipboard.)

The end-user would see his or her entered password above as asterisks, but the PLEXE file would send that password to a textbox in plain text, illustrated below:



Other functions don't require as many parameters as the inputbox function requires. Some are really short, and only need 3 bits of information to run.

The PLEXE MODE SAVEFILE function, for example, will show your users a File Save Dialog box, and then send the name of the saved file (as plain text) to a small textbox, the "tmp.tmp" file, or the user's clipboard – depending upon the mode.

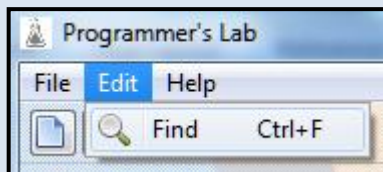
Ensuring Functions Work

The Programmer's Lab isn't set up to simply spit out some usable code. The code that you see in the lab's **Demo and Testing Trough**, in fact, is demonstrative only, and if you use it the way it is, it won't work for your program. You'll need to tweak every function to work the way that you want it to work. That's what programming is all about!

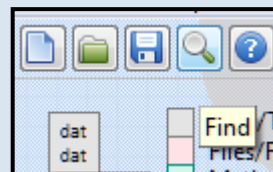
So every time you locate a function that does something similar to what you want it to do, change its parameters in the Testing Trough and then click the Test button (or press CTRL + T) to see if it returns the correct data in the correct format. Only after you're 100% proof positive that your testing is complete and satisfactory, should you copy the function you worked on and paste it into the shell, execute, or run command of your software project.

Finding Functions to Work With

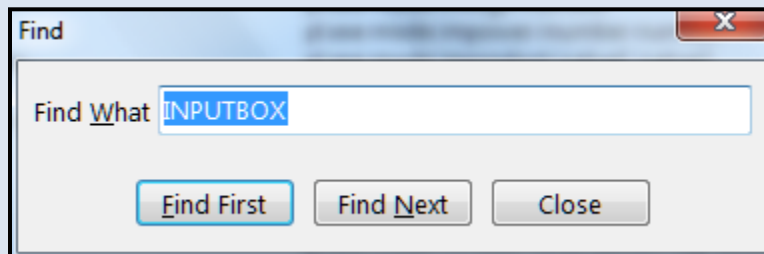
Looking for functions through the **Table of Functions** or the **A-Z index** is pretty straight-forward. But there's an easier way. You can also search for a function with the *Find menu item* or *button*. You can additionally press CTRL + F.



Find Menu Item



Find Toolbar Button

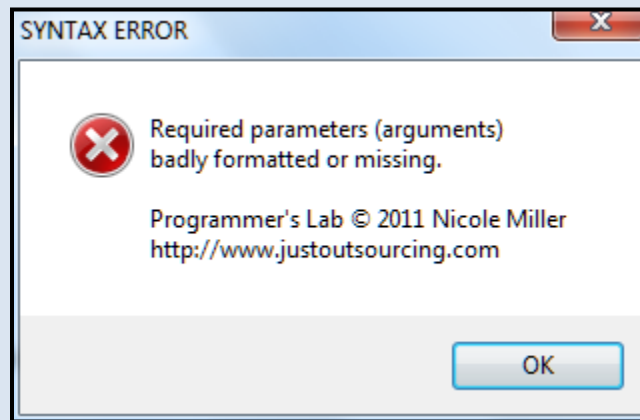


Find Dialog Box

Simply enter the name of the function you want to find (or something similar to it) and press the *Find First* button. The Programmer's Lab will try to locate the appropriate function. Continue searching with the *Find Next* button for more.

The Functions

All of the functions we've discussed so far are compiled into the PL.EXE file, but this file isn't a standalone program. Not quite, that is. The PL.EXE file will do 'something' only when it receives a command with the required parameters (mode, function name, and additional options). Opening this file by itself won't do anything special other than display a Syntax Error message.



That message is displayed every time the PL.EXE program receives a function with badly formatted or missing parameters. Prevent your users from seeing this message by ensuring your functions are formatted correctly!

How can you do that?

Simple.

Always, always, always test your functions in the Programmer's Lab before copying and pasting them into your software project! If a function doesn't work correctly with the lab, it won't work correctly with your software program.

Function List

Here's a list of all the functions available inside the PL.EXE program:

Function Name	Corresponding Action
abs	Gets absolute value.
accrint	Gets accrued interest for security that pays periodically.
accrintm	Gets accrued interest for security that pays at maturity.
acos	Gets arcCosine.
acosh	Gets inverse hyperbolic cosine of a value.
activesheet	Opens specific Excel sheet.
activewindow	Activates opened window.
addfracs	Add two fractions.

amordegrc	Gets depreciation of an asset over a particular period.
amorlinc	Gets depreciation of an asset over a particular period.
appcommondata	Gets path to Application Data.
appdatadir	Gets path to current user's Application Data.
arearect	Find area of rectangle.
areasq	Find area of square.
areatri	Find area of triangle.
asc	Gets ASCII code.
ascw	Gets unicode code.
asin	Gets arcsine.
asinh	Gets inverse hyperbolic sine of a value.
atan	Gets arctangent.
atan2	Gets arctangent of the specified x and y coordinates in radians.
atanh	Gets inverse hyperbolic tangent of a value.
availpagefile	Gets pagefile memory.
availphysram	Gets memory related information.
availvirtual	Gets memory related information.
avedev	Gets average deviation in list of values.
average	Gets arithmetic mean of list values.
bahttext	Gets value converted to Thai text.
barchart	Display a Google bar chart.
beep	Plays a beep.
besseli	Gets modified bessel function for imaginary arguments.
besselj	Gets modified bessel function represented by J(x).
besselk	Gets modified bessel function represented by K(x).
bessely	Gets modified bessel function represented by Y(x).
betadist	Gets cumulative beta probability density function.
betainv	Gets the inverse of cumulative beta probability density.
bin2dec	Gets value converted from binary to decimal.
bin2hex	Gets value converted from binary to hexadecimal.
bin2oct	Gets value converted from binary to octal.
binarystring	Convert string into binary data.
binbytes	Gets bytes from a binary variant.
binexp	Gets binary representation of an expression.
binlen	Gets bytes in binary variant.
binomdist	Gets the individual binomial distribution probability.
bins	Converts a binary variant to a string.
bitand	Gets value of parameters bitwise-AND'ed together.
bitnot	Gets a bitwise NOT operation.
bitor	Performs a BitORoperation.
bitrotate	Performs a BitRotate operation.

bitshift	Performs a BitShift operation.
bitxor	Performs a BitXOR operation.
blue	Gets blue component of a color.
cdtray	Opens or closees the CD tray.
ceiling	Gets value rounded up to next integer.
celldata	Writes data to cell in Excel sheet.
cellvalue	Reads cell from active Excel sheet.
char	Gets character with the corresponding ANSI value.
chart	Draws a small chart.
checkchoice	Shows 3 checkboxes.
chidist	Returns probability of chi-squared distribution.
chiinv	Gets the inverse probability of chi-squared distribution.
chitest	Returns value from chi-squared distribution.
chr	Gets character corresponding to ASCII code.
chrw	Gets character corresponding to unicode code.
clip	Gets text from clipboard.
code	Gets ANSI value for the first character in a text string.
colorbox	Gets selected color.
combin	Gets value of combinations for a given value of items.
combobox	Gets combo box choice.
commission	Get commission paid on sale.
commdir	Gets path to common files folder.
comparestrings	Compares two strings with options.
complex	Gets complex value given real & imaginary coefficient.
concatenate	Gets string that is concatenation of several strings.
confidence	Returns confidence for population mean.
convert	Gets equivalent value converted to measurement system.
copydir	Copies directory files to another directory.
copyfile	Copies one or more files.
copyfiletoclip	Copies file to clipboard.
correlation	Gets the correlation coefficient between two data sets.
cos	Gets cosine.
cosh	Gets hyperbolic cosine of a value.
coupdaybs	Gets value of days from start of coupon to settlement.
coupdays	Gets value of days in the coupon period to the next.
coupdaysnc	Gets value of days from settlement to next coupon.
coupncd	Gets date of the next coupon after the settlement date.
coupnum	Gets coupons paid between settlement maturity date.
coupncd	Gets date of previous coupon before settlement date.
covariance	Returns covariance of two lists of values.
cpuarch	Gets x86 or x64 architecture of pc.

critbinom	Returns smallest value greater than criterion.
ctrltext	Gets text from a control.
cumipmt	Gets cumulative interest paid between two dates.
cumprinc	Gets cumulative principal paid between two dates.
dateadd	Adds date to existing date for a new date.
datedif	Gets value of units between two dates.
datediff	Gets the difference between 2 dates.
datetoday	Gets weekdaynumber for given date.
datetodayv	Gets daynumber for given Gregorian date.
datetomonth	Gets name of month.
datevalue	Gets date serial value represented in a text format.
datewindow	Gets selected date.
day	Gets day as an integer given a date serial value.
dayofdate	Gets name of weekday.
dayofyear	Gets current day of year.
days360	Gets value of days between two dates.
daysinmonth	Gets value of days in a month.
db	Gets amount asset depreciates using declining-balance.
ddb	Gets amount asset depreciates with double-declining balance.
dec	Gets numeric representation of hexadecimal string.
dec2oct	Gets value converted from decimal to octal.
decryptfile	Decrypts file with password.
decryptstring	Decrypts RC4 based string encryption function.
degree	Converts radians to degrees.
deletedir	Deletes directory/folder.
deletefile	Deletes one or more files.
deleteinival	Deletes value from .ini file.
deleteline	Deletes a line of text from file.
deleteregkey	Deletes a key or value from the registry.
deletesheet	Deletes sheet from Excel book.
delta	Gets numerical value indicating if two values are equal.
deskcommdir	Gets path to Desktop.
deskdepth	Gets depth of desktop screen pixels.
deskdir	Gets path to current user's Desktop.
deskheight	Gets height of desktop.
deskwidth	Gets desktop width of screen.
devsq	Returns sum of squares of data points.
dirchange	Changes the current working directory.
dirsiz	Gets size of directory.
disablectrl	Enables a grayed-out control.
disc	Gets discount rate for a security.

disconnectmap	Disconnects network drive.
discount	Get discount value.
discount rate	Get discount rate.
divfracs	Divide two fractions.
divisibility	Checks if NumA is divisible by NumB.
docdir	Gets path to documents.
dollar	Gets string with the dollar formatting \$0,000.00.
dollarde	Gets dollar fraction expressed as a decimal.
dollarfr	Gets dollar decimal expressed as a fraction.
domainlogon	Gets logon domain.
drivelabel	Gets volume label of a drive.
drivemap	Gets details of mapped drive.
driveserial	Gets serial number of a drive.
drivestatus	Gets status of a drive.
drivetype	Gets drive type.
duration	Gets annual duration that pays interest at regular intervals.
edate	Gets serial value months before or after a date.
editwindow	Shows text in an edit window.
effect	Gets effective interest rate given a nominal interest rate.
emptyclipboard	Empties clipboard.
enablectrl	Enables a grayed-out control.
encryptfile	Encrypts a file with password.
encryptstring	Encrypts a string.
eomonth	Gets value of last day of month before or after date.
erf	Gets error function between lower and upper limit.
erfc	Gets complementary error function between x and infinity.
even	Gets value rounded up to the nearest even integer.
exact	Gets boolean value based on whether two strings match.
exe	Executes an expression.
exp	Gets e to the power of a value.
expodist	Returns exponential distribution.
fact	Gets factorial of a value.
factdouble	Gets double factorial of a value.
favscomm	Gets path to favorites folder.
favsdir	Gets path to current user's Favorites.
fdist	Returns F probability distribution.
fileattr	Gets file's attributes.
filecontent	Reads characters from text file.
filedata	Appends text/data to file.
filedrive	Gets drive of file in path.
fileenc	Gets encoding used in a file.

fileextension	Gets a file's extension.
filefolder	Gets a file's folder.
filename	Gets a file's name.
fileopendialog	Gets a selected file.
fileshortcut	Gets working directory of a shortcut.
filesinfolder	Lists and gets selected files in a folder.
filesize	Gets size of file in bytes.
filestatus	Checks if file or directory exists.
filesystemtype	Gets file system type of a drive.
filetime	Gets time and date information for a file.
fileversion	Gets file version information.
fileversions	Compares two file versions.
find	Gets position of a substring within a larger text string.
finv	Returns inverse of F probability distribution.
fisher	Returns Fisher transformation at x.
fisherinv	Returns inverse of Fisher transformation.
fixed	Gets string of value rounded to a fixed decimal place.
flash	Display a flash file in a window.
floor	Gets value rounded down to the closest integer.
folderstatus	Checks if folder exists.
folderwindow	Gets selected folder.
fontbox	Gets selected font.
forecast	Returns future value along linear trend.
formula	Runs a simple Excel formula and Gets result.
freespace	Gets free disk space.
frequency	Returns times values occurs in array.
ftest	Returns result of an F-test.
ftpfile	Gets file from FTP server.
ftpfilesize	Gets size of file on FTP server.
fullpath	Gets engine's full path.
fv	Gets future value of all the payments of an annuity.
fvschedule	Gets future value of principal after compound interest rates.
gammadist	Returns gamma distribution.
gammainv	Returns inverse of a gamma distribution.
gammain	Returns natural logarithm of G(x).
gcd	Gets greatest common divisor of two or more integers.
geomean	Returns geometric mean of an array of values.
gestep	Gets boolean value if a value is greater than threshold.
green	Gets green component of a color.
harmean	Returns harmonic mean of a data set.
hex	Gets hexadecimal.

hex2bin	Gets value converted from hexadecimal to binary.
hex2dec	Gets value converted from hexadecimal to decimal.
hex2oct	Gets value converted from hexadecimal to octal.
hex2string	Converts hex string to string.
homedrive	Gets drive of user's home directory.
homepath	Gets directory of user's home directory.
hosttoip	Gets IP number of host.
hour	Gets hours in 24-hour format.
htmlbody	Gets HTML inside <body> tag.
htmlsource	Gets full HTML source.
htmltext	Gets text inside <body> tag.
hypgeomdist	Gets hyper geometric distribution.
ie	Loads a website and wait for it to load.
ielinks	Gets links from a webpage.
imabs	Gets modulus of a complex value in $x + yi$ or $x + yj$ format.
imaginary	Gets imaginary coefficient of a complex value.
imargument	Gets argument (theta), an angle expressed in radians.
imconjugate	Gets complex conjugate of a complex value.
imcos	Gets cosine of a complex value in $x + yi$ or $x + yj$ format.
imdiv	Gets quotient of two complex values format.
imexp	Gets exponential of a complex value.
imfrac	Change mixed fraction to improper fraction.
imln	Gets natural logarithm of a complex value.
imlog10	Gets common logarithm (base 10) of a complex value.
imlog2	Gets base-2 logarithm of a complex value.
impower	Gets complex value raised to a power.
improduct	Gets product of two or more complex values.
imreal	Gets real coefficient of a complex value.
imsin	Gets sine of a complex value in $x + yi$ or $x + yj$ format.
imsqrt	Gets square root of a complex value.
imsub	Gets difference of two complex values.
imsum	Gets sum of two or more complex values.
inivalue	Reads value from .ini file.
inputbox	Gets input.
int	Gets integer (whole value)..
intercept	Gets intersection with y-axis.
interest	Get interest with principle and time.
intrate	Gets interest rate for a fully invested security.
ip	Gets public IP address of a network/computer.
ipaddress1	Gets IP address of first network adapter.
ipaddress2	Gets IP address of second network adapter.

ipaddress3	Gets IP address of third network adapter.
ipaddress4	Gets IP address of fourth network adapter.
ipmt	Gets interest paid for a given period of an investment.
iptohost	Change IP value to host name.
irr	Gets interest rate for series of unequal cash flows.
isadmin	Sees if user has administrator privileges.
isleapyear	Checks a year to see if it is a leap year.
isoweekday	Gets ISO weekday number for given date.
isoweeknumber	Gets week number of given date.
ispmt	Gets interest paid during a specific period of an investment.
kblayout	Gets keyboard layout.
kurt	Gets kurtosis for list of values.
large	Gets Kth largest value in a list of values.
lcm	Gets least common multiple of integers.
leftstring	Gets characters from left side of string.
leftstringtrim	Trims characters from left side of string.
linechart	Display a Google line chart.
linecount	Gets number of lines in file.
linedata	Appends line of text to file.
linest	Gets coordinates for a straight line.
linewritedata	Writes text to specific line.
log	Gets natural logarithm of a value.
log10	Gets logarithm of a value to the base 10.
logdata	Writes date, time and text to log file.
logest	Gets array of values for exponential curve.
loginv	Gets inverse of lognormal function of x.
lognormdist	Gets lognormal distribution of x.
longfile	Gets long path+name of file.
lowerstring	Converts string to lowercase.
mail	Opens mail client with address, subject, and body.
mailsilent	Sends email without external email program.
map	Display a Google map (chart).
maxavalue	Gets largest value in a list of arguments.
maxvalue	Gets largest value in a set of values.
mday	Gets current day of month.
mduration	Gets modified annual duration for a security.
meminuse	Gets memory related information.
meterchart	Display a Google meter chart.
middlestring	Gets characters from a string.
min	Gets minutes value of clock.
minavalue	Gets smallest value in a set of values.

minimizeall	Minimizes all windows.
minvalue	Gets smallest value in list.
minverse	Gets inverse matrix of an array.
mirr	Gets modified interest rate for series of unequal cash flows.
mmult	Gets matrix product of two arrays.
mod	Gets modulus operation.
mode	Gets value that occurs most frequently.
mon	Gets current month.
monthcal	Gets selected data from month calendar.
mouseclick	Performs mouse click operation.
mousedrag	Performs mouse click and drag operation.
mousemove	Moves the mouse pointer.
mousepos	Gets current position of mouse cursor.
movedir	Moves directory and sub-directories.
movefile	Moves one or more files.
mround	Gets value rounded to the desired multiple.
msdos	Executes DOS command in window.
msec	Gets milliseconds value of clock.
msgbox	Displays message box.
mulang	Gets code denoting multi language.
multfracs	Multiply two fractions.
multinomial	Gets multinomial of a set of values in a list or cell range.
mydocs	Gets path to My Documents target.
negbinomdist	Gets negative binomial distribution.
netfile	Downloads a file from internet.
netname	Gets Computer's network name.
networkdays	Gets days between two dates w/o weekends and holidays.
newclip	Writes text to clipboard.
newdrivemap	Maps a network drive using current user.
newfile	Creates or zeros out file.
newftpdire	Makes directory on FTP server.
newinisection	Writes section to .ini file.
newinivalue	Writes value to .ini file.
newpath	Makes a new path.
newregvalue	Makes key or value in the registry.
newsheet	Makes a new Excel sheet in existing book.
newshortcut	Creates shortcut (.lnk) to file.
newstringinfile	Replaces string in text file.
newsysdate	Sets date.
newwindowtitle	Changes title of window.
newwordpic	Adds in image to a Word document.

nominal	Gets nominal interest rate given an effective interest rate.
normdist	Gets normal cumulative distribution.
norminv	Gets inverse of normal cumulative distribution.
normsdist	Gets standard normal cumulative distribution.
normsinv	Gets inverse of normal cumulative distribution.
now	Gets current date and time.
nowcalc	Gets date and time in YYYY/MM/DD HH:MM:SS format.
nowcalcdte	Gets date in YYYY/MM/DD format.
nowdate	Gets date in pc format.
nowtime	Gets time in requested format.
nper	Gets value of compounding periods of an annuity.
npv	Gets net present value of a series of cash flows (in and out).
ntfslink	Creates NTFS hardlink to file or directory.
number	Gets numeric representation of expression.
oct2bin	Gets value converted from octal to binary.
oct2dec	Gets value converted from octal to decimal.
oct2hex	Gets value converted from octal to hexadecimal.
odd	Gets value rounded up to the nearest odd integer.
oddfprice	Gets price per \$100 face value of a security.
oddlyield	Gets yield of a security that has an odd last period.
osarch	Gets architecture type of operating system.
osbuild	Gets OS build value.
oslang	Gets OS Language.
osverpack	Gets service pack info.
ostype	Gets operating system type.
osversion	Gets operating system version
passwordgen	Generate 8 character password.
pearson	Gets pearson product correlation coefficient.
perc2dec	Change a percent to a decimal.
percentile	Gets value that corresponds to a percentage.
percentrank	Gets percentage rank of a value in a list.
perimrect	Get perimeter of rectangle.
perimsq	Get perimeter of square.
perimtri	Get perimeter of triangle.
permut	Gets value of permutations from a given value.
picturewindow	Makes a picture control.
piechart	Display a Google pie chart.
ping	Gets ping roundtrip-time.
pixel	Gets a rectangle of pixels for provided color.
pixelchecksum	Gets checksum value of a region.
pixelcolor	Gets x,y pixel color.

pmt	Gets payment for a loan with constant payments and fixed interest.
poisson	Gets poisson distribution.
power	Gets value raised to a given power.
ppmt	Gets fixed payment made in each period of an annuity.
price	Gets price/\$100 face value for security that pays periodic interest.
pricedisc	Gets price per \$100 face value for a discounted security.
pricemat	Gets price/\$100 face value for security that pays at maturity.
printfile	Prints a file.
processclose	Closes an active process.
processexist	Sees if a specified process exists.
product	Gets product of all the values in a list.
programfilesdir	Gets program Files folder.
programscommdir	Gets path to Start Menu folder.
programsdir	Gets path to current user's programs.
properstring	Changes string to proper case.
proportion	Solve a proportion.
pv	Gets present value of an annuity with fixed cash flows.
quartile	Gets quartile of a data set.
quotient	Gets integer portion after division.
radarchart	Display a Google radar chart.
radian	Convert degrees to radians.
radiochoice	Gets choice with radio buttons.
randomnumber	Generates a random value.
rate	Gets interest rate for equal cash flows at regular intervals.
readline	Reads line of text from file.
received	Gets amount received at maturity for a fully invested security.
recycle	Empties the recycle bin.
recyclefile	Sends file or directory to recycle bin.
red	Gets red component of a color.
refreshrate	Gets desktop refresh rate.
regval	Gets name of registry value.
remainder	Gets remainder after division.
renamefile	Renames a file.
renameftpfile	Renames file on FTP server.
renameinisection	Renames section in .ini file.
repeatstring	Repeats a string a value of times.
replacestring	Replaces substring in a string.
replacestringinfile	Replaces string in a file.
reversestring	Reverses a string.
rightstring	Returns right chrs from string.
rightstringtrim	Trims right side of string.

roman	Gets roman numerals for a number.
round	Gets value rounded to value of decimal places.
rsq	Gets square of pearson correlation coefficient.
saleprice	Get the sales price.
salestax	Find sales tax.
savefile	Gets a saved file.
scatterchart	Display a Google scatter plot chart.
screencapture	Captures full screen.
sec	Gets seconds value of clock.
sendctrl	Sends string to a control.
seriessum	Gets sum of a power series based on a formula.
serverlogon	Gets logon server name.
setattrib	Sets the attributes of one or more files.
setctrltext	Sets text of a control.
setfiletime	Sets timestamp of one of more files.
shellexecute	Runs external program.
shortfile	Gets 8.3 short path+name of file.
shutdown	Shuts down system.
sign	Gets numerical value indicating if a value is positive or negative.
sin	Gets sine of a value.
sinh	Gets hyperbolic sine of a value.
skew	Gets value representing skewness of a distribution.
slider	Gets slider value.
sln	Gets amount an asset depreciates using the straight-line method.
slope	Gets slope of the linear regression.
small	Gets kth smallest value in an array of values.
sound	Plays a sound file.
splashimage	Makes image popup window.
plashtexton	Makes text popup window.
sqrt	Gets the square-root of a value.
sqrtpi	Gets square root of a value multiplied by PI.
standardize	Gets value from a mean and deviation.
startmenudir	Gets path to current user's start menu.
startmenufolder	Gets start menu folder.
startupdir	Gets current user's startup folder.
startupfolder	Gets path to startup folder
stdev	Gets standard deviation based on sample.
stdeva	Gets standard deviation based text and values.
stdevp	Gets standard deviation based on population.
stdevpa	Gets deviation based on population, text & values.
steyx	Gets standard error of predicted y-value for each x.

stringbetween	Gets string inside string.
stringinstring	Checks if string contains a substring.
stringlength	Gets number of characters in a string.
subfracs	Subtract two fractions.
sum	Gets sum of a list of numbers.
sumproduct	Gets sum of the product of one or more arrays of values.
sumsq	Gets sum of the squares of a list of values.
sumx2my2	Gets sum of difference of squares of corresponding values.
sumx2py2	Gets sum of sum of squares of corresponding values in two arrays
sumxmy2	Gets sum of squares of differences in two arrays.
syd	Gets amount an asset depreciates sing sum-of-years digits method.
systemdir	Gets path to windows' system.
tan	Gets tangent of a value.
tanh	Gets hyperbolic tangent of a value.
tbilleq	Gets bond-equivalent yield for a treasury bill.
tbillprice	Gets price per \$100 face value for a treasury bill.
tbillyield	Gets yield for a treasury bill.
tdist	Gets probability for a paired T-Test distribution.
tempdir	Gets path to temporary files folder.
text	Gets string of a value with a particular format.
tinvs	Gets t-value of student's t-distribution.
tip	Configure a tip amount.
tooltip	Makes tooltip on the screen.
totalpagefile	Gets memory related information.
totalphysram	Gets total physical ram.
totalspace	Gets total disk space.
totalvirtual	Gets total virtual memory .
trimmean	Gets mean of the interior of a data set.
trunc	Gets value with any decimal places removed.
ttest	Gets probability associated with the f-test.
uploadfile	Uploads file to FTP server.
upperstring	Converts a string to uppercase.
username	Gets ID of currently logged on user.
userprofiledir	Gets path to user's profile folder.
var	Gets compound variance based on a sample.
vara	Gets compound variance based on a sample.
varp	Gets variance based on entire population.
varpa	Gets variance based on entire population w/, text and logic.
vdb	Gets amount an asset depreciates using declining balance methods.
vennchart	Display a Google Venn chart.
video	Display an online video in a window.

volcube	Get the volume of a cube.
volcyl	Get the volume of a cylinder.
volrect	Get the volume of a rectangle.
volsphere	Get the volume of a sphere.
webpage	Display a webpage in a window.
weekdaynumber	Gets numeric day of week.
weibull	Gets weibull distribution.
whitespace	Strips white space in string.
windowclose	Closes an opened window.
windowexist	Checks to see if a specified window exists.
windowflash	Flashes an opened window in the taskbar.
windowkill	Forces an opened window to close.
windowmove	Moves and/or resizes an opened window.
windowpid	Gets an opened window's process ID (PID).
windowstate	Shows, hides, min, max, restores opened window.
windowtext	Gets text from an opened window.
windowtitle	Gets full title from an opened window.
windowtrans	Sets the transparency of opened window.
wingetstate	Gets the state of opened window.
worddoc	Opens existing Word doc file.
wordfindreplace	Finds and replaces text in a Word doc file.
wordmacro	Runs a Word doc macro.
wordversion	Gets version of installed Microsoft Word program.
workingdir	Gets current/active working directory.
xirr	Gets interest rate for unequal cash flows not at regular intervals.
xnpv	Gets net present value for cash flows not at regular intervals.
year	Gets current four-digit year.
yield	Gets interest rate for equal cash flows at regular intervals.
yielddisc	Gets annual yield for a discounted security.
yieldmat	Gets annual yield for a security that pays interest at maturity.
youtube	Display a YouTube video in a window.
ztest	Gets two-tailed p-value of a z-test.

Shell, Execute, and Run

To make these functions work from within your software program, you'll need to find out how your programming environment "shells out," or runs external Windows applications. Most software development programs offer a way to shell out to an external application, but they do so differently. Dictating the 'correct' way here, in this guide, wouldn't be appropriate. So the following examples are offered as suggestive guides. Consult your development program's help file for specific instructions.

Shelling Out via Visual Basic

Visual Basic's shell function looks like this:

```
Shell(commandstring$, windowstyle%)
```

Visual Basic's *commandstring\$* stores the name of the program to execute, including any required arguments or command line switches. If you replace "*commandstring\$*" with a function created with the Programmer's Lab, your program will run the required function after compiled and accessed.

The *windowstyle%* switch determines the style of the window in which the program will be executed. Regardless of which switch is used, the PL.EXE file will always run silently in the background unless it's asked to display a GUI component of some sort (textbox, calendar, chart, etc.).

Example:

```
Sub ShellDemo
    Shell("pl.exe 2 availpagefile", 1)
End Sub
```

That example will shell out to the pl.exe file and store the size of the system's pagefile in a document named 'tmp.tmp.'

Shelling Out via WSH (Windows Script Host)

The Windows Script Host's shell function looks like this:

```
.Run(strCommand)
```

The *strCommand* variable is a string value that indicates a program and set of parameters you want to run. If you replace "*strCommand*" with a function created with the Programmer's Lab, your script will run the required function when double-clicked.

Example:

```
Set WshShell = WScript.CreateObject("WScript.Shell")
WshShell.Run "pl.exe 1 cellvalue C:\PROGRA~4\pl.xls A2"
```

That example will shell out to the PL.EXE file, grab the data inside cell A2 of a file named, "pl.xls," and store that data in the user's clipboard.

Shelling Out via MS-DOS Batch

Shelling out via MS-DOS is a simple matter of copying and pasting a function onto a single, blank line. There are no special commands required.

Example:

```
@ECHO OFF  
pl.exe 3 datewindow "My Date Window" 2011/02/01
```

That example will shell out to the PL.EXE file, display a small calendar with the date 2011/02/01 pre-selected, and then store the user's selected date in a small textbox.

Shelling Out via [____]

Because the PL.EXE file is a commandline-operated engine, it doesn't really matter what type of language calls it. PL.EXE will work with any programming language that can shell out, execute, or run external Windows programs.

Special Notes

Parameters

All parameters are separated by spaces. Parameters do not need to be preceded by forward slashes "/", brackets "[]", or any other non-alphanumeric characters.

Double Quotes

Some functions require double quotes to work, such as parameters with spaces in them. The parameter, Hello World, for example, must be formatted as "Hello World" to render it as a single parameter, and not two separate parameters. Since double-quotes can be difficult to work with in some programming languages, you'll need to figure out how to properly pass them through a shell, execute, or run command.

Microsoft Office

Many of the functions inside the Programmer's Lab require Microsoft Word, Excel, or Access to work. Make sure that when you distribute a program that uses PL.EXE's MS Office functions, your end-users have these programs installed. Microsoft Office functions are italicized in the laboratory's ***A-Z Index***.

Branding

If you'd like your own company or software name and icon branded into the PL.EXE file, as well as appended copyright info removed from all textfile and clipboard output, simply [load our Payoneer account](#) with \$12.95 and the following 2 things:

1. A credit card (MasterCard or Visa)
2. Our personal email address (documaker@aol.com) (*support@justoutsourcing won't work!!*)

That's it!

Payoneer Interface

The screenshot of the Payoneer interface below demonstrates how easy it is to load our account.

Welcome to Payoneer Card Loading Services

Please enter the cardholder email or card number to load with the amount you wish to load, reason for loading and choose your payment method. Once you have provided this information please click on the relevant option at the bottom of the page.

1 Amount to load and reason
2 Checkout
3 Confirmation
Need Assistance?

Select a card to load

Please select the method for identifying the beneficiary:

☒ By Email Address
Enter the Email address that the card holder used for registering with Payoneer:

 Cards held by this beneficiary:

☐ By Payoneer Card Number

Amount to load and reason

Amount to load: USD \$ **200.00**
 Reason for loading (Optional):

 Max allowed: \$ 1,000.00

Payment method

Select payment method:

Who pays the fee?

I will pay the fee
 Your bank account will be debited with an additional \$ 7.79 for fees.
 Beneficiary card will be loaded with:
\$ 200.00

OR

Cardholder pays the fee
 A Fee of \$ 7.50 will be deducted from the amount received by the cardholder.
 cardholder's card will be loaded with:
\$ 192.50

☐ ☒

Continue

When finished, [email us](#) with the following:

- your company or software name
- your company or software icon (48 x 48)

You'll receive a branded copy of the distributable PLEXE file within 72 hours.

Program Information

Technical Support

If you have a question about the Programmer's Lab, e-mail support@justoutsourcing.com. Be prepared to send the following information:

1. The version number of the Programmer's Lab you're using.
2. The operating system you're using.
3. The exact wording of any messages that appeared on your screen.
4. A description of what happened and what you were doing when the problem occurred.
5. A description of how you tried to solve the problem.

You should expect to receive a response within 1-3 business days.

For additional information, send electronic mail to support@justoutsourcing.com or make contact through these other means:

Just Outsourcing
1008 Tenth Street, PMB #456
Sacramento, CA 95814-3502 (USA)

Fax: (253) 595-0700
Email: support@justoutsourcing.com
<http://www.justoutsourcing.com>

Distribution Policy

The Programmer's Lab is absolutely FREE. You may freely distribute this program from your own website, from any number of online file libraries, or from your own CDs. Distribution is permitted without any restriction. Distribution with modification of content, however, is NOT permitted.

Disclaimer

Although every attempt has been made to ensure maximum functionality and compatibility, there may be unforeseen circumstances which will cause the program to fail. Therefore, in using the Programmer's Lab, you agree to accept any and all financial or other loss due to program design, error, or misuse.

Neither the program's author, nor representatives, agents, partners nor other related parties shall be held liable for damages of any type, financial or otherwise.

If you are unable to agree to these terms, the software must not be used and should be removed from your system immediately.