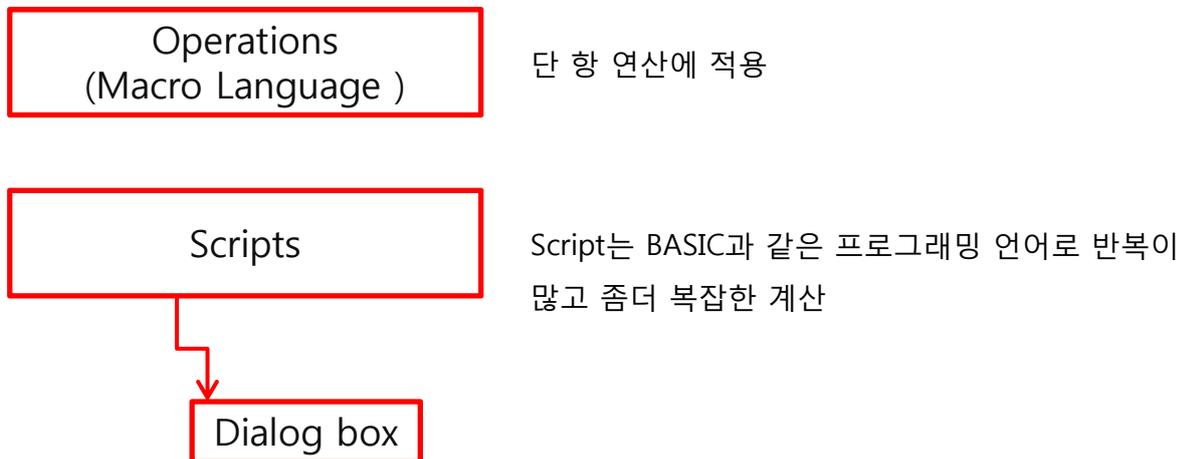


Function

일반적으로 코팅은 지정한 입사각, 물질의 두께 등 하나의 조건 하에서 이루어 지지만 열 방사, 흡수, 두께 또는 입사각 등의 변화에 따른 결과가 필요한 경우 사용자가 이에 대한 조건과 수식을 직접 만들어 빠른 시간 내에 다양하고 정확한 결과를 얻어 내어 업무의 생산성과 효율성을 크게 올릴 수 있는 기능 입니다.

조건과 수식을 위한 방법으로 Operations and Scripts라는 두 가지가 있는데 Operations은 Table 파일에 있는 Data를 다루기 위한 매크로 언어로 매우 쉽고 빠르게 작성, 저장이 가능하며 출력 또한 Table 파일 형식으로 되며 Scripts는 Dialog Box를 만들어 해당 파라미터를 입력 하면 Plot, Table 등 원하는 출력 양식으로 나타낼 수가 있습니다. (상세내용은 웹사이트에 있는 해당 버튼을 클릭하면 됩니다.)

Function 작성은 아래와 같이 2가지 방식이 있습니다.



사용자가 Parameter 입력 방식인 Dialog Box 형태로 만들어 " Script Manger" 라는 곳에 저장, 관리 하여 쉽고 간편하게 원하는 결과를 산출해 낼 수 있습니다.

Dialog box

사용자가 프로그램을 Dialog box 형태로 만들어 쉽고 간편하게 Parameter를 입력, 다양한 결과를 계산해 낼 수 있습니다.

프로그램에 약 40개 정도의 기본 tool이 제공되며 고객이 이외 어떤 Model이 필요한 경우 연락 주시면 제공 가능 여부를 답변 드리겠습니다.

3D system
Admittance
Antireflection Limit
Cauchy
Cauchy Extraction
Cauchy Smoothing
Composite Materials
Contamination Sensitivity
Create CCT Illuminant
Diattenuation
Emittance
Equivalent Parameters
Example 1
Example 2
Example 3
Extended Table
Figure Error
Fringes
Isoreflectance
k to Tint
Make Black Body
Make Gaussian
Metal n and k
OLED Model
Optimal Color Stimuli
Performance with Illuminant
Polarization Coupling
Potential Transmittance
Potential Transmittance II
Psi vs Delta
Reverse Order
Rugate Conversion
Sellmeier
Stack Performance with Illuminant
Stack statistics
Statistics
Tilted admittances
Two-Layer Transformer
Wideband Simulator



현재 기본으로 제공되는 Scripts

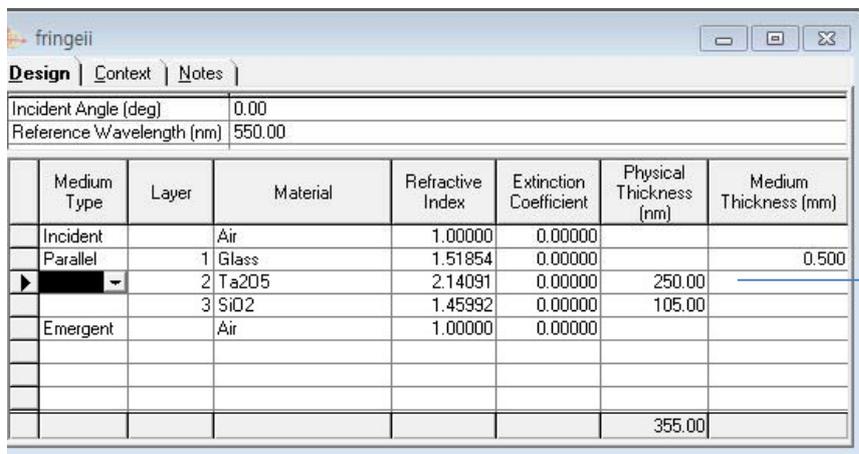
그 중에서 예시를 보여 드리면

- Fringes II

한 Layer의 두께를 단계별로 증감하여 자동으로 계산, 결과 Data를 Table과 Color Patch로 보여 주는 기능.

1. " File > New > Design "

- 해당 Design file을 생성.

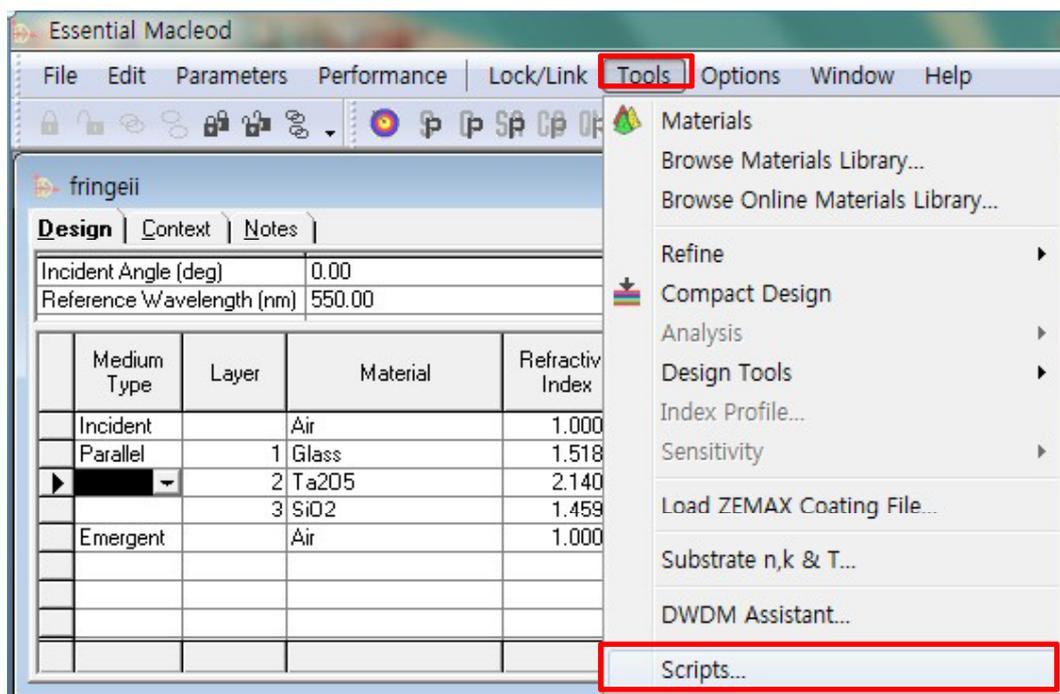


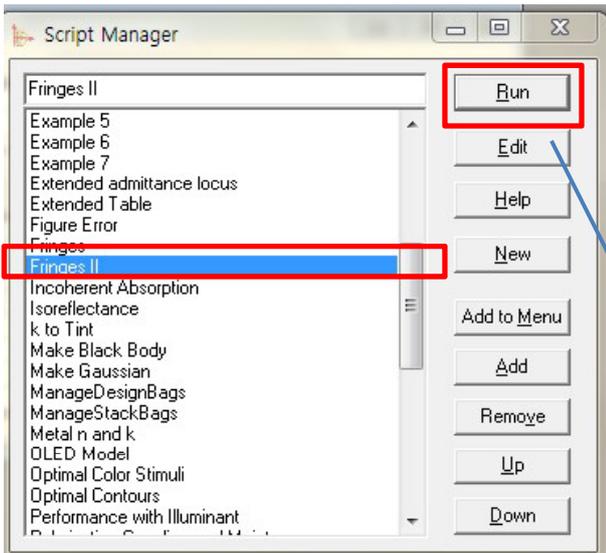
	Medium Type	Layer	Material	Refractive Index	Extinction Coefficient	Physical Thickness (nm)	Medium Thickness (mm)
Incident			Air	1.00000	0.00000		
Parallel		1	Glass	1.51854	0.00000		0.500
		2	Ta2O5	2.14091	0.00000	250.00	
		3	SiO2	1.45992	0.00000	105.00	
Emergent			Air	1.00000	0.00000		
						355.00	

두께 조정 조건

두께 : 250 > 350
Interval : 10

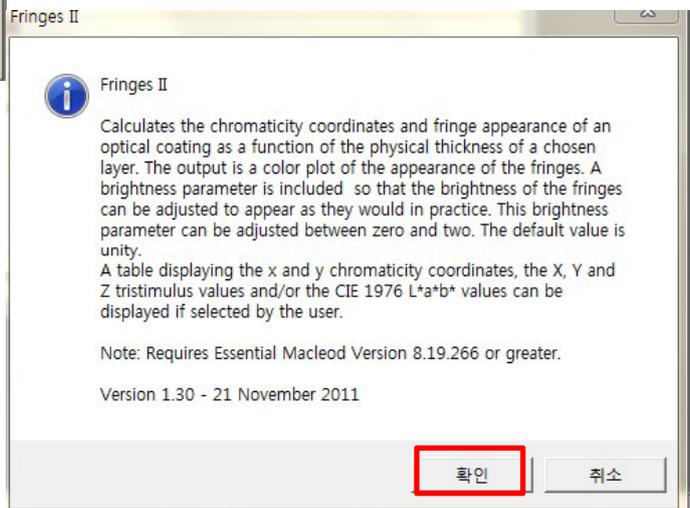
2. " Tools > Scripts "





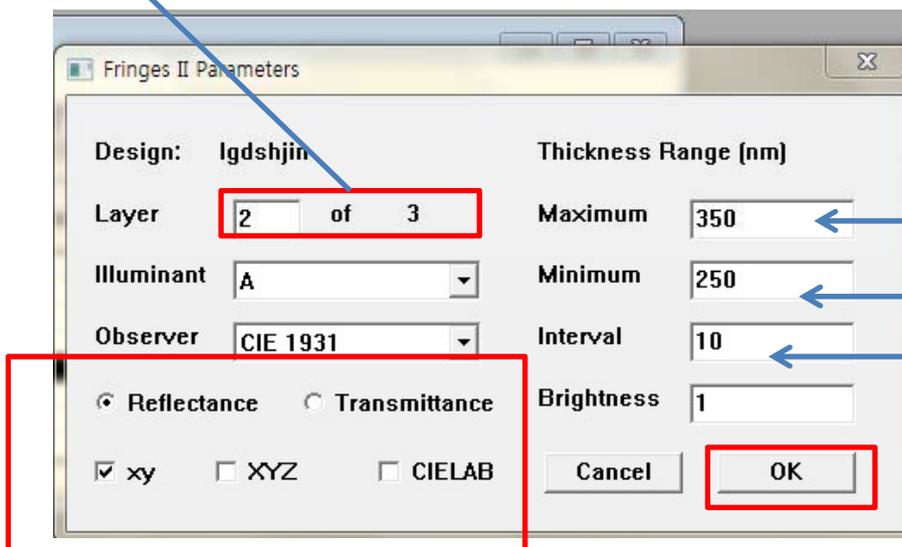
FringeII 선택 > Run

Edit : Scripts file을 보여 주어
사용자가 내용을 보거나 수정이 가능.



3. 결과 Data 조건 설정

3 Layer 중 2번 Layer



최대 두께

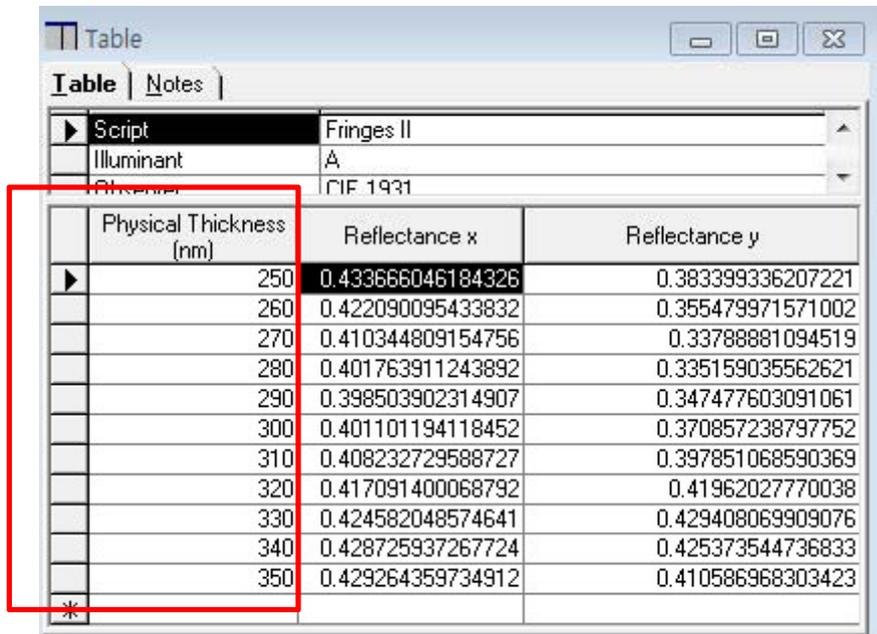
최소 두께

증감 간격

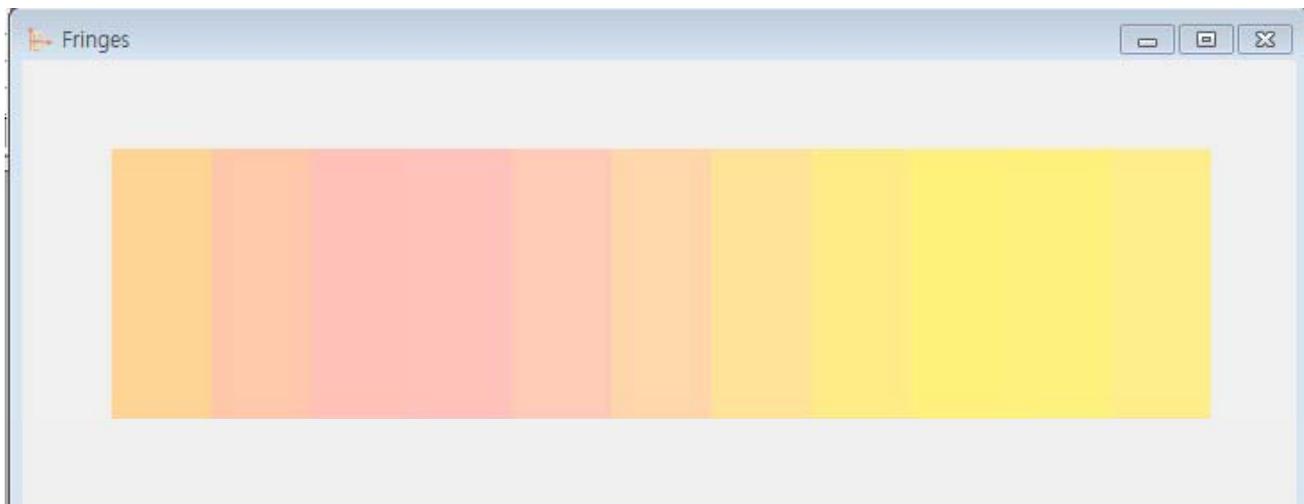
결과 Data 설정



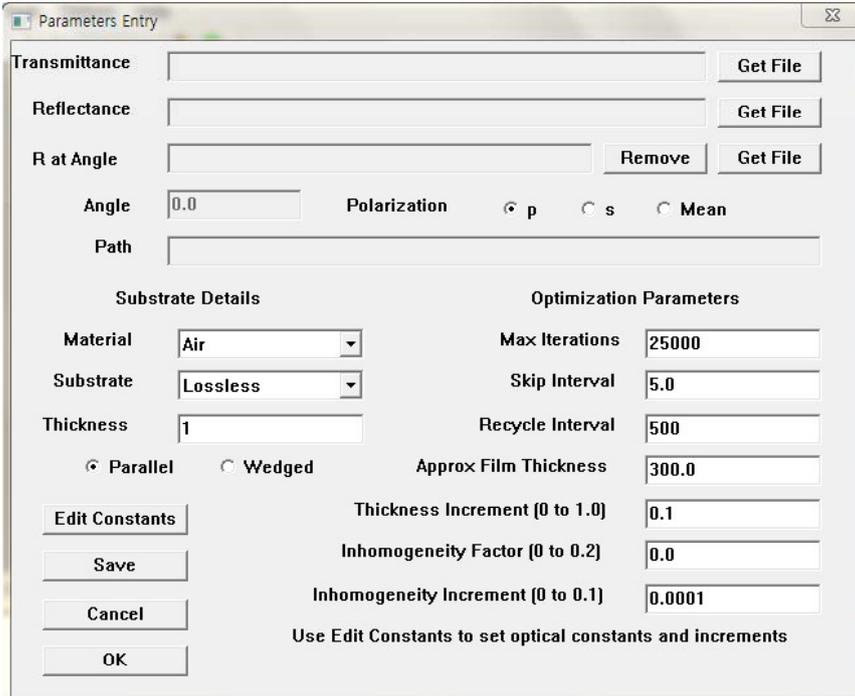
Layer 두께를 단계별로 자동으로 계산한 결과 Data Table과 Color Patch가 나타납니다.



Physical Thickness (nm)	Reflectance x	Reflectance y
250	0.433666046184326	0.383399336207221
260	0.422090095433832	0.355479971571002
270	0.410344809154756	0.33788881094519
280	0.401763911243892	0.335159035562621
290	0.398503902314907	0.347477603091061
300	0.401101194118452	0.370857238797752
310	0.408232729588727	0.397851068590369
320	0.417091400068792	0.41962027770038
330	0.424582048574641	0.429408069909076
340	0.428725937267724	0.425373544736833
350	0.429264359734912	0.410586968303423



- Metal 광학 상수 (Metal n,k) 구하기

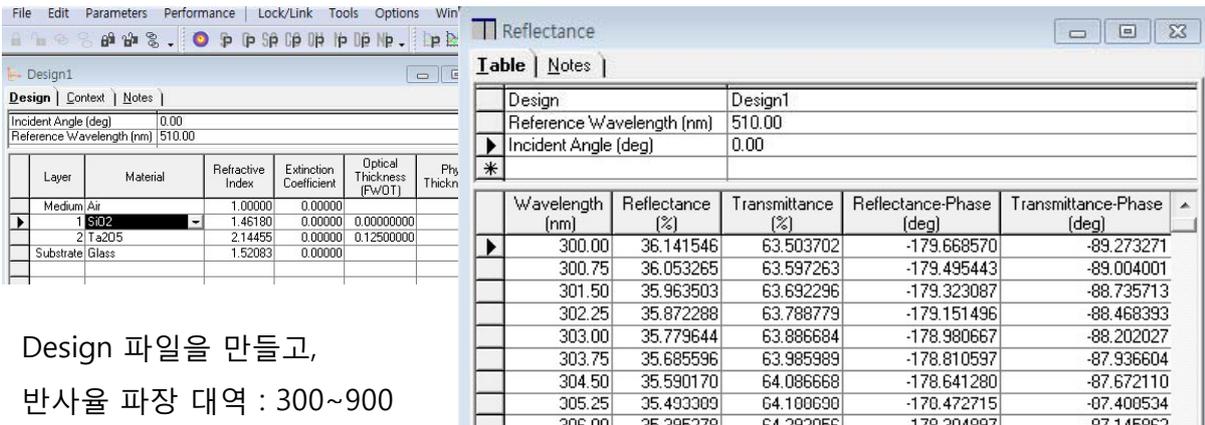


Operations (Macro Language)

단항 연산을 위하여 예를 들어 Illuminant D65 (정규 입사각, 21 레이어, SiO2와 Ta2O5. quarterwave stack)의 스펙트럼 결과 Table을 Macro 언어로 만든다면

$$\text{Output} = \text{D65} \times R/100$$

R 다음에는 0~9 따르는데, 즉 R1. F 다음에는 0 또는1, C는 상수.

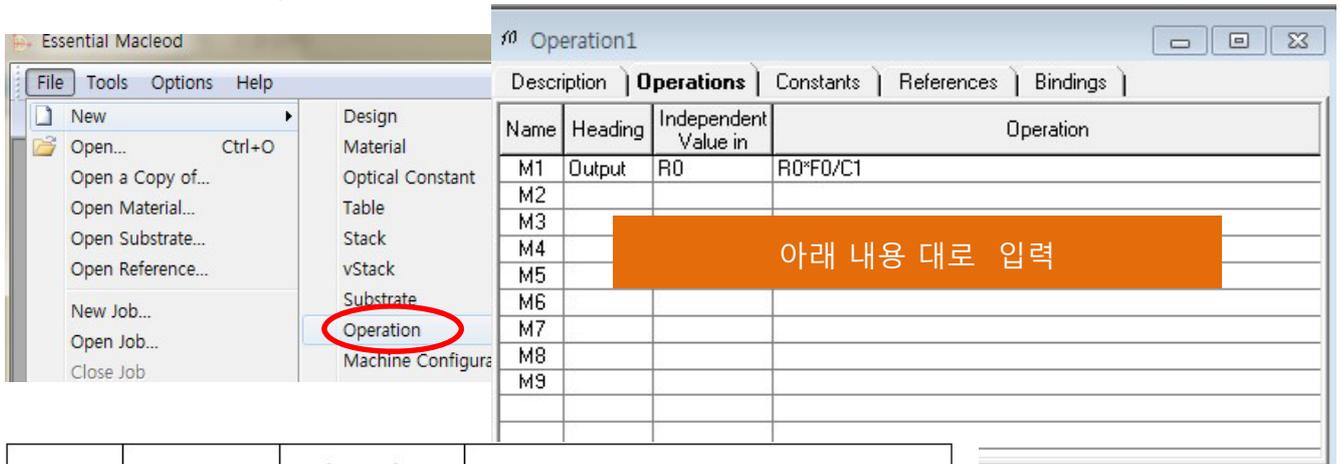


Design 파일을 만들고,
반사율 파장 대역 : 300~900
Interval : 0.75

"Performance > Table "

Table 파일 명을 "Reflectance" 저장

"File > New > Operation "



Name	Heading	Independent Value in	Operation
M1	Output	R0	R0*F0/C1
M2			
M3			
M4			
M5			
M6			
M7			
M8			
M9			

Independent Value : 독립 변수 값의 집합.

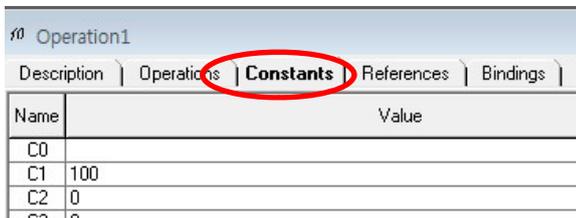
R0 : standard source reference file

F0 : coating response. Scale of 0 to 100%

C1 : 100. Converts from percent to scale of 0 to 1.

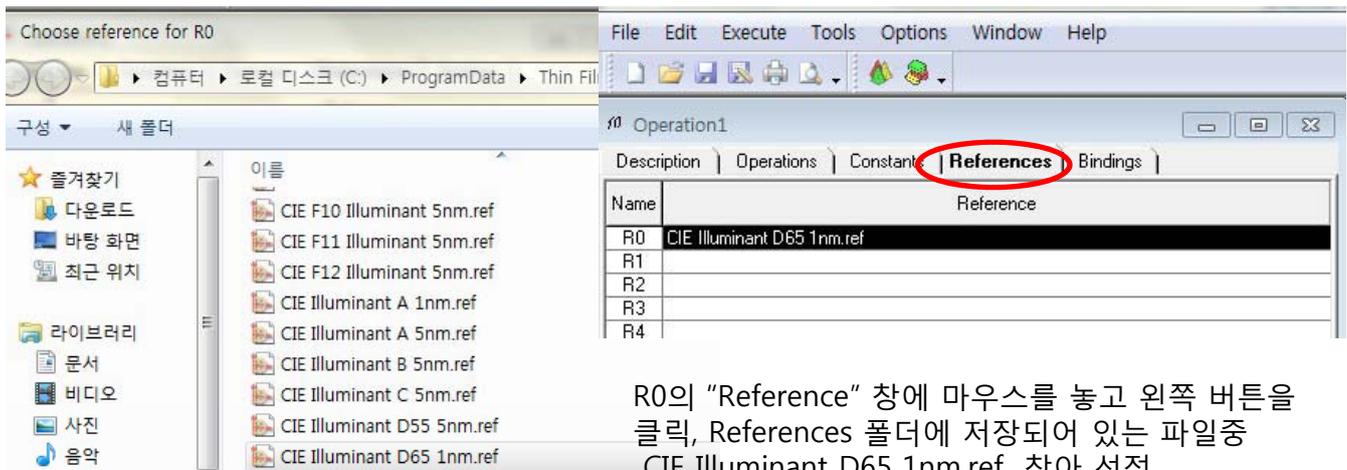
C0 : 특별 상수로 실행 시간을 나타냄.

C1 : 100 입력. 0~1 범위에서 전환된 백분율

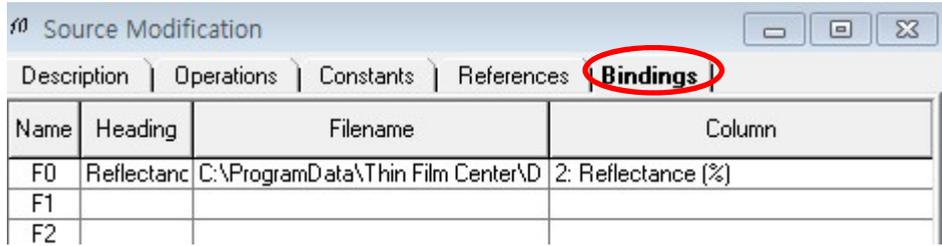


C0 : 특별 상수로 실행 시간을 나타냄.

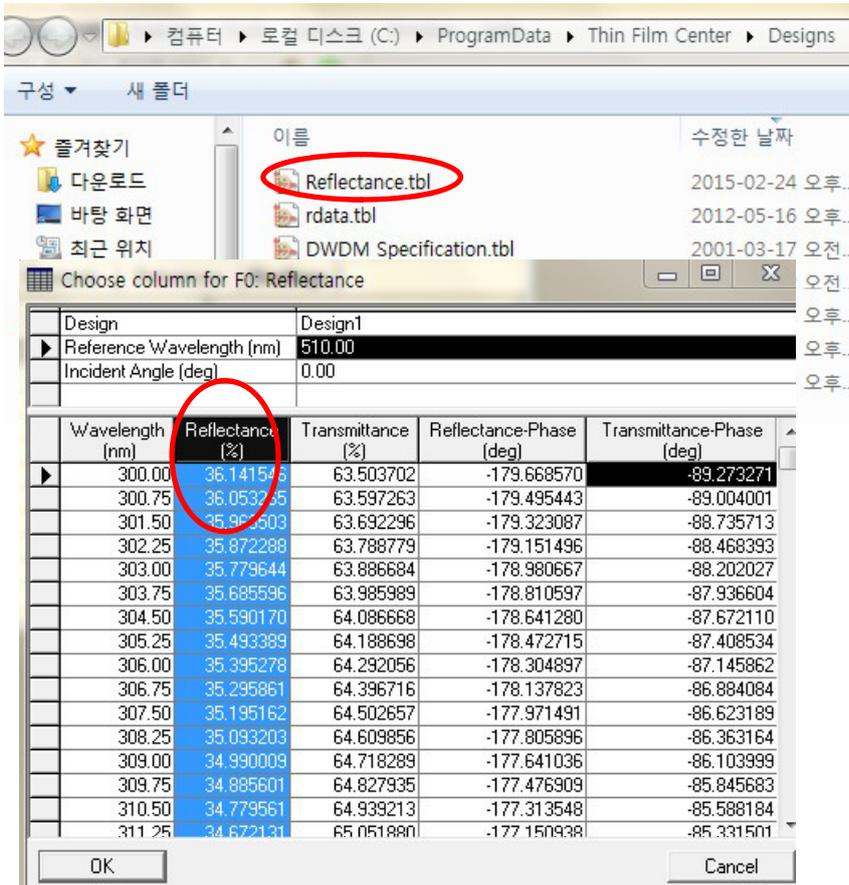
C1 : 100 입력. 0~1 범위에서 전환된 백분율



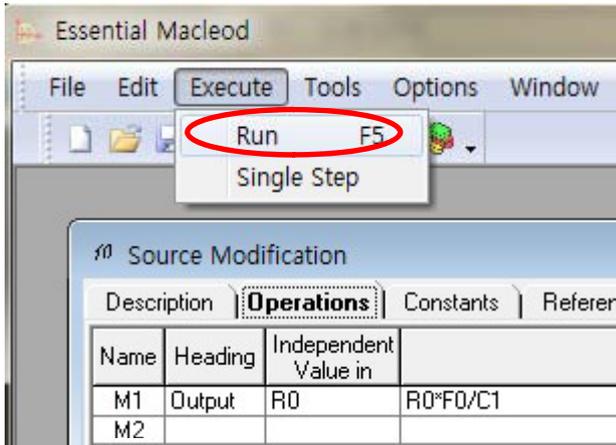
R0의 "Reference" 창에 마우스를 놓고 왼쪽 버튼을 클릭, References 폴더에 저장되어 있는 파일중 CIE Illuminant D65 1nm.ref 찾아 선정.



F0의 Heading에 "Reflectance" "Filename" 창에 마우스를 놓고 왼쪽 버튼을 클릭, Design 폴더에 저장되어 있는 파일 중 "Reflectance.tbl" 찾아 선정.



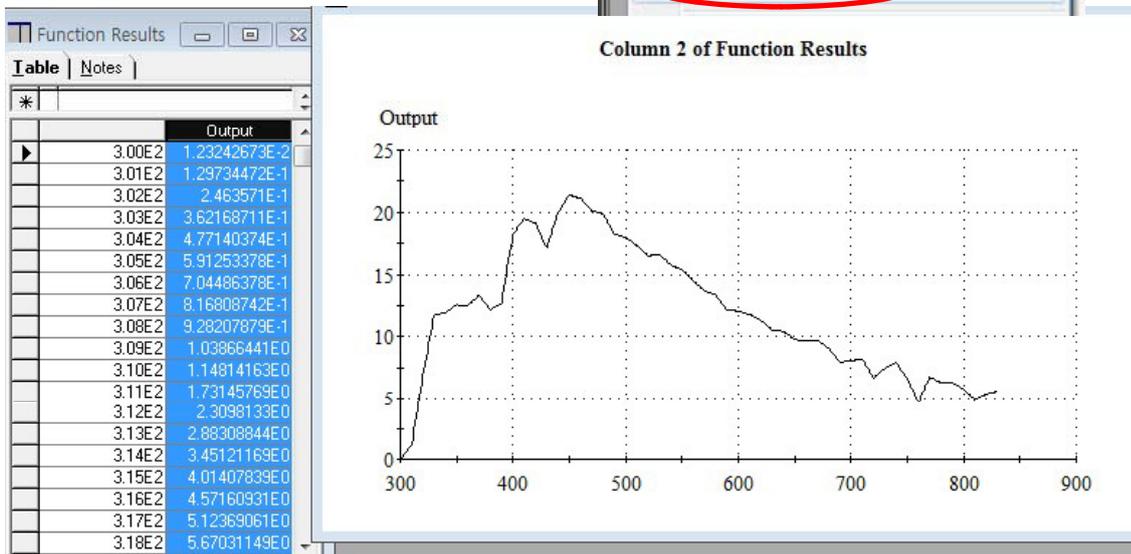
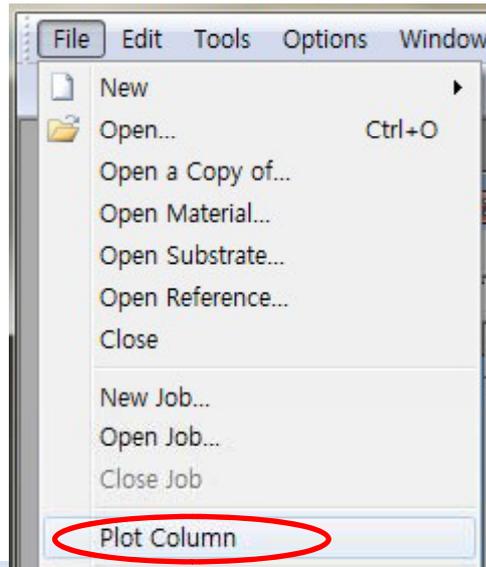
"Column" 창에 마우스를 놓고 왼쪽 버튼을 클릭, 디자인 파일 창이 보이면 "Reflectance(%)" Column 선택 입력. "Source Modification.opn" 이름으로 저장



"Execute > Run "

" Table" 창이 나타나면
"Output" Column을 선택 후

" File > Plot Column "



이상이 간단한 사용 예이며 좀더 복잡한 Operations도 매크로 언어를 이용 간단히 작성이 가능 합니다.

프로그램 구문의 요약 (Summary of the syntax)

Parameters

F1..F9	User defined columns in tables	C1..C9	Constants
R1..R9	User defined references	C0	Value entered each time the operation is executed
M1..M9	Previous results	M0	Independent variable of the current function

Unary Operators. Begin with O and end with :

OD:	Derivative	OH:	Sine
OI:	Integral	OU:	Cosine
OS:	Sum	OG:	Absolute Value
OV:	Mean	OB:	Principal Value (Angle in the range 0..360)
ON:	Number	OP:	Principal Value (Angle in the range -180..180)
OL:	Log	O>:	Maximum Value
OE:	Exponential	O<:	Minimum Value
OQ:	Square Root		

Binary Operators

+	Addition	/	Division
-	Subtraction	^	Exponentiation
*	Multiplication	°	Arctangent (with correct quadrant for result)
%	Search for value from beginning (C1%F1) searches for the first point in F1 in the range given by the independent variable that is equal to C1 and returns the independent value at that point	@	Search for value from end (C1@F1) searches for the first point in F1 in the range given by the independent variable starting from the end that is equal to C1 and returns the independent value at that point

Scripts

Script는 BASIC과 같은 프로그래밍 언어로 반복이 많고 좀더 복잡한 계산에 유용하고 거의 무한하게 사용됩니다.

먼저 주요 개념들을 소개 하면

- (1) Class : 특별한 Object의 내용을 명시하는 정의로 이미 본 프로그램에서 designs, stacks, plots 등이 정의 되어져 있습니다.
- (2) Object : 어떤 실체를 표현하는 한 Methods(subroutines 또는 functions) 또는 Properties (variables 또는 다른 objects)의 정의된 Structure.
- (3) Messages : Object들 간의 Communication
- (4) Properties : 변수 또는 다른 Object로 보통 NameOfObject.NameOfProperty 형식을 갖습니다..
- (5) Methods : Subroutines 또는 functions으로 그 기능에 따라 아래와 같은 형식으로 되는데
NameOfObject.NameOfMethod Parameters.
VariableOrObjectName = NameOfObject.NameOfFunction Parameters
NameOfObject.NameOfSubroutine Parameters
Set DataSet=ThisDesign.CalculateTable(XAxis, YAxis)
Results.SetColumnTitle 1, "Wavelength (nm)"
GetSourceNames IlluminantList, IlluminantFiles, SourceDefinitions
- (6) Declarations : 프로그램 시작 점으로 " Option Explicit " 라는 선언 문으로 사용 되며 변수는

```
Dim : 변수 설정으로 , 변수 이름 as,$, I%(정수) 설정 값  
Dim Result ( 변수 이름) As Double  
Dim J as Long  
Dim I%  
Dim Properties() as Double
```

```
다음에 재 지정 시  
ReDim Properties(1 to 20)
```

(7) Object 만들기

Class는 이미 프로그램에 Scripting에 가능한 Object를 정의해 놓았으며 이미 정의되어 있는 Object는 그대로 어떠한 추가, 수정 없이 사용이 가능하며 아래와 같이 목록에 있는 항목들은 해당 파일이 활성화 되고 Script가 시작 됐을 때 그 것이 Object로 됩니다.

활성화된 파일 사전 정의된 Objects

Active Document	Predefined Objects	Class
Design	ThisSession	Session
	ThisDesign	Design
Stack	ThisSession	Session
	ThisStack	Stack
Material	ThisSession	Session
	ThisMaterials	Materials
Substrate	ThisMaterial	Material
	ThisSession	Session
	ThisMaterials	Materials
Materials	ThisSubstrate	Substrate
	ThisSession	Session
Table	ThisMaterials	Materials
	ThisSession	Session
Other Documents	ThisTable	Table
	ThisSession	Session

ThisSession은 다른 Object에 연결이 될 수 있으므로 반드시 지정된 해당 파일이 열리어 활성화 되어 있어야 하며 ThisMaterials은 substrates를 포함하고 있으며 substrates의 이름을 얻기 위해서는

Dim Names() As String

Names = ThisMaterials.GetSubstrateNames 사용되며

모든 다른 Object들은 먼저 Dim 명시되어 있어야 하는데 Dim MyTable As Object 또는 Dim Dialg As DIALOG_1 (DIALOG_1는 하나의 정의된 dialog object)

다음 단계로 Class의 실제 object는 간단히 " Set " 을 이용하여 가능 합니다.

Set ObjectName = Object Expression

Object 표현에는 많은 다른 방법이 있는데

CreateObject function 생성 : Set ObjectName = CreateObject(Class)

Plot 생성 : Dim MyPlot As Object

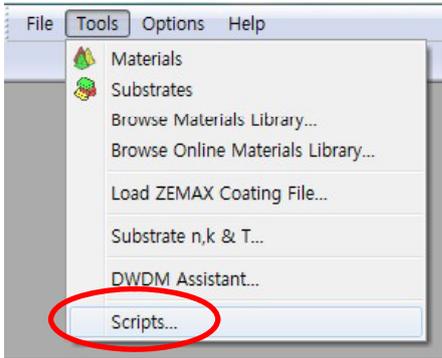
Set MyPlot = CreateObject("EMacleod.PlotCreator")

Table 생성 : Dim MyTable As Object

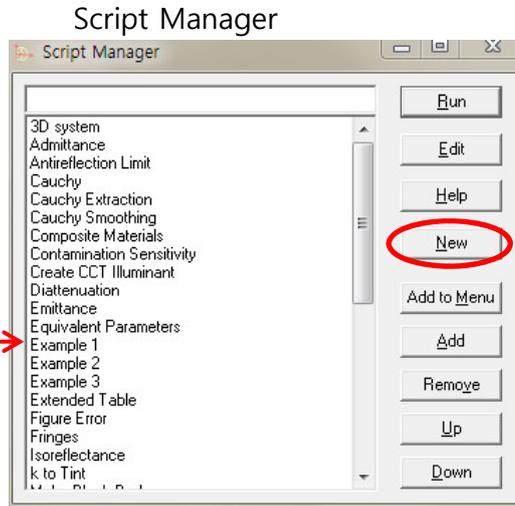
이와 같은 방식으로 Table, 2D, 3D Plot 등은 정리된 내용은 아래 표와 같습니다.

<i>Object</i>	<i>EMacleod Class</i>	<i>Object</i>	<i>Property or Method</i>
Plot	PlotCreator		
ThreeDPlot	ThreeDPlot		
Table	TableCreator	Session	NewTable OpenTable
Design		Session	NewDesign OpenDesign
XAxisDesignParameters		Design	XAxisParameters
YAxisDesignParameters		Design	XAxisParameters
DataSet		Design	CalculateTable
Analysis		Design	Analysis
Admittance		Analysis	CalculateAdmittance CalculateReflectionCoefficient
ElectricField		Analysis	CalculateElectricField
Targets		Design	Targets
BCDataSet		Design	CalculateBC
MatrixDataSet		Design	CalculateMatrix
Stack		Session	NewStack OpenStack
XAxisStackParameters		Stack	XAxisParameters
YAxisStackParameters		Stack	XAxisParameters
DataSet		Stack	CalculateTable
Targets		Stack	Targets
Materials		Session	GetMaterials
Material		Materials	OpenMaterial NewMaterial
Substrate		Materials	OpenSubstrate NewSubstrate
Reference	Reference		

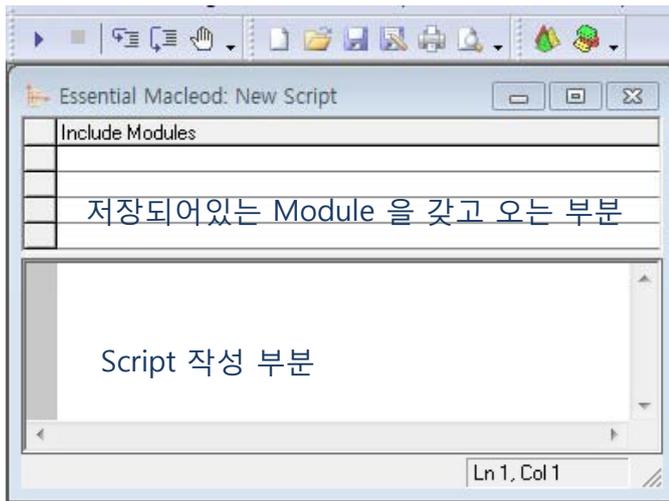
지금부터 Script Manger에 대해 알아 보겠습니다.



"Tools > Scripts.. > New"

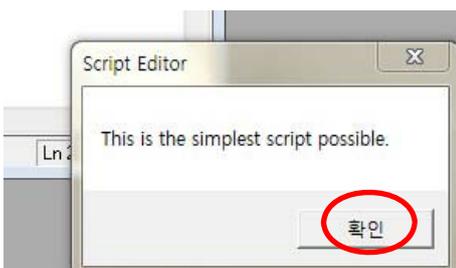
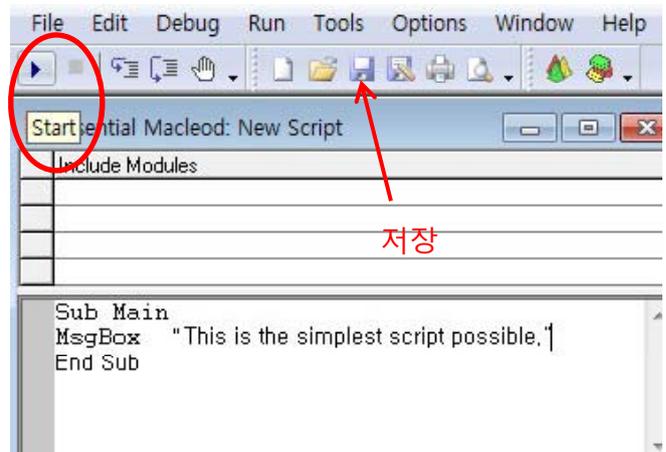


기본적으로 지원되는 Modules



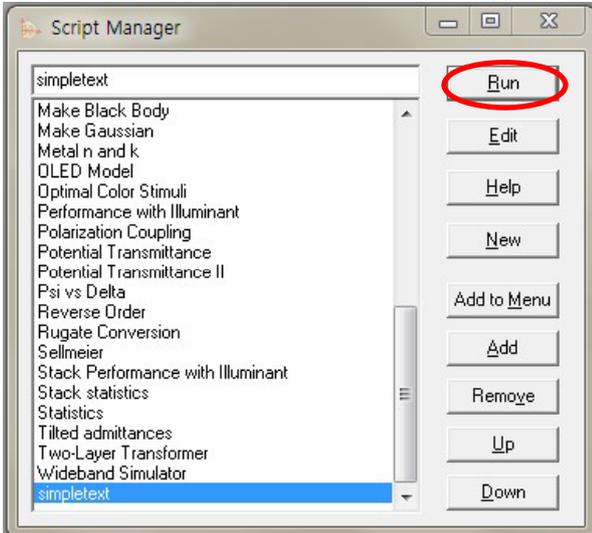
Script에는 반드시 "Main" 불리는 procedure가 있어야 하며 Script Manager가 Script를 실행할 때 먼저 Object들을 정하고 Main을 불러옵니다.

아래와 같이 매우 단순한 Script를 작성
 Sub Main
 MsgBox "This is the simplest script possible."
 End Sub
 입력한 후 "▶ (start) 버튼을 클릭하면 Script가 실행, 결과를 아래와 같이 보여 줍니다.

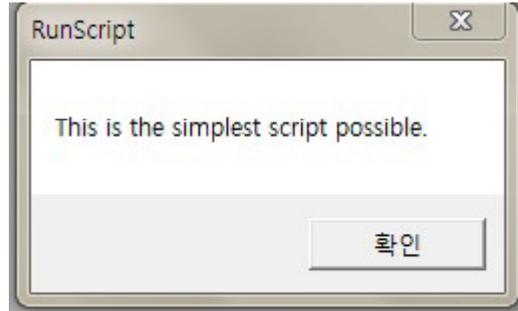


"확인"을 클릭하면 결과 창이 사라집니다.

파일 이름을 "simpletext"로 저장 한 후, Script Manager의 항목에서 해당 파일이 찾아, 선택 후 "Run" 버튼을 이용 그대로 실행이 됩니다.



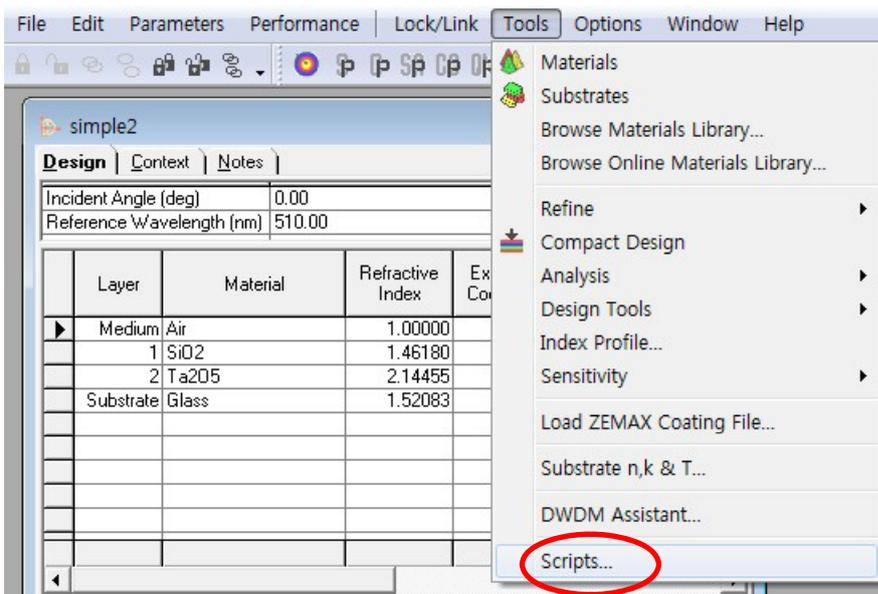
"Tools > Scripts.. >File 선택 > Run "



좀더 복잡한 계산을 하기 위해서 Operation 방식으로 했던 내용을 그대로 Script를 이용하여 작성해 보겠습니다.

먼저 해당 디자인 파일을 열어, 활성화 상태가 되어야 하는데 정의된 Object들은 Script가 실행될 때 활성화된 파일에 의존하며 항상 "ThisSession"라는 이름으로 하나의 Session 대상에 정의 되어져야 있어야 하기 때문 입니다.

해당 디자인 파일을 열고 "Tools > Scripts.. >New" 하여 Script 창을 생성 시킵니다.



일단 Script 창에 "Option Explicit", 다음 줄에 "Sub Main", 몇 줄 뛰어 " End Sub" 입력하고 주석 문은 '..... ' 으로 형식으로 하며 주석 문을 제외 하고는 하나의 syntax는 한 줄 안에 반드시 입력 되어져 있어야 합니다. 그럼 아래와 같이 내용을 작성 합니다.

Option Explicit '시작 점.'

Sub Main

'변수들 지정.'

Dim Performance As Object 'Performance는 본 디자인의 performance를 지정.'

Dim Source As Object ' Source 지정.'

Dim Results As Object ' Table 결과 지정.'

Dim PlotOfResults As Object ' Plot 결과 지정.'

Dim W() As Double ' 파장 배열 지정'

Dim Illuminant() As Double ' 광원 배열 지정'

Dim NewSource() As Double ' new source data 배열지정 '

Dim I As Long ' 계산을 위한 index 설정'

Const Blue = &HFF0000& ' plot line 색. BGR 색상 계 사용'

Const Orange = &H0080FF&

' & : 32 bit 수를 이용되어 진다는 것을 의미.'

' performance 계산 수행 '

Set Performance = ThisDesign.CalculateTable(ThisDesign.XaxisParameters,ThisDesign.YaxisParameters)

' 문장이 길어도 반드시 한 줄 안에 입력 되어 있어야 합니다.'

' 다음은 D65 source information을 불러 오는데 파일 이름을 정확히 확인 합니다.'

Set Source = CreateObject("EMacleod.Reference")

Source.LoadReference "CIE Illuminant D65 1nm.ref"

' the source information 크기를 알게 되며 저장 합니다.'

ReDim W(1 To Source.LastIndex)

ReDim Illuminant(1 To Source.LastIndex)

ReDim NewSource(1 To Source.LastIndex)

'결과를 보여줄 table을 설정 합니다. '

Set Results = ThisSession.NewTable ' 결과를 하나의 table object로 설정하고
적당한 컬럼 제목을 줍니다. '

Results.SetColumnTitle 1, "Wavelength (nm)"

Results.SetColumnTitle 2, "Rel Output"

' 이제는 결과의 표시와 계산 처리를 진행합니다.'

For I=1 To Source.LastIndex

' 먼저 source reference부터 나오는 wavelengths 설정'
W(I)=Source.X(I)

' 다음 plot을 쉽게 하기위한 source data '
' 필요 시 interpolation 이용하여 GetValue method 적용 하거나 '
' interpolation 없이 Illuminant(I)=Source.Y(I) 방식도 사용 가능.'
Illuminant(I)=Source.GetValue(W(I))

' new source 계산을 Interpolate를 이용하고 p-polarization 선택'
NewSource(I)=Illuminant(I)*Performance.InterpolateYp(W(I))

' 결과 Table에 결과를 보여준다 '
' 초기에는 올바른 row의 수를 갖고 있지 않아 AddTableData method 이용하여
wavelength data를 입력하면 row에 더해지고 data를 그 곳에 넣어줍니다.'
Results.AddTableData 1, Format(W(I))

' 이제 새로운 row가 생성되고 new source data를 그 곳에 넣어주게 되었습니다. '
' SetTableData method 사용 , I 를 row를 구별하는데 사용 합니다.'

Results.SetTableData 2, I, Format(NewSource(I))
Next I

' table은 이제 준비됐고 이제는 그 것을 표시 해야 합니다. '

Results.ShowMe
' 결과를 plot 합니다. '
Set PlotOfResults=CreateObject("EMacleod.PlotCreator")
' plot 만들고 axis 이름을 설정'

PlotOfResults.XAxisTitle = "Wavelength (nm)"
PlotOfResults.YAxisTitle = "Relative Output"
' 첫 plot은 D65 source에서 있고 line styles 설정'
PlotOfResults.LineColor=Blue
PlotOfResults.SymbolSize=0
PlotOfResults.LineWidth=1
' first trace에 대한 data 추가'
PlotOfResults.AddTrace W(), Illuminant(), "D65 Source"
' the second trace에 대한 line styles 설정'
PlotOfResults.LineColor=Orange
PlotOfResults.SymbolSize=0
PlotOfResults.LineWidth=3
' 그리고 data.'
PlotOfResults.AddTrace W(), NewSource(), "New Source"
End Sub

한글은 Script 컴파일에 문제가 있을 수 있으므로 영문으로 하여 아래와 같이 작성 합니다.

Option Explicit

Sub Main

```
Dim Performance As Object 'Performance will contain the performance of the design.
Dim Source As Object ' Source will contain the illuminant
Dim Results As Object ' This will be the table of results
Dim PlotOfResults As Object ' This will be the plot of the results
Dim W() As Double 'This array will contain the wavelength'
Dim Illuminant() As Double ' This array will contain the original source relative output'
Dim NewSource() As Double ' This array will contain the new source data '
Dim I As Long ' We will need this as the index for the calculations '
Const Blue = &HFF0000& ' Line color for plot. The order of color is B G R '
Const Orange = &H0080FF& ' Note the use of the & to force the value to be treated
'As a 32 bit number.'
' Now we carry out the calculation of performance '
Set Performance = ThisDesign.CalculateTable(ThisDesign.XaxisParameters,ThisDesign.YaxisParameters)

'Note this should be on one line
'Next we load the D65 source information.
'Warning: check the file Name is correct
Set Source = CreateObject("EMacLeod.Reference")
Source.LoadReference "CIE Illuminant D65 1nm.ref"
'Now we know the size of the source information we can dimension the arrays that 'will store it'
ReDim W(1 To Source.LastIndex)
ReDim Illuminant(1 To Source.LastIndex)
ReDim NewSource(1 To Source.LastIndex)
'We also set up the table that will contain and display the results '
Set Results = ThisSession.NewTable ' Results is now set up as a table object'
' We give the columns appropriate titles '
Results.SetColumnTitle 1, "Wavelength (nm)"
Results.SetColumnTitle 2, "Rel Output"
' Now we are ready to proceed with the calculations and display of results'
For I=1 To Source.LastIndex
' First we set up the wavelengths. These are taken straight from the source reference'
W(I)=Source.X(I)
' Next we need the source data in an array so that we can easily plot it. '
' The GetValue method interpolates if necessary. '
' We could also use Illuminant(I)=Source.Y(I) without interpolation.'
Illuminant(I)=Source.GetValue(W(I))
' Calculation of the new source. We interpolate if necessary and choose p-polarization'
NewSource(I)=Illuminant(I)*Performance.InterpolateYp(W(I))
' We display the results in the Results table '
' To begin with the table does not have the correct number of rows '
' so we insert ' the wavelength data using the AddTableData method '
'that adds a row and ' puts the data into it.'
Results.AddTableData 1, Format(W(I))
' Now the new row has been created we can insert the new source data into it '
' For this we use the SetTableData method and use I to identify the row.'
```

```
Results.SetTableData 2, I, Format(NewSource(I))
```

```
Next I
```

```
'The table is now ready but we need to display it '
```

```
Results.ShowMe
```

```
' We will also plot the results. '
```

```
'The creation of a plot uses a different approach'
```

```
Set PlotOfResults=CreateObject("EMacLeod.PlotCreator")
```

```
' Creating a plot , We set up the axis titles'
```

```
PlotOfResults.XAxisTitle = "Wavelength (nm)"
```

```
PlotOfResults.YAxisTitle = "Relative Output"
```

```
'The first plot will be of the D65 source. We set up the line styles '
```

```
PlotOfResults.LineColor=Blue
```

```
PlotOfResults.SymbolSize=0
```

```
PlotOfResults.LineWidth=1
```

```
'Then we add the data for the first trace'
```

```
PlotOfResults.AddTrace W(), Illuminant(), "D65 Source"
```

```
'Then the line styles for the second trace.'
```

```
PlotOfResults.LineColor=Orange
```

```
PlotOfResults.SymbolSize=0
```

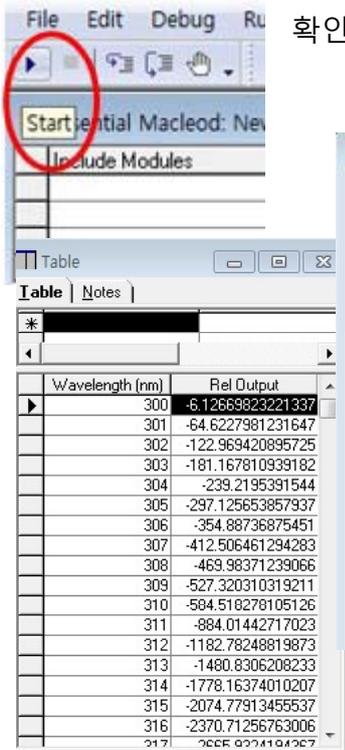
```
PlotOfResults.LineWidth=3
```

```
' And the data.'
```

```
PlotOfResults.AddTrace W(), NewSource(), "New Source"
```

```
End Sub
```

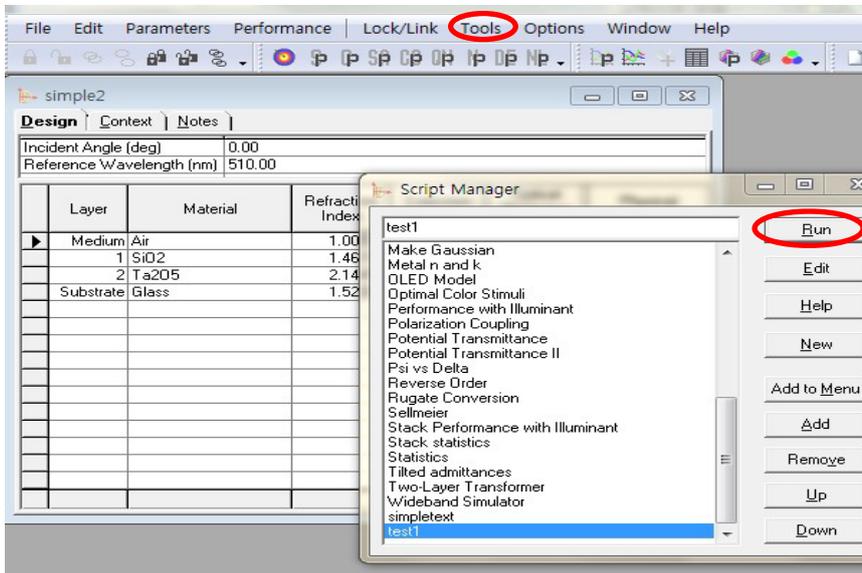
이와 같이 작성이 완료되면 "▶ start 버튼을 눌러 작성된 내용의 에러가 있는지를 확인 하면서 실행하면 table과 plot 창이 나타납니다.



Wavelength (nm)	Rel Output
300	-6.12669823221337
301	-64.6227981231647
302	-122.969420895725
303	-181.167810939182
304	-239.2195391544
305	-297.125653857937
306	-354.88736875451
307	-412.506461294283
308	-469.98371239066
309	-527.320310319211
310	-584.518278105126
311	-641.716246089181
312	-698.914214084236
313	-756.112182079291
314	-813.310150074346
315	-870.508118069401
316	-927.706086064456
317	-984.904054059511



이상이 없으면 이름을 정하여 (예 : test1)저장하고 그 다음부터는 해당 Design 파일을 연 상태에서 " Tools > Script " Script Manager의 항목에서 해당 파일이 찾아, 선택 후 "Run" 버튼을 클릭하면 그대로 실행이 됩니다.



" Tools > Script " Script Manager의 항목에 보면 Example 1,2,3과 다른 항목들이 보이는데 선택 후 "Edit" 하면 작성 내용을 볼 수가 있어 유사 프로그램 작성시 참고 바라며