

IDL Graphics

IDL 8.0 부터는 별 다른 이름 없이 IDL Graphics라 불리는 새로운 그래픽 체계를 제공합니다. 이전 버전을 사용해 보았다면 iTools와 닮은 고품질 그래픽에 다이렉트 그래픽스를 닮은 프로그래머 친화적 사용법을 목표로 개발되었다는 사실을 알 수 있을 것입니다.

아무거나 시작해 봅시다

```
IDL> x=indgen(361)
IDL> y=sin(x*!dtr)
IDL> p=plot(x, y, title='Welcome to IDL 8.0')
```

좌변의 변수(위 예제에선 p 변수)를 가지고 있다면 표출한 그래픽에 대한 완전한 통제권을 쥐고 있는 셈입니다. 그림의 품질을 보면 감이 오겠지만, 이 그래픽의 기반은 오브젝트 그래픽스이며 다이렉트 그래픽과 달리 그려진 그림을 수정하는 것이 가능합니다. 그림창을 닫는 것도 그래픽 변수를 이용합니다.

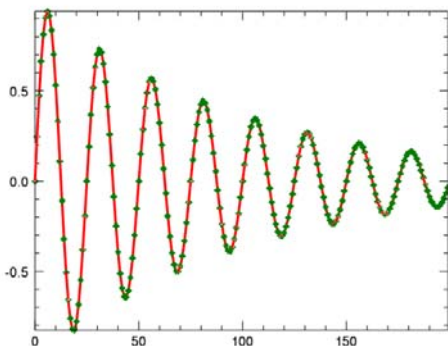
```
IDL> p.close ; 예전 식으로, p->close 도 작동합니다.
```

IDL은 키워드의 언어입니다. PLOT() 함수도 PLOT 프로시저 못지 않게 많은 키워드를 가지고 있습니다. 이제 도움말에서 PLOT을 검색하면 함수와 프로시저 두 가지가 검색될 것입니다. 키워드를 다 기억할 수는 없습니다.

```
IDL> ? plot
```

도움말의 모양도 바뀌었습니다. 도움말에 보이는 것을 몇 가지 따라 해보겠습니다. IDL Graphics에서는 /TEST 키워드를 지원하여 시험용 데이터를 제공합니다. 연습용으로 편리하게 사용할 수 있습니다.

그래픽 변수의 활용

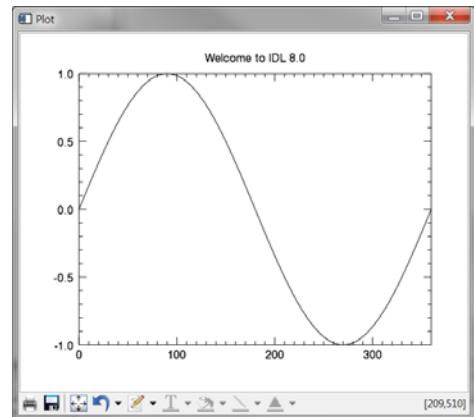


```
IDL> p=plot(/test, '-r3D') ;좌변의 리턴값 p 변수가 매우 중요합니다.
IDL> p.sym_filled=1
IDL> p.sym_color='green'
```

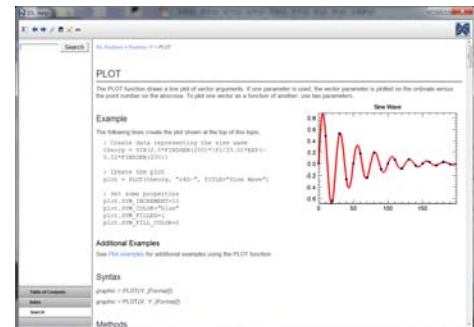
그래픽 변수만 가지고 있다면 해당 창에 대해 모든 일을 수행할 수 있습니다. SYM_FILLED 키워드를 변경하여 심볼의 내부를 채우거나 SYM_COLOR를 변경하여 심볼의 색을 바꾸는 일이 간단하게 이루어 집니다. 새로운(재미있는) 내용이 보이는데, '-r3D'로 보여지고 있는 플롯 포맷 부분입니다. 생략을 할 수도 있고 1~4개의 문자로 선과 심볼을 결정하는 것입니다. 사실상 순서는 상관없이 쓸 수 있는데, -는 실선, r은 빨간색, 3은 선의 굵기, D는 다이아몬드 형 심볼을 의미 합니다. 익숙해지면 이 방식이 사용하기 편할 거라고 생각합니다만,

기존 방식에 익숙하다면, LINESTYLE, COLOR, THICK, SYMBOL을 이용하는 것도 괜찮습니다. 즉 다음 두가지 방법은 같은 결과를 보여주며 내부 작동 방식도 똑같습니다. 사용자가 편한 방법을 선택하여 사용하면 됩니다.

```
IDL> p1=plot(/test, 'S1-b')
IDL> p2=plot(/test, SYMBOL='Star', THICK=1, LINESTYLE=3, color='Blue')
```



iTools와 비슷하지만 마우스 조작 보다는 프로그래밍을 위한 그래픽 환경입니다.



PLOT() 함수 관련 도움말. 예제가 앞에 등장합니다.

플롯 포맷에서 색을 나타내는 알파벳 한 글자는 꼭 소문자로 사용해야 합니다. `LINESTYLE=3` 과 `LINESTYLE='dash_dot'` 은 같은 표현입니다. `Color='Blue'`와 `Color=[0, 0, 255]` 도 같은 표현입니다. IDL Graphics에서는 이와같이 다양한 방법의 표현이 가능하도록 하여 사용자의 혼란을 유도하고 있습니다. ^_^ . 그럴까요? 프로그래머에게 이런 다양한 표현 중 상황에 따라 골라 쓸 수 있게 된다는 것은 꽤 좋은 일입니다. 당분간은 도움말에 의존을 해야 할 테지만요.

예를 들어, SYMBOL 키워드에는 오른쪽 표와 같이 숫자 INDEX를 쓸 수도 있고, 약식기호를 쓸 수도 있고 긴 이름을 쓸 수도 있습니다. SYMBOL=6, SYMBOL='s', SYMBOL='square' 중 어떤 것을 써도 됩니다. 아마도 숫자 6이 사각형이라고 기억하기는 쉽지 않을 것입니다. 그렇지만 메뉴에서 여섯 번째 항목을 골랐을 때 사각형 심볼을 쓰도록 GUI를 설계한다면 이 방법이 편리하겠지요. 조금 익숙해지면 's'와 같은 약어를 쓰는 것이 가장 일반적일 거라고 생각합니다. 저 긴 이름은 아무도 안 쓸 것 같지만, 여러 사람이 보아야 하는 프로그램을 만들 때는 긴 이름을 쓰는 것이 가독성은 최고입니다.

Symbol	Index	Short Name	Long Name
	0		none
+	1	+	plus
*	2	*	asterisk
.	3	.	period or dot
◇	4	D	diamond
▲	5	tu	triangle
■	6	s	square
×	7	x	x
>	8	>	greater_than
<	9	<	less_than
▼	10	td	triangle_down
◀	11	tl	triangle_left
▶	12	ti	triangle_right
▲	13	Tu	tri_up
▼	14	Td	tri_down
◀	15	Tl	tri_left
▶	16	Ti	tri_right
◇	17	d	thin_diamond
⬠	18	p	pentagon
⬡	19	h	hexagon_1
⬢	20	H	hexagon_2
	21		vline
—	22	—	hline
☆	23	s	star
●	24	o	circle

SYMBOL 키워드 대응표

그래픽 변수에 들어 있는 내용들

그래픽 변수에는 도대체 어떤 내용들이 들어 있을까요? print 해 보는 것이 가장 좋은 방법입니다.

```
IDL> print, p1
```

출력되는 내용을 살펴 보면 대략적인 감이 올 것이라고 생각합니다. 감이 오지 않으면 도움말을 찾아 보면 됩니다. 지금 보시는 내용은 Plot() 함수와 관련된 내용이고 Surface(), Contour(), Map() 등 그래픽 종류마다 특성화된 키워드들이 존재합니다. 몇 가지 값을 시험 삼아 수정해 보는 것이 좋겠습니다. 다음과 같이 해 보는 것이죠.

```
IDL> P1.window_title='Try what you want.'
```

그래픽 변수로 수행하는 기본 작업

기술적으로는 그래픽 변수가 오브젝트의 레퍼런스이고, 제공되는 작업 명령어는 메소드입니다. 이렇게 설명하는 것이 더 편한 부류의 사람들이 있어서 이렇게도 설명을 하는 것이고 대부분의 사용자는 그냥 이런 일이 이런 식으로 수행된다고 기억하면 됩니다. 가장 중요한 것은 그래픽 변수를 잃지 않고 가지고 있는 것입니다.

- 화면 저장 : 화면의 내용을 그래픽 파일로 저장합니다. 그래픽 포맷에 따라 사용할 수 있는 키워드가 다릅니다.

```
IDL> p1.save, 'C:/Users/yi/test.pdf', /CENTIMETERS, PAGE_SIZE='A4'
```

```
IDL> p1.save, 'C:/Users/yi/test.jpg'
```

- 화면 캡처 : copyWindow (다이렉트 그래픽스의 TVRD())와 같은 역할을 합니다)

```
IDL> capt=p1.CopyWindow(width=500, border=0)
```

```
IDL> help, capt
```

```
CAPT          BYTE          = Array[3, 426, 323]
```

```
IDL> window, xsize=426, ysize=323
```

```
IDL> tv, capt, /TRUE
```

- 화면 출력 : 프린터로 해당 화면을 출력합니다.

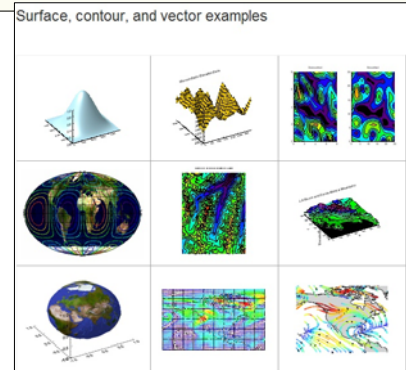
```
IDL> p1.print
```

- 창 닫기 : 열린 창을 닫는 일은 앞에서 한번 해 보았습니다.

```
IDL> p1.close
```

TIP! 예제 둘러 보기

IDL Graphics는 도움말에서 예제를 가장 앞에 내세우고 있습니다. 카테고리 별로 예제 그림의 아이콘들이 제시되어 있으며 관심 있는 그림을 클릭하면 소스 코드와 연관 항목이 나타납니다. 작동 원리에 대한 이해도 필요하겠지만 예제를 몇 가지 따라해 보는 것이 IDL Graphics 체계를 이해하는 가장 빠른 방법이 될 수도 있습니다. 특히, 기존 IDL 다이렉트 그래픽스를 많이 써본 사용자라면 예제를 몇 개 확인해 보는 것만으로 IDL Graphics를 쉽게 활용할 수 있을 것입니다.



Surface, contour, Vector 관련 예제