

# 사용자가 만드는 도구 팔레트



## 도구 팔레트의 개인 설정

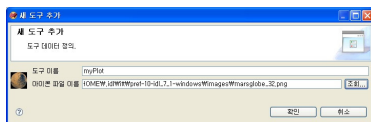
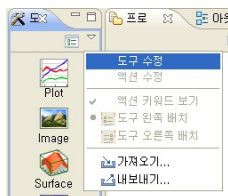
도구 팔레트에는 디폴트로 iTools와 관련된 7개의 도구가 설치되어 있습니다. 사용자는 이 외에도 자신이 자주 사용하는 도구를 추가할 수 있습니다.

도구는 사실상 IDL 프로시저이고, 도구 버튼 또는 이에 속한 액션은 IDL 프로시저를 실행하는 것입니다. 그러므로 도구팔레트를 통하여 실행하려는 프로시저는 반드시 IDL의 경로에 포함되어 있어야 합니다.

## 예제 1 - Plot 도구 추가

IDL 워크벤치에는 기본적으로 Plot 도구가 설치되어 있는데, 이는 iTools의 iPlot를 시동합니다. 이 예제에서는 다이렉트 그래픽의 Plot 명령을 도구로 추가해 보겠습니다. Plot 프로시저가 도구 팔레트에 어떻게 등록되는지 확인해 두세요.

- 도구 팔레트의 메뉴 버튼(아래쪽을 향한 삼각형 버튼)을 클릭하고 **도구 수정**을 실행합니다. 도구 팔레트가 수정 모드로 전환 됩니다.
- **“새 도구”** 버튼을 클릭합니다. “새 도구 추가” 창이 열릴 것입니다.
- 도구 이름 : myPlot 이라고 입력하십시오. 이름은 사실 사용자가 지정하는 어떤 것이어도 좋습니다(한글 가능).
- 아이콘 파일 이름은 아이콘으로 사용하고자 하는 그림 파일을 “조회” 버튼으로 선택하면 됩니다. JPEG, PNG, GIF, BMP 파일 중 어떤 것이어도 사용할 수 있습니다. 아이콘의 크기는 자동으로 조정 됩니다. (이 예에서는 IDL 설치 디렉토리에 examples/data 디렉토리에 있는 marsglobe를 선택했습니다.)

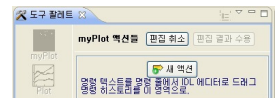


- “확인” 버튼을 클릭합니다.
- 도구 팔레트에 새로운 버튼 myPlot이 등록되어 있습니다. 버튼의 왼쪽에 있는 “편집” 링크를 통해 수정을 할 수 있습니다. IDL에 디폴트로 설치된 Plot, Image 등의 버튼은 편집이 안된다는 사실도 확인하세요. 버튼의 오른쪽의 X 마크는 버튼을 제거하는 것입니다. ≡ 마크를 마우스로 잡아서 이동하면



버튼의 위치를 자유롭게 배치할 수 있습니다.

- “수정 내용 수용” 버튼을 클릭하면 설정한 내용이 저장됩니다.
- myPlot 버튼을 클릭해 보십시오. 아직 아무 일도 일어나지 않습니다. 도구와 연관된 실제 “액션”이 정의되지 않았기 때문입니다. 액션 보기 버튼 을 클릭해 보면 액션이 텅 비어 있음을 확인할 수 있습니다.
- 도구 팔레트 메뉴에서 **액션 수정**을 실행하십시오. “새 액션” 버튼을 클릭합니다.
- 액션 편집 창에 필요한 내용을 입력하십시오. 기본적인 IDL 문법을 고려해야 합니다.



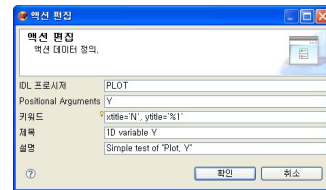
IDL 프로시저 : PLOT

Positional Arguments : Y (일단 단일 변수 가정)

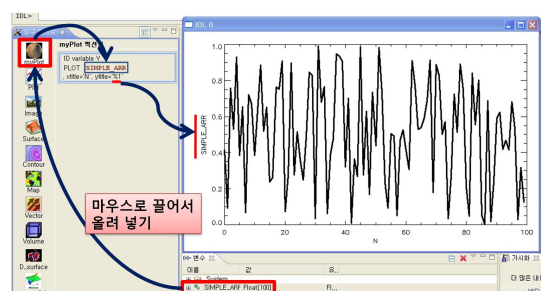
키워드 : xtitle='N', ytitle='%1' (어떤 키워드든 OK)

제목 : 1D Variable Y (액션 리스트에 보일 제목)

설명 : 사용자가 보기 위한 어떤 내용이든 좋음



- %1에 대한 설명이 필요하겠군요. %1은 Positional Argument 중 첫 번째 것의 변수 이름입니다. 만일 사용자가 Hello라는 변수를 던진다면 %1은 Hello로 치환됩니다.
- “편집 결과 수용” 버튼을 클릭하면 액션이 등록됩니다.
- 테스트를 해 보겠습니다. 1차원 배열을 하나 만들어 보세요.  
IDL> simple\_arr=randomu(-1, 100)
- 변수 뷰에 **SIMPLE\_ARR** 변수가 등록되어 있습니다. 마우스로 이를 잡아서 myPlot 버튼에 올려 놓아 보세요.



- 실제 액션은  
IDL> Plot, Y, Xtitle='N', Ytitle="변수이름"  
으로 작동하였습니다.
- 이 정도는 아마 손으로 치는 것이 더 빠를 수도 있을 것입니다. 하지만 키워드가 매우 많이 사용되는 정교한 Plot을 그릴 경우에는 유용하게 쓰일 수 있습니다. 개인적인 설정을 얼마든지 할 수 있으니까요.

## 예제 2 - Plot, X, Y 의 구현

앞의 예에서 1 변수 Plot인 Plot, Y를 구현하였습니다. 하지만 IDL 문법에는 Plot, X, Y 형태도 지원이 됩니다. 이 액션도 추가해 봅시다.

- 도구 팔레트의 메뉴에서 “액션 편집”을 실행하고 “새 액션” 버튼을 클릭합니다.
- 다음 내용을 입력하고 확인 버튼을 클릭합니다.

IDL 프로시저 : PLOT

Positional Arguments : X, Y

키워드 : xtitle='%1', ytitle='%2'

제목 : 1D variable Y vs X

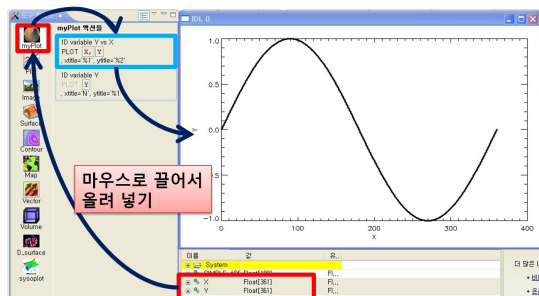
설명 : Simple test of "Plot, x, y"

- “편집 결과 수용” 버튼으로 액션 등록을 완성합니다.
- 테스트할 데이터를 생성합니다.

IDL> x=findgen(361)

IDL> y=sin(x\*dtor)

- 변수 뷰에서 **x, y** 두 변수를 선택하여(Ctrl 키를 누른채로 클릭하면 다중 선택이 됩니다), 이들을 끌어다가 myPlot 버튼 위에 올립니다.

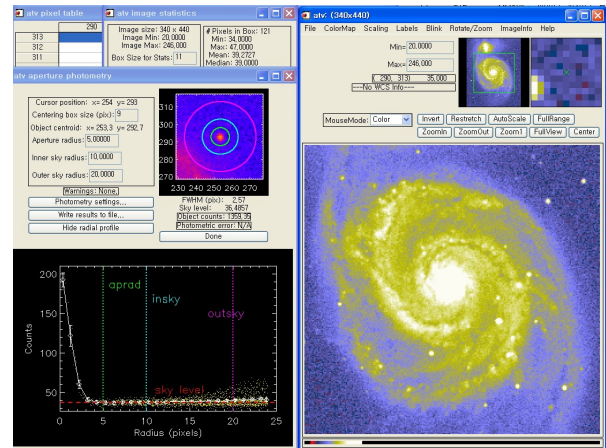


- 결과를 보세요. 변수 두 개를 던지면 변수 두 개를 받아 처리하는 액션이 나서고, 변수 하나를 던지면 변수 한 개를 받아 처리하는 액션이 나서고 있는 것을 확인하십시오.
- 같은 수의 변수(Positional Arguments)를 가지는 액션이 많을 경우에는 변수를 액션에 직접 올려 놓아 실행할 수 있습니다. 도구 버튼에 올려 놓으면 위에 있는 액션부터 우선권을 가집니다(액션의 순서 변경은 ≡ 버튼으로 할 수 있습니다).

## 예제 3 - ATV를 이용한 Fits File 도구 등록

좀 더 의미 있는 예를 들어 보기 위해 Aaron Barth의 ATV를 등록해 보겠습니다. ATV는 SAOImage(DS9)와 같은 기능을 IDL에서 구현한 것으로, Fits 파일 또는 2차원 배열을 열어 보고 분석하는 도구입니다.

<http://www.physics.uci.edu/~barth/atv>



천문학자들이 사용하는 FITS 포맷 전용 도구이므로 IDL Astro 라이브러리와, cmps\_form.pro, 그리고 ATV를 다 운받아 IDL 경로에 포함시켜야 합니다. 제대로 설치되었다면 다음과 같은 명령이 IDL에서 잘 실행되어야 합니다.

IDL> atv, 'D:\myimage.fits' ;fits 파일을 직접 입력

IDL> atv, img ;img는 영상(2차원 배열)

ATV 프로시저는 FITS 영상 파일 문자열이나 2차원 배열 형태의 영상 데이터 두가지를 모두 받아들일 수 있게 설계되어 있습니다.

- 도구 수정으로 들어가 FitsView라는 이름으로 새 도구를 등록합니다. 아이콘은 보기 좋은 어떤 이미지여도 좋습니다.
- 액션 수정으로 들어가 다음과 같은 내용으로 액션을 등록합니다.

IDL 프로시저 : ATV

Positional Arguments : arg1

키워드 : 빈칸으로 둬 (사용 안해도 무방)

제목 : Fits 분석 (사용자가 지정한 제목)

설명 : ATV를 이용한 Fits 분석 도구

- ATV는 Fits 파일이나 2차원 배열(이미지 데이터)를 모두 받을 수 있습니다. 그러므로,
  1. 폴더에서 Fits 파일 아이콘을 직접 끌어다가 FitsView 버튼 위에 놓으면 Fits 파일이 열립니다.
  2. 워크벤치 변수 뷰에서 2차원 배열을 끌어다가 FitsView 버튼 위에 올려 놓으면 해당 이미지가 열립니다.