

# Graphic object creating

This section contains the description and examples on how to create various types of graphic objects in ADA language.

How to create the graphic object:

1. [Graphic object creating](#)
2. [Set the parameters](#)
3. [Completion of the graphic object](#)

## 1. Graphic object creating

You can create a graphic object by calling the call-back procedure [CreateObj](#) in which enter the number of created object. In this table you can find a list of objects and pseudo objects that can be created with the help of this function.

Object	objType	Description
<a href="#">Line</a>	0	Creates the object of <a href="#">Line</a> type.
<a href="#">PLine</a>	1	Creates the object of <a href="#">Multiline</a> or <a href="#">Polygon</a> .
<a href="#">DLine</a>	2	Creates the object of <a href="#">Disjointed multiline</a> .
<a href="#">Arc</a>	3	Creates the object of <a href="#">Arc</a> .
<a href="#">3Arc</a>	4	Creates the object of <a href="#">3-point arc</a> .
<a href="#">Box</a>	5	Creates the object of <a href="#">Rectangle</a> .
<a href="#">PAngle</a>	6	Creates the object of <a href="#">Polygon</a> .
<a href="#">Circle</a>	7	Creates the object of <a href="#">Circle</a> .
<a href="#">PiArc</a>	8	Creates the object of <a href="#">Circle sector</a> .
<a href="#">Ellipse</a>	11	Creates the object of <a href="#">Ellipse</a> .
<a href="#">Text</a>	12	Creates the object of <a href="#">Text</a> .
<a href="#">Group</a>	32	Creates the object Group of objects that groups more objects in one group (till the group is closed).
<a href="#">LineCombined</a>	-1	Creates the object of Line type that (on condition that the line has the same attributes as the created lines) combines with the previous lines to the one object of <a href="#">Multiline</a> or <a href="#">Disjointed multiline</a> .
<a href="#">PLineAdd</a>	-2	Adds the parameters in the created object of <a href="#">Multiline</a> type (if it does not exist, create the new one).

## 2. Set the parameters of new object

After creating the new object, set its parameters by calling [call-back procedures](#) of Set\_xxx type. First parameter defines the type of parameter of graphic object which is being created, the second or third one represents new value of this parameter. In this table you can find the group of parameters:

Parameters	Description
<a href="#">Position</a>	Position points, i.e. position or distance from the previous point.
<a href="#">Lines</a>	Line parameters.
<a href="#">Fill</a>	Parameters of fill.
<a href="#">Circle objects</a>	Parameters for circle objects.
<a href="#">Texts</a>	Parameters for texts.

## 3. Completion of the graphic object

[Call-back procedure](#) ObjAction closes the graphic object. This procedure executes the action with the object according to the defined parameter.

### Characteristics of actions:

Action	objType	Description
<a href="#">closeFigure</a>	0	Closes the position points of Polyline - the object is changed to polygon.

closeObject	1	Closes the object (the object is closed automatically as soon as you start creating the new object).
closeGroup	2	Closes the group of objects - completes the current object and last group of objects.
closeAll	3	Closes all objects - completes the current object and all group of objects. The function with this parameter is called automatically when finishing the import.

The graphic objects [Line](#) up to [Text](#) are closes automatically as soon as the next object is created.



#### Related pages:

[Import of vector formats into pictures of D2000 System](#)  
[Graphic object creating](#)