

## 12. Geometry 공통

### 12.1. Geometry 관련 Method 정리

#### 12.1.1. countBoundaries()

- 파라미터 : 없음
- 내용 : Boundary 수 출력
- 출력값 : Boundary수
- 스크립트 예 1 (모델러에서 Line 1개를 정의한 후 테스트)

행번	스크립트
1	\$ENGINE = VBSCRIPT
2	MsgBox database.getObject("line",1).countBoundaries()
2	Line 1번의 Boundary 수를 메시지창으로 표시

#### 12.1.2. getAssignments([attributeType])

- 파라미터 : "Loading","Support","Material" 등과 같이 Geometry에 적용되는 속성 종류
- 내용 : Geometry에 적용된 속성값을 가지는 Objects들의 배열.  
배열로 보내는 이유는 "Loading"의 경우 같은 Geometry에 여러 개가 적용될 수 있기 때문이다.
- 출력값 : Geometry에 적용된 속성값을 가지는 Objects들의 배열
- 스크립트 예 1 (모델러에서 Surface 1개를 정의하고 Mesh를 구성한 후 테스트)

행번	스크립트
1	\$ENGINE = VBSCRIPT
2	set surface = database.getObject("surface", 1)
3	asgnArray = surface.getAssignments("Mesh")
4	If (UBound(asgnArray) < 0) Then
5	MsgBox "Mesh 속성이 적용되지 않은 Surface입니다."
6	Else
7	set attr = asgnArray(0).getAttribute()
8	call msgbox(attr.getValue("size"))
9	End If
2	Surface 1번을 surface라는 object로 지정
3	1번 Surface에 적용된 Mesh 속성을 asgnArray 배열로 저장. mesh는 같은 Geometry에 1개만 적용될 수 있는 것이므로, 사실상은 asgnArray(0) 만이 유효한 값을 가지고 있게 된다.
4	asgnArray 배열변수의 크기가 < 0 이라는 의미는 Mesh 속성이 적용된 것이 없다는 것이다.
5	Mesh 속성 적용이 없다는 메시지 안내
6	Mesh 속성 적용된 것이 있다면,
7	asgnArray(0), 즉 Mesh 속성값을 사용자가 확인할 수 있는 변수명으로 전환시킨 attr Object 정의
8	1번 Surface에 적용된 요소의 크기 설정 상태를 메시지창으로 표시

- 주의 : 위 예제에서 Mesh가 적용된 것이 없는 상태에서 7번 문장을 실행하면 asgnArray(0) 이 존재하지 않는 상태이기 때문에 에러가 발생합니다. 4번 문장으로 이러한 상황에 대해 대비하여 에러발생을 방지합니다.

## ■ 참고 :

8번 문장에서 `getValue("size")` 외에 `getValue("xDivisions")`, `getValue("yDivisions")` 등을 사용할 수도 있습니다. 이와 같이 Geometry에 적용된 속성별로 사용자가 추출할 수 있는 값들의 종류는 `getValueNames()`를 사용하여 확인할 수 있습니다.

행번	스크립트
1	\$ENGINE = VBSCRIPT
2	set surface = database.getObject("surface", 1)
3	asgnArray = surface.getAssignments("Mesh")
4	set attr = asgnArray(0).getAttribute()
5	valuenames = attr.getValueNames()
6	For i=0 To UBound(valuenames)
7	MsgBox valuenames(i)
8	Next
2	Surface 1번을 surface라는 object로 지정
3	1번 Surface에 적용된 Mesh 속성을 asgnArray 배열로 저장. mesh는 같은 Geometry에 1개만 적용될 수 있는 것이므로, 사실상은 asgnArray(0) 만이 유효한 값을 가지고 있게 된다.
4	asgnArray(0), 즉 Mesh 속성값을 사용자가 확인할 수 있는 변수명으로 전환시킨 attr Object 정의
5	attr Object에서 추출할 수 있는 속성값들의 변수명을 valuenames라는 배열로 저장
6-8	valuenames 배열에 저장된 값들(Mesh 속성과 관련한 속성값 확인용 변수명)을 메시지창으로 확인. - 결과: size, xDivisions, yDivisions, transition, allowIrregular, DefinedByName

12.1.3. `getElement()`

## ■ 파라미터 : 없음

## ■ 출력값 : Geometry에 적용된 Element Objects 들을 배열로 전달.

## ■ 스크립트 예 1 (모델러에서 Surface 1개를 정의하고 Mesh를 구성한 후 테스트)

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	set surface = database.getObject("surface", 1)
3	attrArray = surface.getElements()
4	For i=0 To UBound(attrArray)
5	MsgBox attrArray(i).getName()
6	Next
2	Surface 1번을 surface라는 object로 지정
3	1번 Surface에 정의된 요소 Object를 배열 attrArray 에 저장
4-6	각 요소의 번호를 메시지창으로 표시

■ 참고 : Element Object와 관련한 Method들은 별도로 다루게 됩니다. 5번행의 `getName()`은 요소의 번호를 출력하는 함수입니다.

#### 12.1.4. getHOFs()

- 파라미터 : 없음
- 출력값 : 현재 Geometry Object의 상위 Geometry Object를 배열로 출력
- 스크립트 예 1 (모델러에서 Surface 1개를 정의하고 Mesh를 구성한 후 테스트)

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	set line = database.getObject("Line", 1)
3	TmpGeometry = line.getHOFs()
4	For i=0 To UBound(TmpGeometry)
5	MsgBox TmpGeometry(i).getName()
6	Next
2	Line 1번을 line 이라는 이름의 object로 지정
3	1번 Line의 상위 Object (즉, Surface 1번)를 TmpGeometry 배열로 저장
4-6	1번 Line을 사용하는 Surface들의 번호 출력 (현재는 1개의 Surface만을 정의하였으므로 1회만 출력)

#### 12.1.5. getID()

- 파라미터 : 없음
- 내용 : Geometry object의 번호를 확인.
- 출력값 : Geometry object의 번호
- 스크립트 예 1 (모델러에서 surface 1개를 정의하고 Mesh를 적용한 후 테스트)

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	Set surface = database.getObject("Surface",1)
3	eleArray = surface.getElements()
4	For i=0 To UBound(eleArray)
5	MsgBox eleArray(i).getID()
6	Next
4	Surface의 모든 요소를 eleArray 배열로 지정
5	For 문으로 정의된 요소수 만큼 반복
6	element 번호를 순차적으로 메시지창에 출력

## 12.1.6. getLOFs()

- 파라미터 : 없음
- 출력값 : 현재 Geometry Object의 하위 Geometry Object를 배열로 출력
- 스크립트 예 1 (모델러에서 Surface 1개를 정의하고 Mesh를 구성한 후 테스트)

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	set surface = database.getObject("Surface", 1)
3	TmpGeometry = Surface.getLOFs()
4	For i=0 To UBound(TmpGeometry)
5	MsgBox TmpGeometry(i).getName()
6	Next
2	Surface 1번을 surface 이라는 이름의 object로 지정
3	1번 Surface의 하위 Object (즉, Line 1,2,3,4번)를 TmpGeometry 배열로 저장
4-6	1번 Surface 구성에 포함된 Line들의 번호 출력

## 12.1.7. getName()

- 파라미터 : 없음
- 내용 : Geometry object의 절점 배열
- 출력값 : Geometry object의 절점
- 스크립트 예 1 (모델러에서 surface 1개를 정의하고 Mesh를 적용한 후 테스트)

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	Set surface = database.getObject("Surface",1)
3	eleArray = surface.getElements()
4	For i=0 To UBound(eleArray)
5	MsgBox eleArray(i).Name()
6	Next
4	1번 Surface의 모든 요소를 eleArray 배열로 지정
5	For 문으로 정의된 요소수 만큼 반복
6	element 번호를 순차적으로 메시지창에 출력

### 12.1.8. getNodes()

- 파라미터 : 없음
- 출력값 : 현재 Geometry Object에 포함되어 있는 절점들을 배열로 저장
- 스크립트 예 1 (모델러에서 Surface 1개를 정의하고 Mesh를 구성한 후 테스트)

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	set surface = database.getObject("Surface", 1)
3	nodeArray = surface.getNodes()
4	For i=0 To UBound(nodeArray)
5	MsgBox nodeArray(i).getName()
6	Next
2	Surface 1번을 surface 이라는 이름의 object로 지정
3	surface Object 내의 절점 Objects 들을 배열로 저장. nodeArray(0)은 1번 Surface의 첫 번째 절점.
4-6	Surface 1번에 포함된 절점들의 번호 출력 (절점 개수 만큼 반복)

### 12.1.9. getTypeCode()

- 파라미터 : 없음
- 출력값 : Object의 종류를 숫자로 출력

1	point	2	Line
3	Combined line	4	Surface
5	Volume or Hollow volume		

- 스크립트 예 1 (모델러에서 Surface 1개를 정의하고 Mesh를 구성한 후 테스트)

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	set surface = database.getObject("Surface", 1)
3	MsgBox surface.getTypeCode()
2	Surface 1번을 surface 이라는 이름의 object로 지정
3	Surface 1번의 object 종류 번호 출력

## 12.1.10. isMemberOfGroup(group)

- 파라미터 : 없음
- 출력값 : 해당 Object가 선택한 group에 속해 있는지 여부를 출력 (True / False)
- 스크립트 예 1 (모델러에서 Volume 3개를 정의하고 각 volume을 1stVolume, 2ndVolume, 3rdVolume으로 지정한 후 테스트)

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	set volume = database.getObject("Volume", 1)
3	MsgBox volume.isMemberOfGroup("2ndVolume")
2	Volume 1번을 volume 이라는 이름의 object로 지정
3	Volume 1번이 2ndVolume이라는 그룹에 속해있는지 확인

## 12.1.11. isSelected()

- 파라미터 : 없음
- 출력값 : 해당 Object가 선택되어있는지 여부를 출력 (True / False)
- 스크립트 예 1 (모델러에서 Volume 3개를 정의한 후 테스트)

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	set volume = database.getObject("volume",1)
3	msgbox volume.isSelected()
2	Volume 1번을 volume 이라는 이름의 object로 지정
3	Volume 1번이 선택되어있는지 확인

## 12.1.12. isVisible()

- 파라미터 : 없음
- 출력값 : 해당 Object가 visible 상태인지 여부를 출력 (True / False)
- 스크립트 예 1 (모델러에서 Volume 3개를 정의한 후 테스트)

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	set volume = database.getObject("volume",1)
3	msgbox volume.isVisible()
2	Volume 1번을 volume 이라는 이름의 object로 지정
3	Volume 1번이 visible 상태인지 확인

### 12.1.13. setName(newName)

- 파라미터 : 새로정의할 이름
- 내용 : Geometry object의 이름을 변경
- 출력값 : Geometry object의 이름
- 스크립트 예 1 (모델러에서 surface 1개를 정의한 후 테스트)

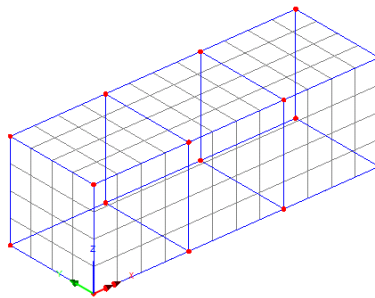
행번	스크립트
1	\$ENGINE=VBSCRIPT
2	Set surface = database.getObject("Surface",1)
3	Call surface.setName("newsurface")
4	MsgBox surface.getName()
2	1번 Surface를 surface라는 Object로 정의
3	1번 Surface의 이름을 newsurface로 변경
4	1번 Surface의 이름을 출력

## 12.2. 예제

### 12.2.1. Volume과 Surface

아래 그림과 같이 Modeler에서 한 변의 길이가 10 인 Volume 3개를 정의하고, Solid 요소를 부여합니다. 모델러에 작도되어 있는 Volume Object를 이용해서 아래와 같이 각 Volume에 포함된 surface의 개수를 메시지창으로 출력하는 스크립트를 작성합니다.

- 요구 결과
- 1번 Volume에 포함된 Surface는 6개입니다.
- 2번 Volume에 포함된 Surface는 6개입니다.
- 3번 Volume에 포함된 Surface는 6개입니다.



행번	스크립트
1	\$ENGINE=VBSCRIPT
2	volumeArray = database.getObjects("Volume")
3	For i=0 to ubound(volumeArray)
4	SurfaceArray = VolumeArray(i).getLOFs()
5	MsgBox VolumeArray(i).getID() & "번 Volume에 포함된 Surface는 "&(UBound(SurfaceArray)+1)&"개
6	입니다."
7	Next

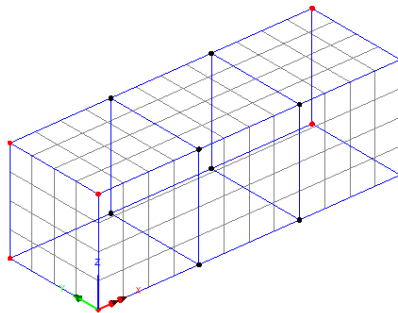
### 12.2.2. Volume과 Point

앞서와 같은 모델을 사용하되, vol\_no = 2 라는 변수를 정의합니다.

vol\_no 로 지정된 번호를 가진 Volume에 포함된 포인트만 선택되도록 합니다.

#### ■ 요구 결과

아래 그림과 같이 Volume 2번에 포함된 포인트가 선택되어 지도록 한다.



행번	스크립트
1	\$ENGINE=VBSCRIPT
2	vol_no=3
3	Set volume = database.getObject("Volume", vol_no)
4	SurfaceArray = Volume.getLOFs()
5	For i=0 To UBound(SurfaceArray)
6	LineArray = SurfaceArray(i).getLOFs()
7	For j=0 To UBound(LineArray)
8	PointArray = LineArray(j).getLOFs()
9	PointselectArray=selection.getObjects("point")
10	For k=0 To UBound(PointArray)
11	Call selection.add("point",PointArray(k).getID())
12	Next
13	Next
14	Next

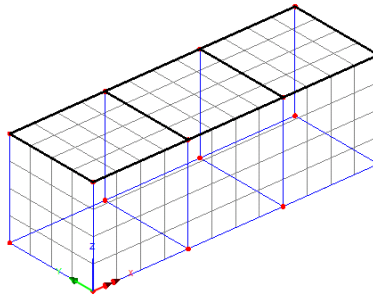


### 12.2.3. 일부 Surface 선택

앞서와 같은 모델을 사용하되, z 좌표가 5 이상인 Point로만 구성된 Surface만 선택되도록 합니다.

■ 요구 결과

아래 그림과 같이 상단의 3개 Surface 가 선택되도록 합니다.



행번	스크립트
1	\$ENGINE=VBSCRIPT
2	VolumeArray = database.getObjects("Volume")
3	For i=0 To UBound(VolumeArray)
4	SurfaceArray = VolumeArray(i).getLOFs()
5	For j=0 To UBound(SurfaceArray)
6	is_surfacetop = 1
7	LineArray = SurfaceArray(j).getLOFs()
8	For k=0 To UBound(LineArray)
9	PointArray = LineArray(k).getLOFs()
10	For l=0 to UBound(PointArray)
11	Pointz = PointArray(l).getZ()
12	If (PointZ < 5) Then
13	is_surfacetop = 0
14	End If
15	Next
16	Next
17	If (is_surfacetop = 1) Then
18	Call selection.add("Surface",SurfaceArray(j).getID())
19	End If
20	Next
21	Next