

13. Attribute의 정의 및 적용

13.1. Attribute의 정의

LUSAS Modeler에서 Attribute는 기하특성, 재료특성, 경계조건, 하중조건 등을 의미합니다.

이러한 특성값의 정의하는 프로그래밍은 Script Recording을 통해서 얻을 수 있으며, 아래의 예시와 같습니다.

13.1.1. Geometric Properties

- Section Library에서 선택하여 정의하는 경우 (Beam 요소)

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	set attr = database.createGeometricLine("TestBeam")
3	call attr.setValue("elementType", "3D Thick Beam")
4	call attr.setBeam(0,0368, 5,042E-3, 0,156E-3, 0,0, 9,26E-6, 0,0180687, 0,0196928, 0,0, 0,0, 0)
5	call attr.setPlastic(0,0, 0,0, 0,0, 0,0, 0,0, 0)
6	call attr.setSectionName("", 0)
7	call attr.addSectionQuad(-0,4122, -9,75E-3, 0,4122, -9,75E-3, -0,4122, -9,75E-3, 5, 1, 0)
8	call attr.addSectionQuad(-0,4633, 0,15385, -0,4313, 0,02885, -0,4633, 0,02885, 1, 5, 0)
9	call attr.addSectionQuad(-0,4633, -0,02885, -0,4313, -0,02885, -0,4313, -0,15385, -0,4633, -0,15385, 1, 5, 0)
10	call attr.addSectionQuad(-0,4633, 0,15385, 0,4633, 0,02885, 0,4313, 0,02885, 0,4313, 0,15385, 1, 5, 0)
11	call attr.addSectionQuad(-0,4633, -0,02885, -0,4313, -0,15385, -0,4633, -0,4313, -0,02885, 1, 5, 0)
12	call attr.addSectionQuad(-0,4122, -9,75E-3, -0,4122, -9,75E-3, 0,1120390092903448, -0,4122, -9,75E-3, 2, 2, 0)
13	call attr.addSectionQuad(-0,4195092535581732, -0,1120390092903437, -0,4257057579520663, -0,01534426047933689, -0,4195092535581732, 0,1120390092903448, 2, 2, 0)
14	call attr.addSectionQuad(-0,4195092535581732, -0,01534426047933689, -0,4633, -0,1534426047933689, -0,4257057579520663, 0,01534426047933689, 2, 2, 0)
15	call attr.addSectionQuad(-0,4298460990709656, -0,02154074644182673, -0,4633, -0,01534426047933689, -0,4257057579520663, -0,01534426047933689, 2, 2, 0)
16	call attr.addSectionQuad(-0,4313, -0,02885, -0,4633, -0,02885, -0,4633, -0,2154074644182673, -0,4298460990709656, -0,02154074644182673, 2, 2, 0)
17	call attr.addSectionQuad(-0,4195092535581732, -0,01534426047933689, -0,4633, -0,01534426047933689, -0,4257057579520663, 0,01534426047933689, 2, 2, 0)
18	call attr.addSectionQuad(-0,4298460990709656, -0,02154074644182673, -0,4633, -0,02154074644182673, -0,4633, -0,02885, 0,1120390092903442, -0,0120390092903442, 0,1120390092903442, 2, 2, 0)
19	call attr.addSectionQuad(-0,4122, -9,75E-3, -0,4122, -9,75E-3, 0,1120390092903442, -0,0120390092903442, 0,1120390092903442, 2, 2, 0)
20	call attr.addSectionQuad(-0,4195092535581732, -0,01120390092903442, -0,4257057579520663, -0,01534426047933689, -0,4195092535581732, -0,01120390092903442, 2, 2, 0)
21	call attr.addSectionQuad(-0,4298460990709656, -0,01534426047933689, -0,4633, -0,01534426047933689, -0,4257057579520663, -0,01534426047933689, 2, 2, 0)
22	call attr.addSectionQuad(-0,4298460990709656, -0,02154074644182673, -0,4633, -0,02154074644182673, -0,4298460990709656, -0,02154074644182673, 2, 2, 0)
23	call attr.addSectionQuad(-0,4298460990709656, -0,02154074644182673, -0,4633, -0,02154074644182673, -0,4298460990709656, -0,02154074644182673, 2, 2, 0)
24	call attr.addSectionQuad(-0,4298460990709656, -0,02154074644182673, -0,4633, -0,02154074644182673, -0,4298460990709656, -0,02154074644182673, 2, 2, 0)
25	call attr.addSectionQuad(-0,4298460990709656, -0,02154074644182673, -0,4633, -0,02885, 0,4313, -0,02885, 2, 2, 0)
26	call attr.addFibre("1", 0,4633, -0,15385, 0)
27	call attr.addFibre("12", 0,4633, 0,15385, 0)
28	call attr.addFibre("13", -0,4633, 0,15385, 0)
29	call attr.addFibre("14", -0,4633, -0,15385, 0)
30	call attr.setLibNames(0, "UK Sections", "Universal Beams (BS4)", "914x305x289kg UB")
31	call attr.setRotateFlag(0, 0)
32	call attr.setDescription()
33	redim temp(4)
34	temp(0) = 0,3077
35	temp(1) = 0,9266
36	temp(2) = 0,032
37	temp(3) = 0,0195
38	temp(4) = 0,0191
39	redim names(4)
40	names(0) = "b"
41	names(1) = "d"
42	names(2) = "t"
43	names(3) = "tw"
44	names(4) = "r"
45	call attr.setStandardSection("l", names, temp, 0)
46	erase temp
47	erase names
48	set attr = nothing
2	기하특성값 데이터셋의 이름 부여
3	기하특성값의 종류 지정
7-25	기하형상의 단면을 화면에 표시하기 위해, 단면의 형상을 여러 조각의 사각형으로 분할하고, 각 사각형의 꼭지점 좌표를 단면의 중립축을 원점으로 하는 값으로 표시한 것. (생략가능)
26-29	단면의 형상이 작도되면, 보 해석의 결과로 얻은 단면력으로부터 응력을 산출할 수 있는데, 이 때 응력을 산출하고자 하는 위치를 단면 내에서의 좌표로 지정함.
30-48	Modeler에 내장된 단면 DB의 정보. 추후 모델러에서 단위 변경시 7-29 값이 자동업데이트

LPI 강좌

- 직접 입력하여 정의하는 경우 (Beam 요소)

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	set attr = database.createGeometricLine("Beam(1x1)")
3	call attr.setValue("elementType", "3D Thick Beam")
4	call attr.setBeam(1.0, 0.0833333333333333, 0.0833333333333333, 0.0, 1.0, 1.0E3, 1.0E3, 0.0, 0.0, 0)
5	call attr.setPlastic(0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0)
6	call attr.setSectionName("", 0)
7	call attr.setRotateFlag(0, 0)
8	call attr.setDescription()
9	set attr = nothing
2	기하특성 데이터셋의 제목
3	기하특성값의 종류 지정
4	보 요소의 특성값 (A, lyy,lzz,lyz,Jxx,Asy,Asz,Ry,Rz)
5	보 요소의 기하비선형해석을 위한 특성값
7	기본 정의 상태에서의 회전각(베타각)

- 직접 입력하여 정의하는 경우 (Surface 요소)

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	call database.createGeometricSurface("SGeo"),setSurface(0.1, 0.0)
2	기하특성 데이터셋의 제목은 'SGeo'로 하고 두께 0.1, 편심 0으로 하는 데이터셋 정의

13.1.2. Material Properties

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	set attr = database.createIsotropicMaterial("Concrete1", 3.0E10, 0.2, 2.4E3)
3	set attr = nothing
4	set attr = database.createIsotropicMaterial("Concrete",3.0E10, 0.2,2.4E3).setValue("alpha",10.0E-6, 0)
5	set attr = nothing
2	재료특성 데이터셋의 제목은 'Concrete1'로 하고 탄성계수, 포화송비, 질량을 정의
4	재료특성 데이터셋의 제목은 'Concrete2'로 하고 탄성계수, 포화송비, 질량, 선팽창계수, 댐핑을 정의

13.1.3. Support Conditions

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	call database.createSupportStructural("Support").setStructural("R","R","R","F","F","F","F","F","F")
2	재료특성 데이터셋의 제목은 'Support'로 하고, 병진변위 x,y,z에 대해 구속 정의

제13장 Attribute의 정의 및 적용

13.2. Attribute의 적용

13.2.1. Selection을 사용하는 적용 방법

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	call database.getAttribute("Line Geometric", "Beam(1x1)").assignTo(selection, assignment)
3	call database.getAttribute("Isotropic Material", "Concrete1").assignTo(selection, assignment)
4	call database.getAttribute("Structural Support", "Support").assignTo(selection, assignment)
2	'Beam(1x1)' 이름을 가진 Line Geometric 데이터셋을 모델러에서 선택되어져 있는 Object에 적용
3	'Concrete1' 이름을 가진 등방성재료특성을 모델러에서 선택되어져 있는 Object에 적용
4	'Support' 이름을 가진 구속조건을 모델러에서 선택되어져 있는 Object에 적용

13.2.2. Object를 사용하는 적용 방법

- 단일 Object에 적용

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	Set Line = database.GetObject("line",24)
3	Call database.getAttribute("Line Geometric", "Beam(1x1)").assignTo(line, assignment)
2	24번 Line을 'Line'이라는 Object로 지정.
3	'Beam(1x1)' 이름을 가진 Line Geometric 데이터셋을 24번 Line에 적용

- Object들을 가지는 배열

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	Lines = database.GetObjects("line")
3	Call database.getAttribute("Line Geometric", "Beam(1x1)").assignTo(line, assignment)
2	모델러상에 있는 Line Object들을 값으로 갖는 배열 Lines를 정의
3	'Beam(1x1)' 이름을 가진 Line Geometric 데이터셋을 모델러 상의 모든 Line 들에 적용

13.2.3. Group을 사용하는 적용 방법

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	set group = database.getGroupByName("Lines")
3	call database.getAttribute("Line Geometric", "Beam(1x1)").assignTo(group, assignment)
4	set group = nothing
2	모델러 상에서 'Lines'라는 이름의 그룹을 group 이라는 Object로 지정
3	'Beam(1x1)' 이름을 가진 Line Geometric 데이터셋을 Lines 그룹에 속한 Line들에 적용

LPI 강좌

13.3. Attribute의 적용 취소 (deassign)

적용한 속성을 다시 해제시키는 것은 assignTo 대신 deassignFrom을 사용하면 되며 아래 예와 같다.

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	call assignment.setAllDefaults()
3	call assignment.setLoadsetOff()
4	call assignment.setDeassignMatch("Attribute only")
5	call database.getAttribute("Line Geometric", "Beam(1x1)").deassignFrom(selection, assignment)
6	call database.getAttribute("Line Geometric", "Beam(1x1)").deassignFrom("all", assignment)
5	'Beam(1x1)' 이름을 가진 Line Geometric 데이터셋을 모델러상에서 선택되어져 있는 Line에서 해제
6	'Beam(1x1)' 이름을 가진 Line Geometric 데이터셋을 모델러상의 모든 Line에서 해제

참고 : "all" 이라고 하면 모델러상의 모든 Object를 의미하게 됩니다.

13.4. 기타

13.4.1. getAttribute([속성종류],[속성제목])

앞서 예제에서 이미 사용한 바와 같이

```
set geoAttr = database.getAttribute("Line Geometric", "Beam(1x1)")
```

와 같이 사용할 수 있다.

13.4.2. getValueNames()

- 파라미터 : 없음
- 출력값 : Attribute에 정의된 내용 목록
- 스크립트 예

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	set attr = database.getAttribute("Line Geometric", "Beam(1x1)")
3	valuenames = attr.getValueNames()
4	For i=0 To UBound(valuenames)
5	MsgBox valuenames(i)
6	Next
2	'Beam(1x1)' 이름을 가진 Line Geometric 데이터셋을 attr이라는 Object로 정의
3-6	'Beam(1x1)' 이름을 가진 Line Geometric 데이터이 가지고 있는 변수값들의 제목을 메시지창으로 표시

■ 결과

아래와 같이 메시지창으로 저장하고 있는 속성값들의 변수명이 출력됩니다.



제13장 Attribute의 정의 및 적용

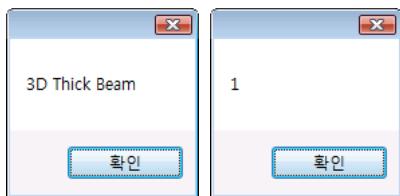
13.4.3. getValue(varName, [row], [units])

- 파라미터 : 알고자 하는 변수명
- 출력값 : Attribute에 저장된 속성값 중 지정된 변수명에 대한 값
- 스크립트 예

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	set attr = database.getAttribute("Line Geometric", "Beam(1x1)")
3	msgbox attr.getValue("elementType")
4	msgbox attr.getValue("A")
2	'Beam(1x1)' 이름을 가진 Line Geometric 데이터셋을 attr 이라는 Object로 정의
3	'Beam(1x1)' 이름을 가진 Line Geometric 데이터셋이 적용되는 요소의 형태를 메시지창으로 출력
4	'Beam(1x1)' 이름을 가진 Line Geometric 데이터셋이 가지고 있는 단면적 값을 메시지창으로 출력

- 결과

아래와 같이 element Type은 "3D Thick Beam" 으로, 단면적값은 "1" 으로 표시됩니다.



13.4.4. getAssignments()

- 파라미터 : 없음
- 출력값 : Attribute가 적용되어져 있는 적용상태와 관련된 값들을 가지는 Objects(Assignments Objects)들을 가지는 배열
- 스크립트 예

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	Set attr = database.getAttribute("Line Geometric", "Beam(1x1)")
3	geo = attr.getAssignments()
4	num_assign = UBound(geo) + 1
5	MsgBox num_assign
6	MsgBox geo(0).getBetaAngle()
2	'Beam(1x1)' 이름을 가진 Line Geometric 데이터셋을 attr 이라는 Object로 정의
3	'Beam(1x1)' 이름을 가진 Line Geometric 데이터셋의 적용상태 관련값들을 geo 라는 변수로 출력
4	geo 배열의 최대인수값에 1 더한 값을 num_assign 변수에 저장 (= 'Beam(1x1)'이 적용된 Line 개수)
5	num_assign 값을 메시지창에 출력
6	'Beam(1x1)'기하특성이 적용된 Line들 중 첫 번째 Line에 적용 상태 중 부재의 회전각을 메시지창 출력

☞ 참고 : 6번 행의 getBetaAngle()과 같이 Assignment Object와 같이 사용될 수 있는 Method 들은 C:\LUSAS147\Programs\LPIHelp\IFAssignment.html에서 참조하실 수 있습니다.

LPI 강좌

13.4.5. 속성이 적용된 Geometry 선택

아래와 같이 현재의 속성이 적용되어 있는 Geometry를 모델러 상에서 선택되도록 할 수 있습니다.

■ 스크립트 예

행번	스크립트
1	\$ENGINE=VBSCRIPT
2	call selection.remove("All")
3	set attr = database.getAttribute("Line Geometric", "Beam(1x1)")
4	call selection.add(attr)
5	set attr = nothing
6	selectedLines = selection.getObjects("Line")
7	num_lines = Ubound(selectedLines) + 1
8	MsgBox UBound(num_lines)
2	선택된 것이 없도록 selection Object 초기화
3	'Beam(1x1)' 이름을 가진 Line Geometric 데이터셋을 attr 이라는 Object로 정의
4	'Beam(1x1)' 이름을 가진 Line Geometric 데이터셋이 적용된 것들을 선택
5	attr 변수 초기화 (메모리 절감을 위함)
6	현재 선택되어져 있는 Geometry 중 Line Objects 들을 selectedLines 배열에 저장
7	선택된 Line의 개수를 num_lines 변수에 저장
8	선택된 Line의 개수를 메시지창으로 출력