

LPI

<목 차>

1. 개요	1
1.1. Parametric Language	1
1.2. 프로그래밍 개요	1
2. 모델러 시작	2
3. Sub-Session File의 구성	3
3.1. Script 기록	3
3.2. Geometry의 구성	4
3.2.1. Surface의 정의	4
3.3. Script 기록 종료	5
4. Sub-Session File의 수정	6
4.1. 치수의 변수화	6
5. Sub-Session File 실행	6
6. Sub-Session File을 이용한 그림파일 저장	8
6.1. File open	8
6.2. Script 기록	8
6.3. 그림 저장	9
6.4. Script 기록 종료	11
7. Sub-Session File의 수정	12
7.1. 파일 이름 및 저장장소 정의	12
8. Sub-Session File 실행	13
8.1. Sub-Session File 적용 확인	14

1. 개요

For software product(s):	Standard / Lite
With product option(s):	Basic

1.1. Parametric Language

LUSAS Modeller에서 작업하는 모든 내용은 명령어 형태로 표현될 수 있습니다. 이 명령어를 활용하여 명령어 내 수치들을 변수화 시키고 프로그램 제어문 등을 추가하면 LUSAS 내부에서 수행시킬 수 있는 내부 프로그램이 되는데 이를 Parametric Language라고 합니다.

V12에서는 C++를 기초로 하였으며, V13 이후 버전에서는 Visual Basic Script를 사용할 수 있게 되었습니다.

1.2. 프로그래밍 개요

Parametric Language를 이용한 프로그래밍이라고 해서 LUSAS의 내부 명령어를 모두 알아야 할 필요는 없습니다. Sub-Session File을 사용해서 필요한 명령어들을 구성시키고 이를 일부 수정할 수 있으면 됩니다.

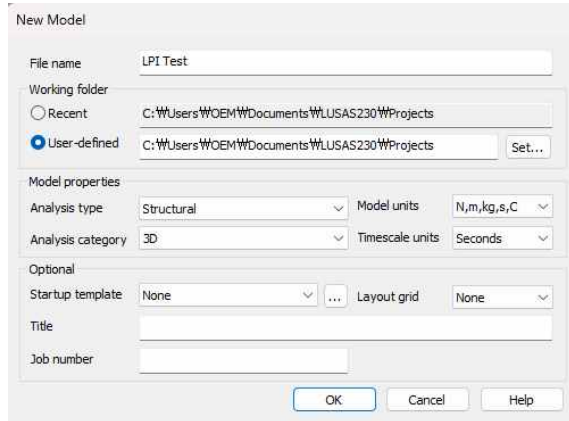
일반적인 프로그래밍 절차는 다음과 같습니다.

- 1) Modeller 시작
- 2) Sub-Session File 기록 시작
- 3) 프로그래밍 관련 필요한 작업 수행
- 4) Sub-Session File 기록 종료
- 5) 프로그래밍
 - 필요한 수치의 변수화 및 대화창 구성
- 6) 프로그램 테스트

본 예제에서는 Visual Basic Script를 사용하여 Surface를 정의하고 결과처리 중 그림파일을 자동으로 저장하도록 정의하는 간단한 프로그래밍을 이용하였습니다.

2. 모델러 시작

File> New...



🕒 File name: 파일명

🕒 Analysis type :

- 🕒 Structural - 구조해석 모델에 용이한 환경 구성
- 🕒 Thermal - 열해석 모델에 용이한 환경 구성
- 🕒 Coupled - 열해석 결과를 바탕으로 하는 구조해석, 또는 그 반대의 상호 작용 해석 환경 구성

🕒 Model units : 모델링에 사용할 단위계. 내장 DB (단면제원, 재료특성 등) 사용 시 초기값 설정에 영향

🕒 Timescale Units : 해석에 사용할 시간 단위

🕒 Analysis category : 해당 카테고리에 적합한 메뉴 항목, 대화 상자 설정 및 선택 사항들만 나타나도록 사용자 인터페이스 변경

🕒 Startup template : 자주 사용하는 데이터셋 정의. 필요시 사용자 고유의 Script를 작성하여 등록 가능

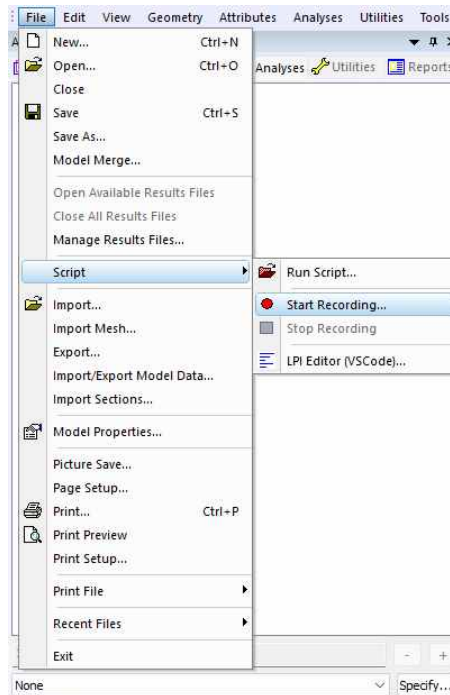
🕒 Title : 모델에 대한 설명 (생략가능)

3. Sub-Session File의 구성

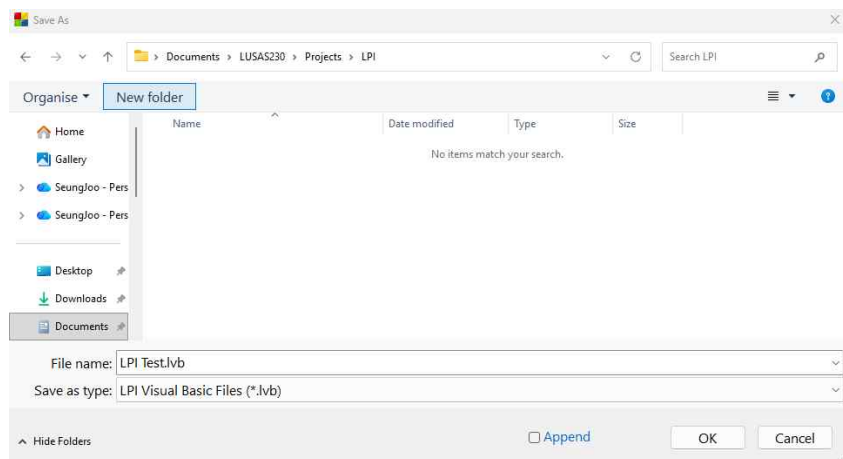
3.1. Script 기록

File> Script> Start Recording ...

⌚ 메뉴에서 Start Recording을 선택하여 이후 수행하는 작업을 Sub Session file에 기록 합니다.



⌚ Sub Session File의 이름을 지정하고, 저장 버튼을 누릅니다.



3.2. Geometry의 구성

3.2.1. Surface의 정의

Geometry> Surface> By Coords ...

⌚ 메뉴 혹은 아이콘 을 선택하고 Surface의 좌표를 다음과 같이 입력합니다.

Enter Coordinates ✕

Grid style

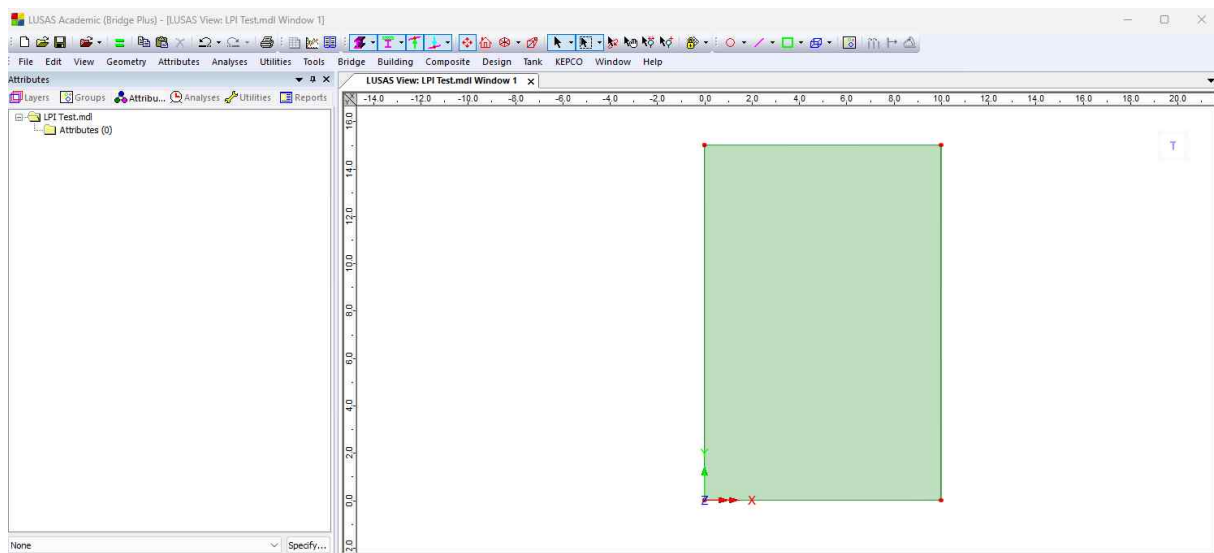
3 columns

	X	Y	Z
1	0	0	0
2	10	0	0
3	10	15	0
4	0	15	0

Local coordinate

Global coordinates ▼

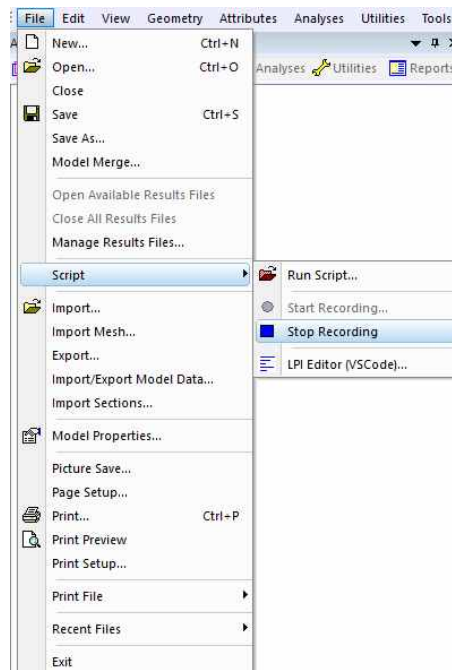
Set as active local coordinate



3.3. Script 기록 종료

File> Script> Stop Recording ...

🕒 메뉴에서 Stop Recording을 선택하여 Sub Session file에 기록하는 것을 종료합니다.



4. Sub-Session File의 수정

4.1. 치수의 변수화


- ⌚ 저장한 Sub-Session File을 열고, Surface의 X, Y방향 길이의 값을 삭제하고, X, Y의 변수로 나타냅니다. 상단에 X, Y의 값을 정의하고 저장합니다.

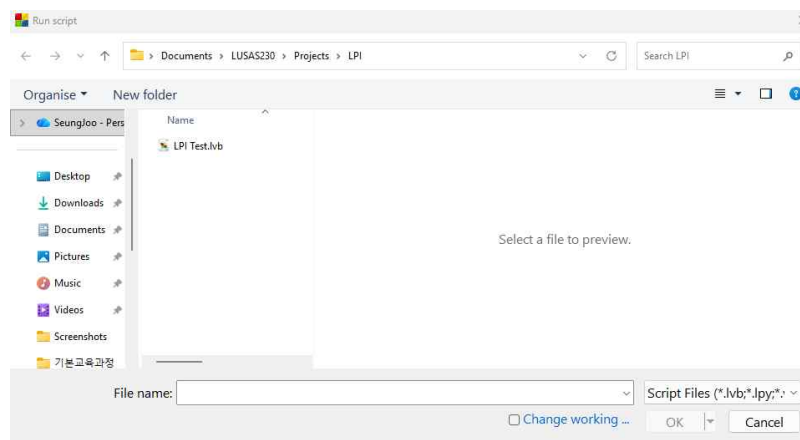
```

LPI Test.lvb X
C: > Users > OEM > Documents > LUSAS230 > Projects > LPI > LPI Test.lvb
1  $ENGINE=VBScript
2  ' LUSAS Modeller session file
3  ' Created by LUSAS 23.0-2c4 - Modeller Version 23.0.1636.56115
4  ' Created at 12:41 on Thursday, February 19 2026
5  ' (C) Finite Element Analysis Ltd 2026
6  '
7  call setCreationVersion("23.0-2c4, 23.0.1636.56115", "", "64bit unset", false)
8  '
9  x=100
10 y=100
11
12 '*** Create surface
13
14 call geometryData.setAllDefaults()
15 call geometryData.setCreateMethod("coons")
16 call geometryData.addCoords(0.0, 0.0, 0.0)
17 call geometryData.addCoords(x, 0.0, 0.0)
18 call geometryData.addCoords(x, y, 0.0)
19 call geometryData.addCoords(0.0, y, 0.0)
20 call geometryData.setLowerOrderGeometryType("coordinates")
21 call database.createSurface(geometryData)
22
  
```

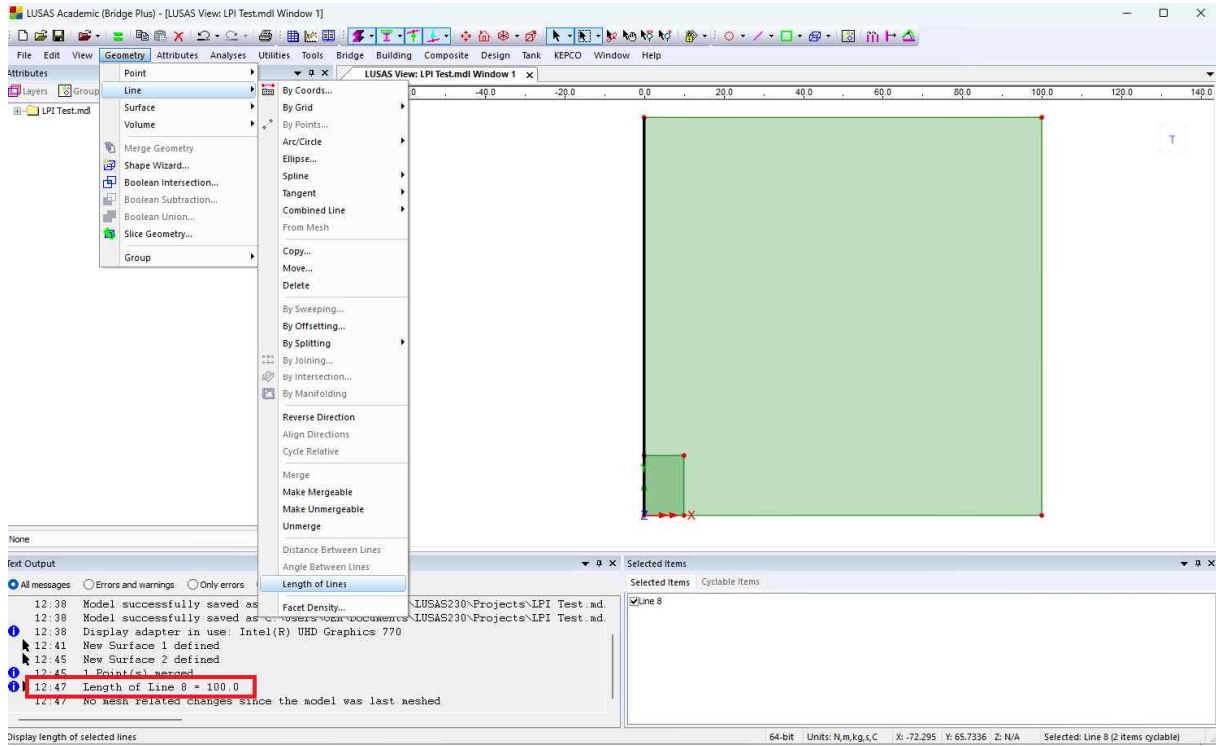
5. Sub-Session File 실행

File> Script> Run Script...

- ⌚ 메뉴 혹은  아이콘을 눌러 모델러에서 저장한 Sub-Session File을 실행합니다.



- ⑦ 변수 $x=100$, $y=100$ 의 값으로 정의한 내용이 적용되었는지 알아보기 위해 Surface의 Line하나를 선택한 후 Geometry> Line> Length of Lines를 선택하여 확인합니다. 모델러 하단의 Length of straight line 4 = 100.0을 통해 정의한 내용이 적용되었음을 확인할 수 있습니다.



6. Sub-Session File을 이용한 그림파일 저장

기본 개념과 실행 순서는 이전과 동일합니다.

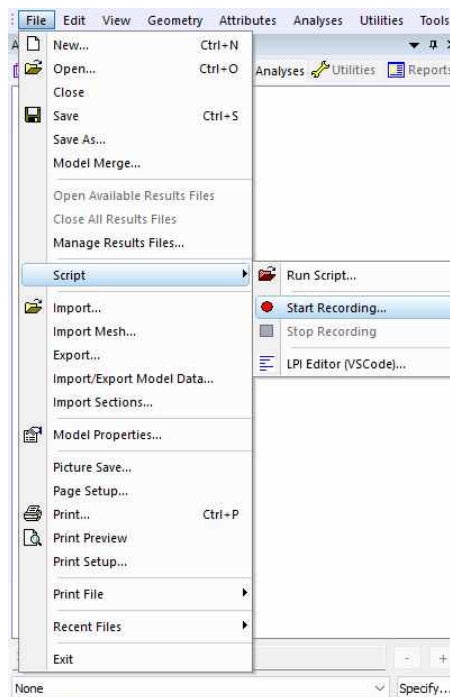
6.1. File open

⌚ 모델 파일을 Open 합니다.

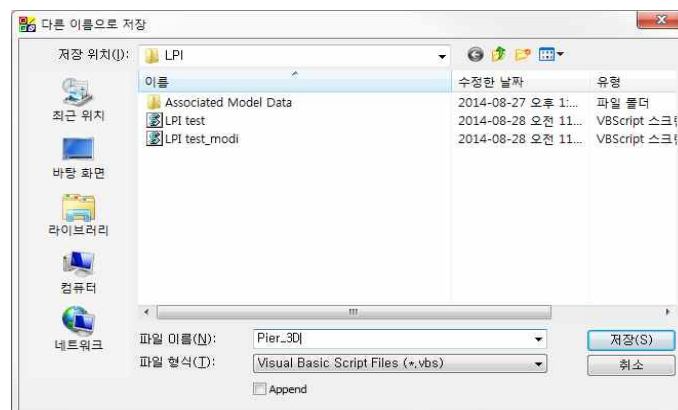
6.2. Script 기록

File> Script> Start Recording ...

⌚ 메뉴에서 Start Recording을 선택하여 이후 수행하는 작업을 Sub-Session file에 기록 합니다.



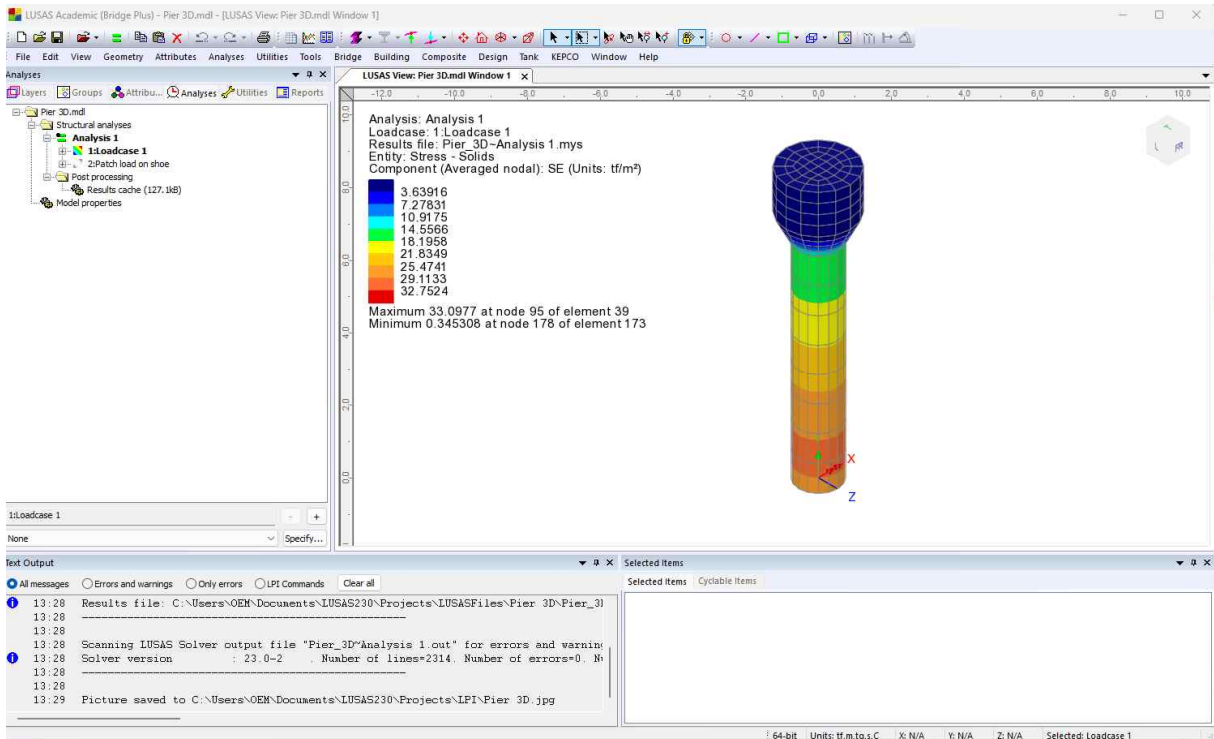
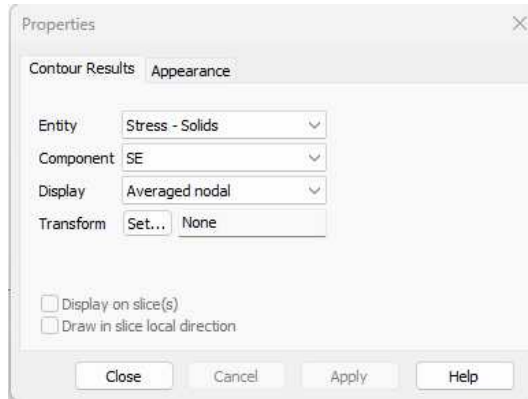
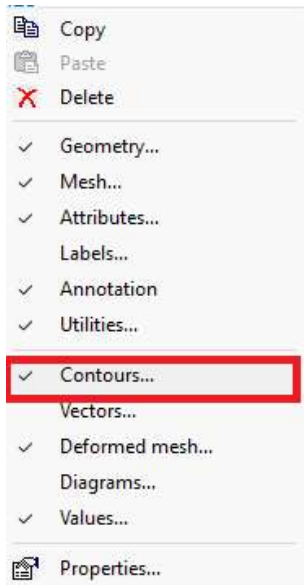
⌚ Sub-Session File의 이름을 지정하고, 저장 버튼을 누릅니다.



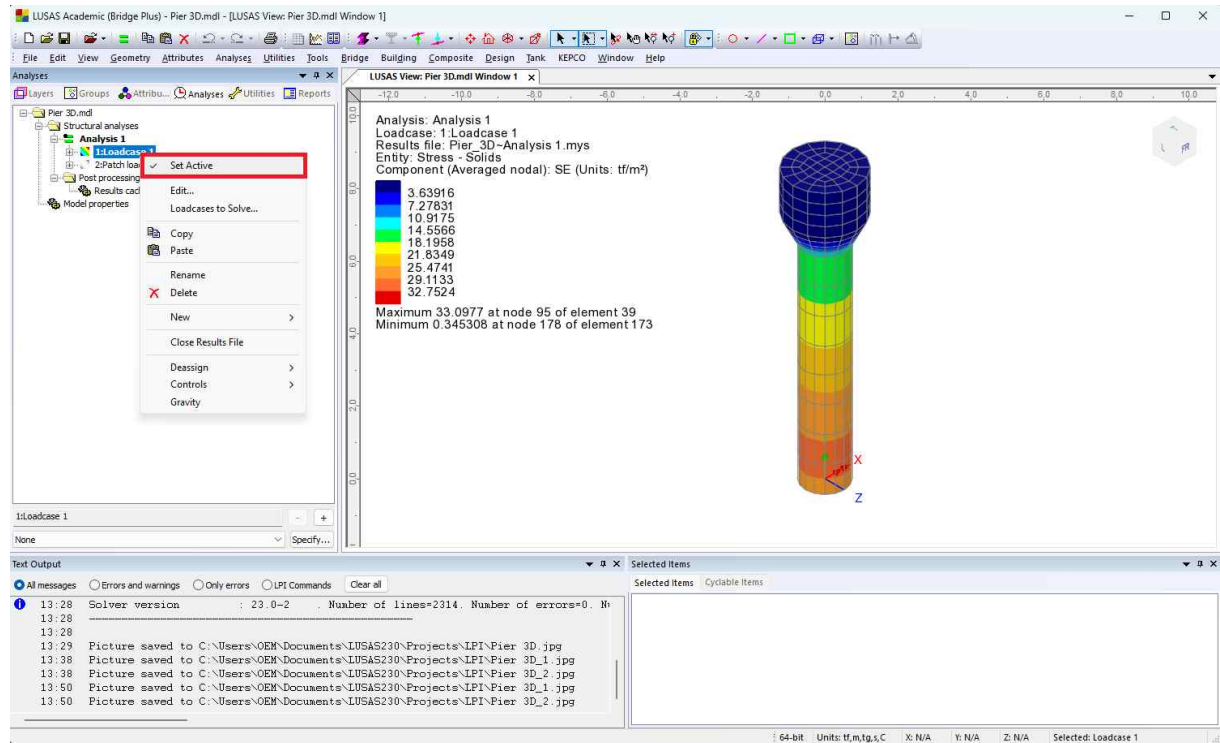
6.3. 그림 저장

File>Picture Save...

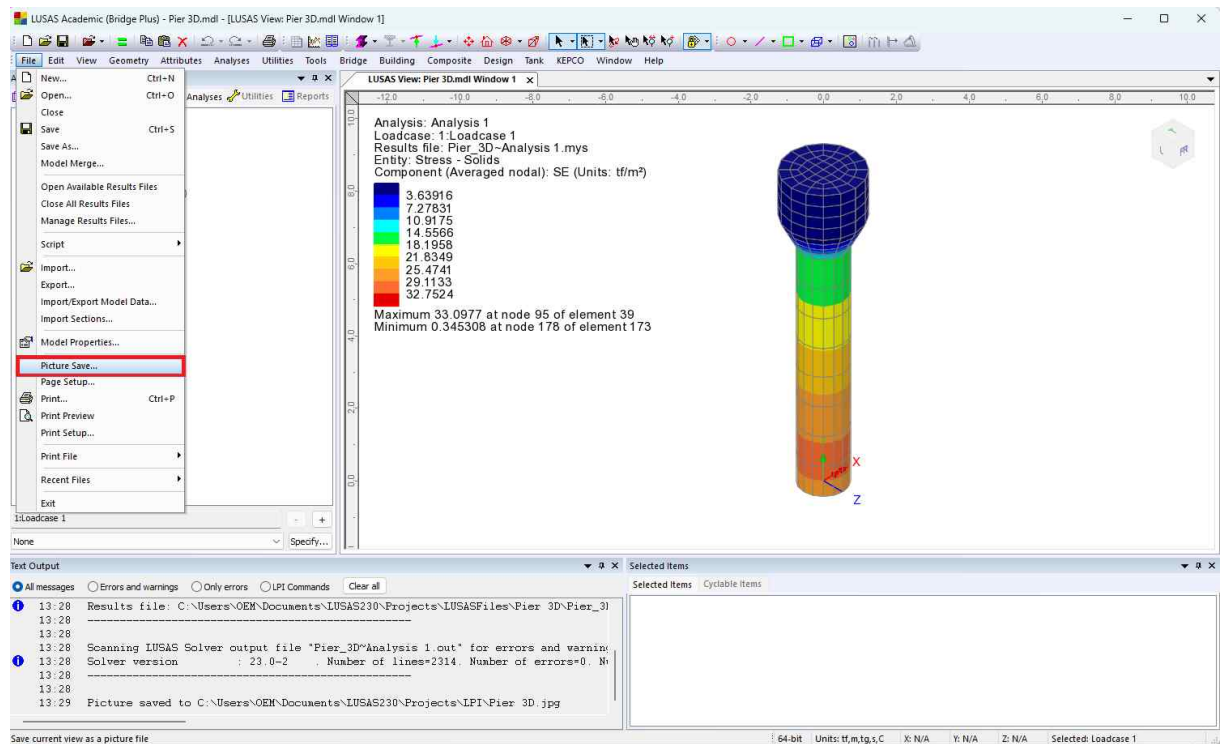
- ⌚ 그림으로 저장하고자 하는 상태를 실행하고 메뉴에서 **Picture save**를 선택합니다. 본 예제에서는 교각 입체해석에 사용된 모델의 Stress Contour를 이용하여 그림파일을 저장하였습니다.
- ⌚ **Treeview>Layer**탭에서 우측 마우스를 클릭하여 **Contour**를 선택합니다.



🕒 Treeview> Analyses탭에서 이미지를 출력할 대상 Loadcase를 Set Active 합니다.



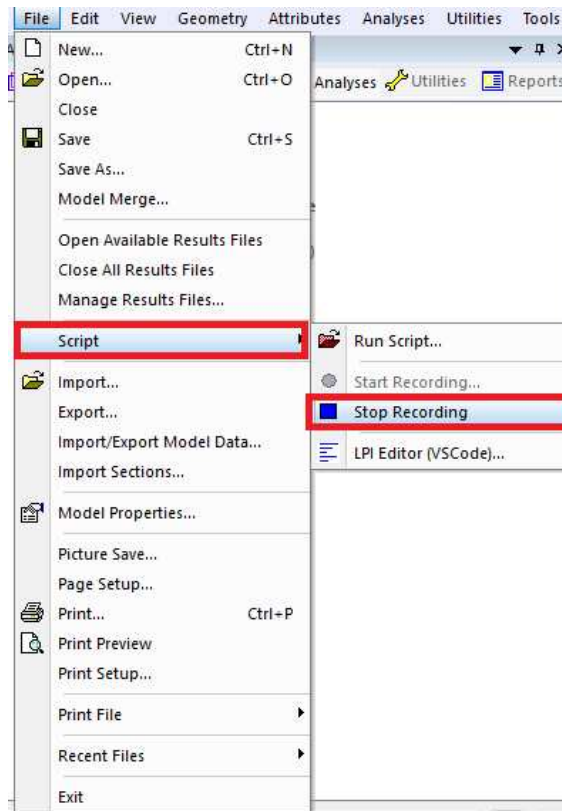
🕒 File> Picture save를 클릭한 후, 그림파일을 저장할 위치와 이름을 설정합니다.



6.4. Script 기록 종료

File> Script> Stop Recording ...

🕒 메뉴에서 Stop Recording을 선택하여 Sub Session file에 기록하는 것을 종료합니다.



7. Sub-Session File의 수정

7.1. 파일 이름 및 저장장소 정의

① 저장한 Sub-Session File을 열고, 다음의 파란색으로 표시된 부분을 수정하여 파일이름과 저장장소를 정의합니다.

```

3D Pier.lvb X
C: > Users > OEM > Documents > LUSAS230 > Projects > LPI > 3D Pier.lvb
1  $ENGINE=VBScript
2  ' LUSAS Modeller session file
3  ' Created by LUSAS 23.0-2c4 - Modeller Version 23.0.1636.56115
4  ' Created at 13:29 on Thursday, February 19 2026
5  ' (C) Finite Element Analysis Ltd 2026
6  '
7  call setCreationVersion("23.0-2c4, 23.0.1636.56115", "", "64bit unset", false)
8  '
9  '
10 '*** Settings/Options/Properties change
11 call view.insertContoursLayer()
12 call view.contours.setResultTransformNone()
13 call view.contours.setResult("Stress - Solids", "SE")
14 '
15 '*** Modify loadcase/control
16 '
17 set loadset = database.getLoadset("Loadcase 1", 0)
18 call view.setActiveLoadset(loadset)
19 set loadset = nothing
20 '
21 '*** Save picture
22 '
23 call getCurrentView().savePicture("%ModelLocn%\LPI\Pier 3D.jpg", "JPEG", 1073, 566

```

① 그림파일 명 'Pier 3D.jpg'를 순서에 따라 'Pier 3D_1.jpg'와 같은 형태로 저장되도록 하기 위한 명령어 'Pier 3D_&i&.Jpg'와 같이 설정합니다. 또한 각 Loadcase별로 Stress Contour를 그리고, 결과 그림파일을 출력하도록 하는 명령어를 설정하기 위해 Database.getLoadset("Patch load on shoe", 0)을 아래 그림과 같이 (i, 0)로 변경합니다. Save Picture상단에 변수 i값의 범위를 지정하는 For i = 1 to 2를 정의합니다. 변수 i가 1에서 2까지 진행되도록 맨 마지막 행에 Next를 반드시 지정해야 합니다.

```

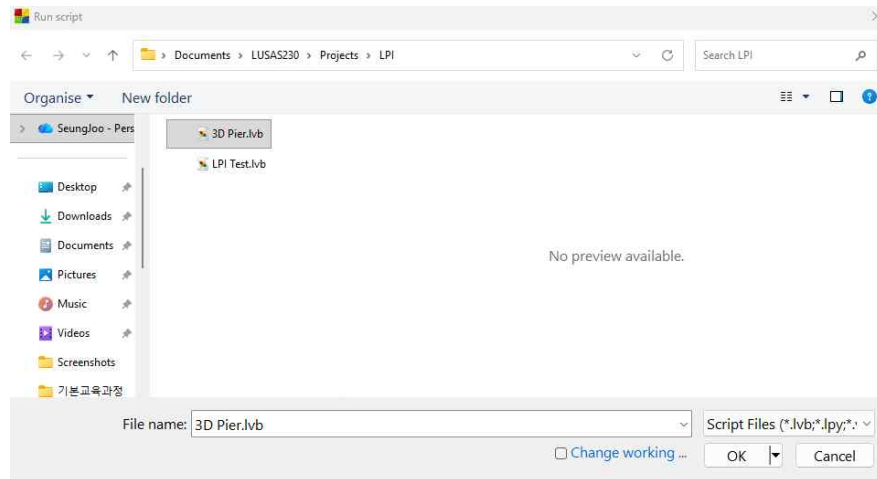
3D Pier.lvb X
C: > Users > OEM > Documents > LUSAS230 > Projects > LPI > 3D Pier.lvb
10 '*** Settings/Options/Properties change
11 call view.insertContoursLayer()
12 call view.contours.setResultTransformNone()
13 call view.contours.setResult("Stress - Solids", "SE")
14 '
15 for i = 1 to 2
16 '
17 '*** Modify loadcase/control
18 '
19 set loadset = database.getLoadset(i, 0)
20 call view.setActiveLoadset(loadset)
21 set loadset = nothing
22 '
23 '*** Save picture
24 '
25 call getCurrentView().savePicture("%ModelLocn%\LPI\Pier 3D_ "&i&".jpg", "JPEG", 1073, 566)
26 '
27 next

```

8. Sub-Session File 실행

File>Script>Run Script...

🕒 메뉴 혹은 🖱️ 아이콘을 눌러 모델러에서 저장한 Sub-Session File을 실행합니다.



8.1. Sub-Session File 적용 확인

🕒 그림파일이 저장되었는지 해당 폴더에서 확인합니다.

Analysis: Analysis 1
 Loadcase: 2: Patch load on shoe
 Results file: Pier_3D-Analysis 1.mys
 Entity: Stress - Solids
 Component (Averaged nodal): SE (Units: tf/m²)

477.677
 355.354
 533.031
 710.708
 888.385
 1.06606E3
 1.24374E3
 1.42142E3
 1.59909E3

Maximum 1.65498E3 at node 208 of element 149
 Minimum 55.8862 at node 191 of element 177

Text Output

13:28
 13:29 Picture saved to C:\Users\OEM\Documents\LUSAS230\Projects\LPI\Pier_3D.jpg
 13:38 Picture saved to C:\Users\OEM\Documents\LUSAS230\Projects\LPI\Pier_3D_1.jpg
 13:38 Picture saved to C:\Users\OEM\Documents\LUSAS230\Projects\LPI\Pier_3D_2.jpg
 13:50 Picture saved to C:\Users\OEM\Documents\LUSAS230\Projects\LPI\Pier_3D_1.jpg
 13:50 Picture saved to C:\Users\OEM\Documents\LUSAS230\Projects\LPI\Pier_3D_2.jpg
 13:58 Picture saved to C:\Users\OEM\Documents\LUSAS230\Projects\LPI\Pier_3D_1.jpg
 13:58 Picture saved to C:\Users\OEM\Documents\LUSAS230\Projects\LPI\Pier_3D_2.jpg

C:\Users\OEM\Documents > This PC > Documents > LUSAS230 > Projects > LPI

Name	Date modified	Type	Size
3D Pier.lvb	19/02/2026 1:58 pm	LUSAS Modeller V...	1 KB
LPI Test.lvb	19/02/2026 12:45 pm	LUSAS Modeller V...	1 KB
Pier 3D.jpg	19/02/2026 1:29 pm	JPG File	54 KB
Pier 3D_1.jpg	19/02/2026 1:58 pm	JPG File	54 KB
Pier 3D_2.jpg	19/02/2026 1:58 pm	JPG File	55 KB

5 items 1 item selected 53.8 KB