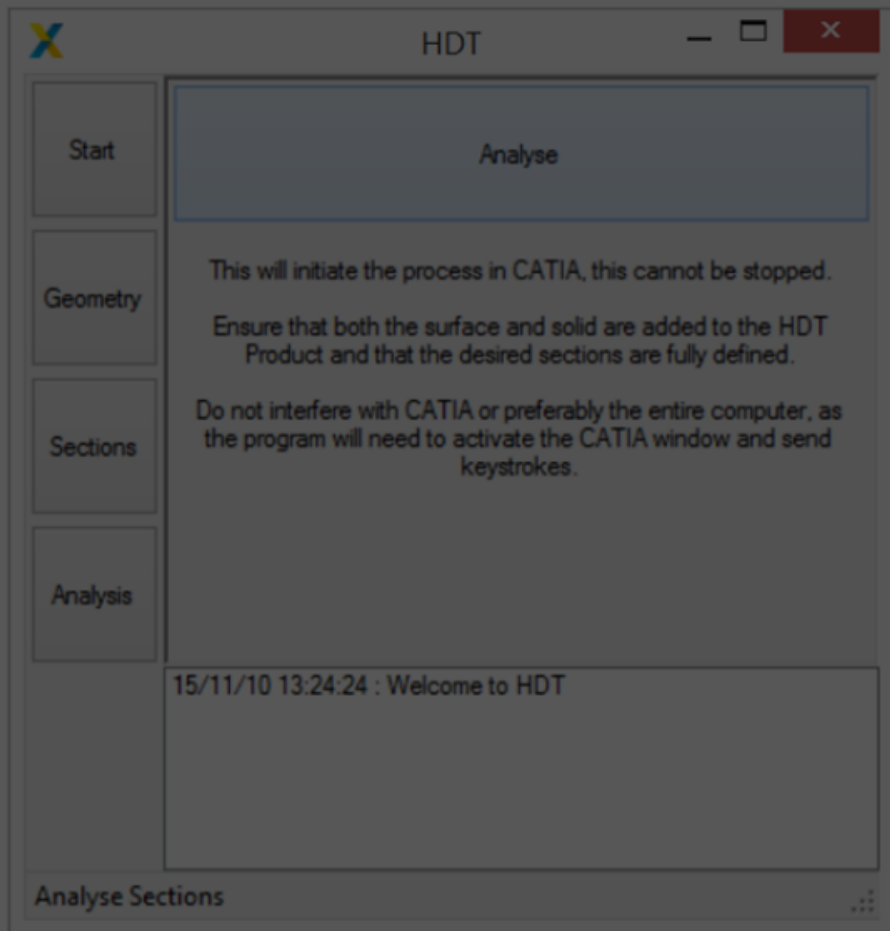
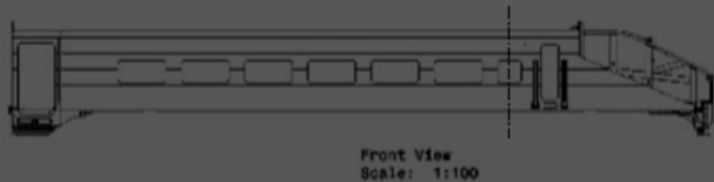


Automatic Drawings

Reason for this tutorial

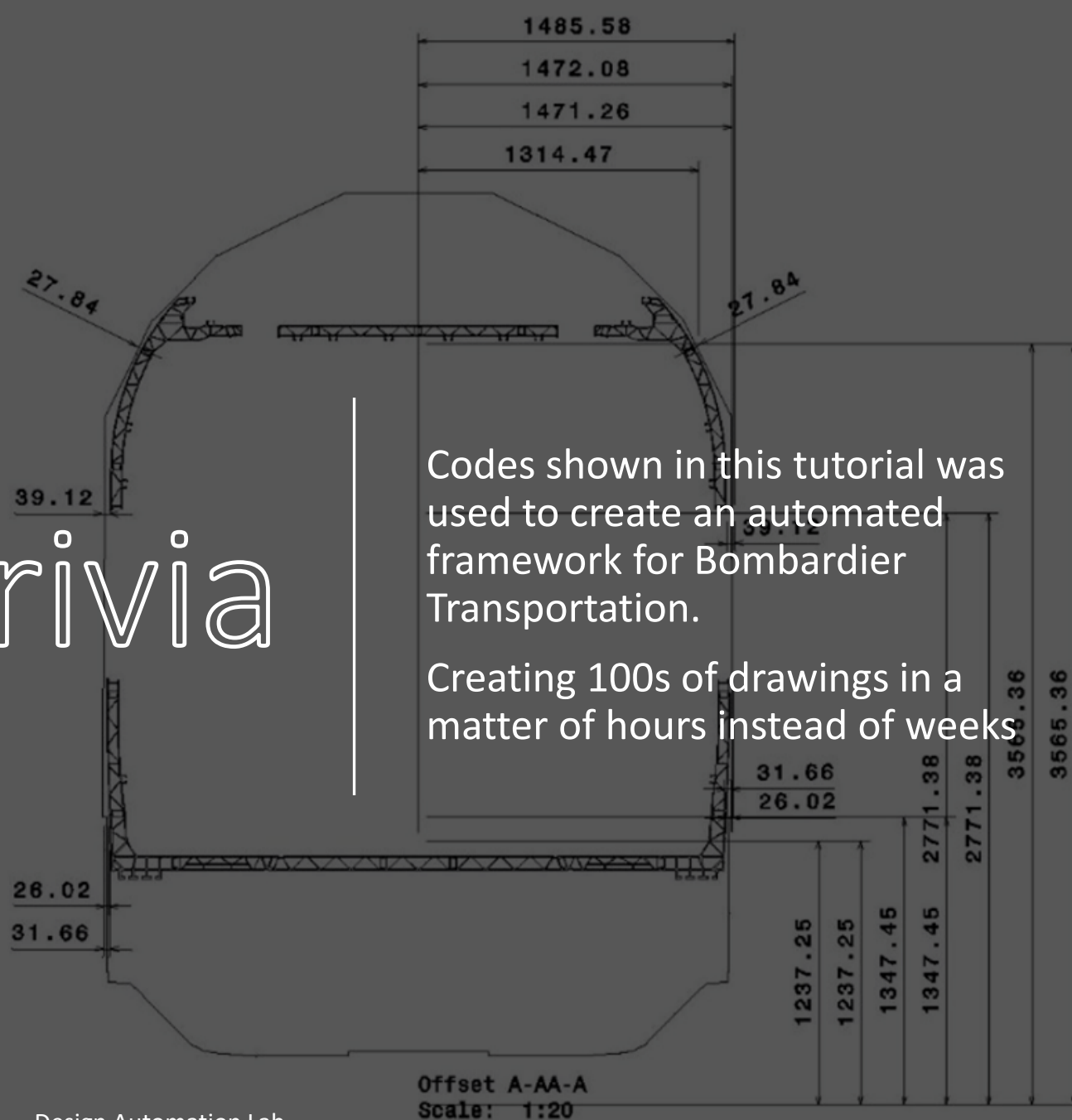
- You are creating a baseline for yourself that you will need in order to complete the group project.
- During the project you will create measurements for a much more advanced setup.
- The purpose of this tutorial is to introduce the functionalities needed to do so.
- After completing this tutorial you will need to test and re-evaluate the results with different values in order to understand how it works.



Trivia

Codes shown in this tutorial was used to create an automated framework for Bombardier Transportation.

Creating 100s of drawings in a matter of hours instead of weeks

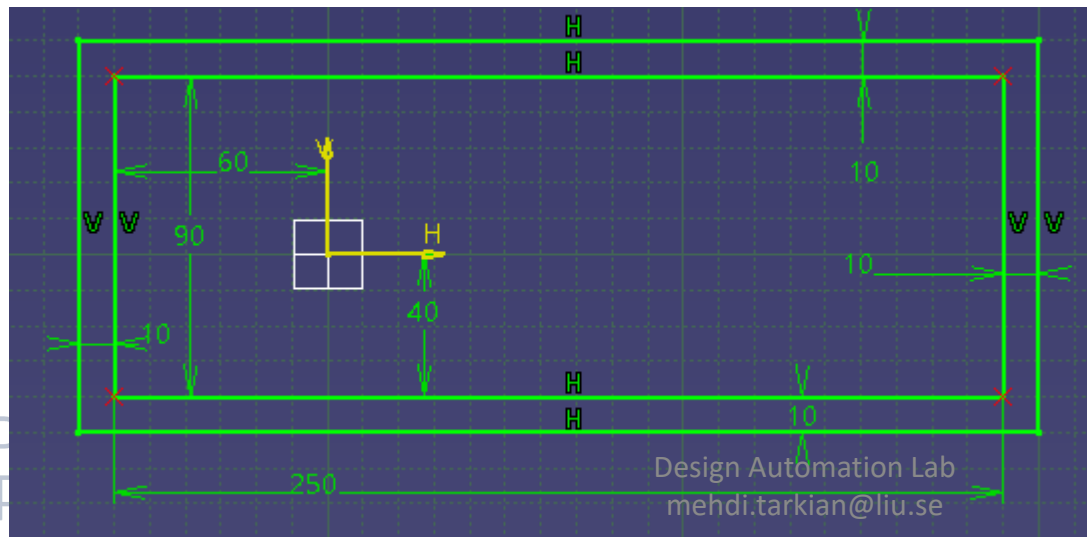
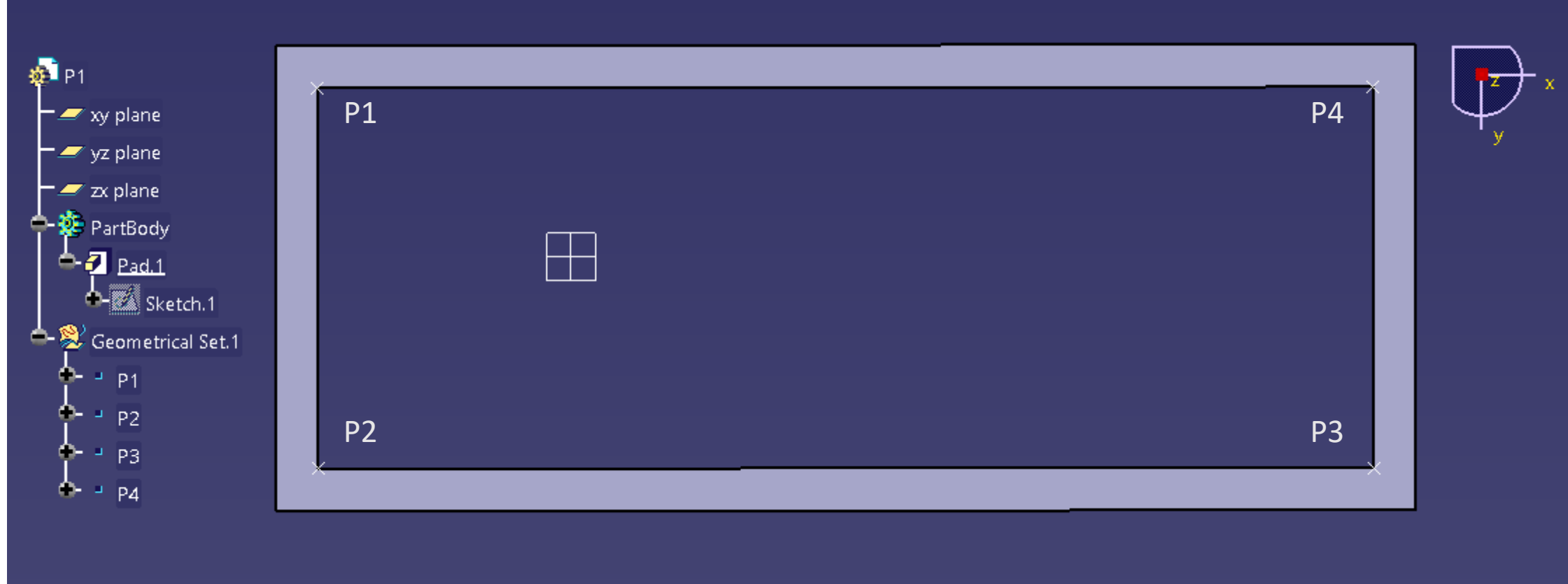


2D drawing – Class, properties and functions

- http://catiadoc.free.fr/online/interfaces/interface_DrawingDimension.htm

CATPart needed for the 2D drawing

- Next slide is not a lesson in efficient modeling or DA.
- Simply make sure you have similar measurements in the sketch so that you can compare with 2D drawing script

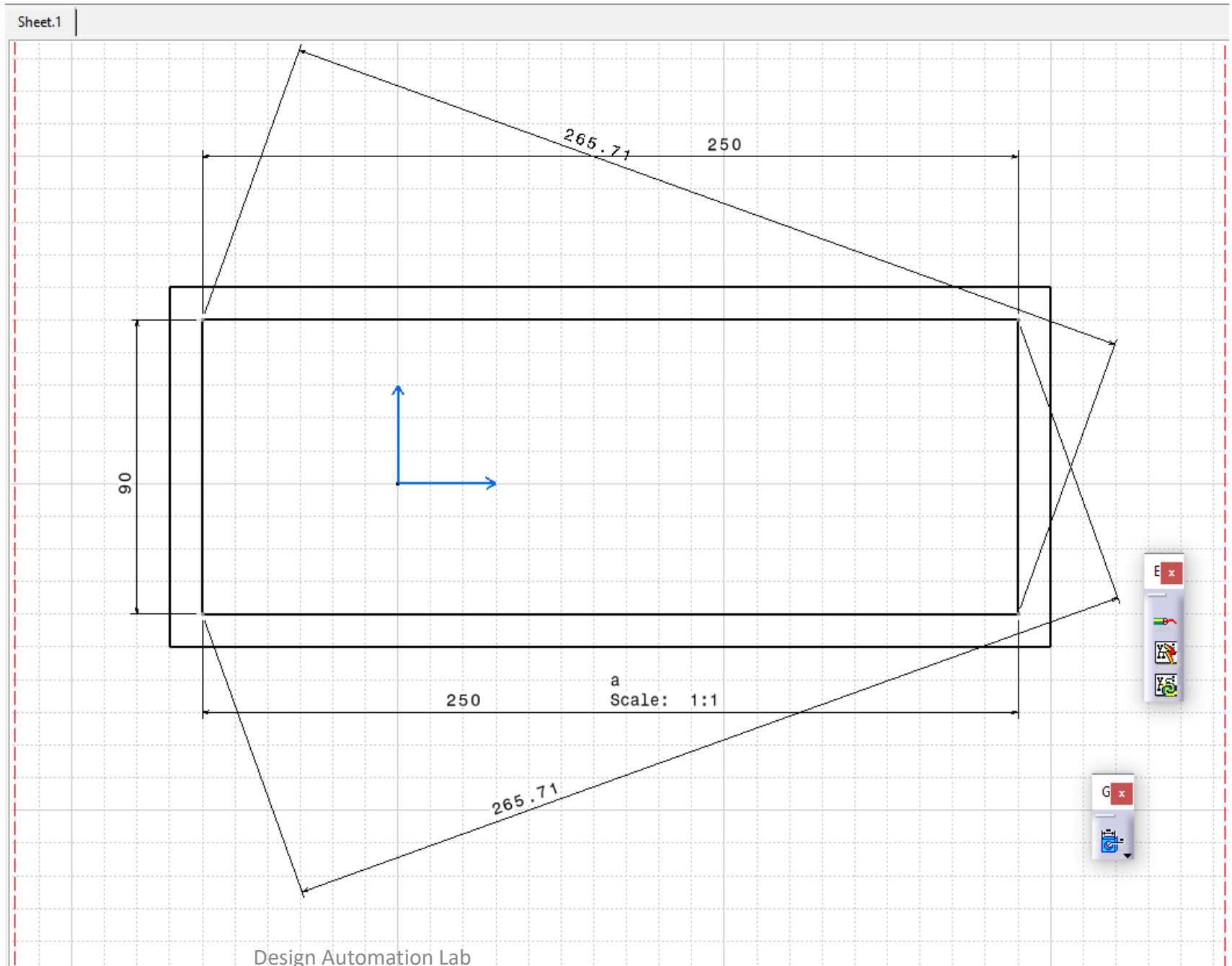


P1.CATPart

After running the script

- By assembling the code you should be able to generate a drawing resembling next slide

Results



How drawings can be automated

- The problem with drawings is that the auto generated view-elements are not named in a structured way.
- In order to create auto-measurements you need to first use the information in the 3D model and then create new 2D geometries which you re-name and can then use to create measurements on.

Setting up the drawing document

```
Set CATIA = GetObject(, "Catia.Application")
```

```
Set documents1 = CATIA.Documents
```

```
Set drawingDocument1 = documents1.Add("Drawing")
```

```
'drawingDocument1.SaveAs ("D:\LiU\Kursen\Master\TMKU01\Ex1.CATDrawing")
```

```
Set drawingSheets1 = drawingDocument1.Sheets
```

```
Set DrawingSheet1 = drawingSheets1.Item(1)
```

```
DrawingSheet1.PaperSize = DRAFTINGITF.CatPaperSize.catPaperA0
```

```
'for this line to work you need to have selected the catia refs in VBA-Tools-References
```

Adding a sheet and translation/rotation/scale

Dim oView As Object

Set oView = DrawingSheet1.Views.Add("a")

oView.Angle = 0 'Math.PI / 2

oView.X = 200

oView.Y = 200

oView.Scale2 = 1

oView.Activate

Setting up an xy-view

```
Set oParentDoc = documents1.Item("P1.CATPart")
```

```
Dim oGen As DrawingViewGenerativeBehavior
```

```
Set oGen = oView.GenerativeBehavior
```

```
oGen.Document = oParentDoc.Product
```

```
oView.GenerativeBehavior.DefineFrontView 1#, 0#, 0#, 0#, 1#, 0#
```

Various other views

' "xy"

'drawingViewGenerativeBehavior1.DefineFrontView 1#, 0#, 0#, 0#, 1#, 0#

' "-xy"

'drawingViewGenerativeBehavior1.DefineFrontView -1#, 0#, 0#, 0#, 1#, 0#

""yz"

'drawingViewGenerativeBehavior1.DefineFrontView 0#, 1#, 0#, 0#, 0#, 1#

' "-yz"

'drawingViewGenerativeBehavior1.DefineFrontView 0#, -1#, 0#, 0#, 0#, 1#

' "zx"

'drawingViewGenerativeBehavior1.DefineFrontView 1#, 0#, 0#, 0#, 0#, 1#

' "-zx"

'drawingViewGenerativeBehavior1.DefineFrontView -1#, 0#, 0#, 0#, 0#, 1#

Creating points on a drawing

Dim point2D1 As Object

Set point2D1 = hybridShapes1.Item("point1")

Dim Point3D1 As Object

Set Point3D1 = point2D1

Dim point2D2 As Object

Set point2D2 = hybridShapes1.Item("point2")

Dim Point3D2 As Object

Set Point3D2 = point2D2

Dim Coords1(2)

Point3D1.GetCoordinates Coords1

Dim Coords2(2)

Point3D2.GetCoordinates Coords2

Set selection1 = drawingDocument1.Selection

Set DrwView = DrawingSheet1.Views.Item(sheet_name) 'ActiveView

' sheet_name should contain the name of the sheet in string value

DrwView.Activate

Set Fact2D = DrwView.Factory2D

Set point1 = Fact2D.CreatePoint(Coords1(0), Coords1(1))

Set point2 = Fact2D.CreatePoint(Coords2(0), Coords2(1))

Creating a line between the points

```
Dim Line1 As Line2D
```

```
Set Line1 = Nothing
```

```
Set Line1 = Fact2D.CreateLine(Coords1(0), Coords1(1), Coords2(0), Coords2(1)) ' Along this direction
```

Setting up the measurements

```
Dim iType As CatDimType  
Dim catDimDistance As Variant  
iType = catDimDistance
```

```
Dim myElements(1)  
Set myElements(0) = point1  
Set myElements(1) = point2  
Dim selpoints(3)  
selpoints(0) = 0  
selpoints(1) = 0  
selpoints(2) = 0  
selpoints(3) = 0
```


Creating the measurements

Dim MyDimension As DrawingDimension

Set MyDimension = DrwView.Dimensions.Add2(iType, myElements, selPoints, Line1, 0)

MyDimension.MoveValue move_dim_x, move_dim_y, SubPart, rotate_dim

'move_dim_x and move_dim_y and rotate_dim should have float values

Removing the un-necessary geomtries

Set GeomElems = DrwView.GeometricElements

Set LineD = GeomElems.Item(Line1.Name)

selection1.Clear

selection1.Add (LineD)

selection1.Delete

selection1.Clear

Creating a table in the 2D drawing

Part Number	7533889- /89-
Prod Revision	-
Definition	GATE PART LOWER RI/LE
English Subtitle	Wldr
Swedish Title	GRINDDEL UNDRE HÖ/VÄ
Swedish Subtitle	Svr
Designed By	AR
Originated Date	2020-10-26
Last Preliminary Revision	1
Machined Surface Code	
Welding Code	A
Casting Code	CT
Surface Finish Tolerance	
Surface Treatment Standard	BT57-02-035
Drawing Color	001/BT57-00-001
Material	
Model Purpose	
Drawing Letter	C
Scale	1:2
Total Number Of Pages	2
Drawing Remark	DXF: 7533889- /89--1.dxf
Description	
Mass	
Drawing Mass	
Revision	1

Creating a table (1/3)

Set myTables = oView.tables

'select a range in excel. For example something like the pic to the right

Set mySheet = ActiveSheet

Set myrange = Selection

Bredd	1000	Svr Ram och Stolpar	
7533851-2510	2500	7533852-2510	
7533851-2410	2400	7533852-2410	
7533851-2310	2300	7533852-2310	
7533851-2210	2200	7533852-2210	
7533851-2110	2100	7533852-2110	
7533851-2010	2000	7533852-2010	
7533851-1910	1900	7533852-1910	
7533851-1810	1800	7533852-1810	
7533851-1710	1700	7533852-1710	
7533851-1610	1600	7533852-1610	
7533851-1510	1500	7533852-1510	
7533851-1410	1400	7533852-1410	
7533851-1310	1300	7533852-1310	
7533851-1210	1200	7533852-1210	
7533851-1110	1100	7533852-1110	
7533851-1010	1000	7533852-1010	
Art.nr (-läbr)	L	A	

Creating a table (2/3)

```
nRows = myrange.Rows.Count
```

```
nCols = myrange.Columns.Count
```

```
cellW = 50
```

```
cellH = 15
```

```
fSize = 5
```

```
'Set myTable = myTables.Add(At x1mm from the sheet origin along x, At y1 from the sheet origin along y, numb  
of Rows, numb of Cols, cell Height, cell Width)
```

```
Set myTable = myTables.Add(500, 500, nRows, nCols, cellH, cellW)
```

Creating a table (3/3)

'set up a double loop in order to loop through the value to be sent to table with i and j.

Call myTable.GetCellObject(i, j).SetFontSize(0, 0, fSize)

Call myTable.SetCellAlignment(i, j, 5) 'CatTableMiddleCenter = 5

Call myTable.SetCellString(i, j, myStr) 'myStr=the text to be sent