//These are Rhino aliases/scripts that I find useful for productivity.
//Remember to save Python- and RVB-scripts to your own folder name as indicated!

//Alias: InFastener
//Tooltip: Draw Inboard/X fastener polyline to create centroid
//Active layer: appropriate fastener layer (iron nail, iron bolt, treenail)

```
! _polylineOnMesh
_Pause
_Multipause
_SetRedrawOff
-_Patch
_Pause
_PointSpacing=1
_USpans=30
_VSpans=30
_Enter
_SelLast
_AreaCentroid
_Pause
_Enter
_Delete
-_LayerBook
_N
_S
_Enter
_AllLayersOn
_SelLast
_ChangeToCurrentLayer
-_LayerBook
_P
_S
_Enter
_AllLayersOn
_SetRedrawOn
```

//Alias: OutFastener
//Tooltip: Draw Outboard/O fastener polyline to create centroid
//Active layer: appropriate fastener layer (iron nail, iron bolt, treenail)

```
! _polylineOnMesh
_Pause
_Multipause
_SetRedrawOff
-_Patch
_Pause
_PointSpacing=1
_USpans=30
_VSpans=30
_Enter
_SelLast
_AreaCentroid
```

_Pause
_Enter
_Delete
-_LayerBook
_N
_N
_S
_Enter
_AllLayersOn
_SelLast
_ChangeToCurrentLayer
-_LayerBook
_P
_P
_S
_Enter
_AllLayersOn
_SetRedrawOn

//Alias: Axis_12
//Tooltip: Point-snap to draw axis and Ã˜12 iron nail cylinder
//Active layer: appropriate fastener AXIS layer (iron nail - axis)
//Uses a Python script (below), the folder location of which must match the script text exactly.

!_line
_pause
_pause
_selLast
_-runPythonScript "C:\Users\[USER]\AppData\Roaming\McNeel\Rhinoceros\
[VER.NO]\scripts\ExtendLInesByLengthFromMidPt.py"
_pipe
_pause
6
_Enter
_Enter
_Enter
_selLast
_-runPythonScript "C:\Users\[USER]\AppData\Roaming\McNeel\Rhinoceros\
[VER.NO]\scripts\ChangeObjsToHiddenLayer.py"

//Alias: Axis_30
//Tooltip: Point-snap to draw axis and Ã˜30 treenail/bolt cylinder
//Active layer: appropriate fastener AXIS layer (iron bolt -, treenail - axis)
//Uses a Python script (below), the folder location of which must match the script text exactly.

!_line
_pause
_pause
_selLast
_-runPythonScript "C:\Users\[USER]\AppData\Roaming\McNeel\Rhinoceros\
[VER.NO]\scripts\ExtendLInesByLengthFromMidPt.py"
_pipe
_pause

15
_Enter
_Enter
_Enter
_selLast
_-runPythonScript "C:\Users\[USER]\AppData\Roaming\McNeel\Rhinoceros\
[VER.NO]\scripts\ChangeObjsToHiddenLayer.py"

//Alias: CutCyl_12
//line-snap to draw Ã˜12 iron nail cylinder only
//use on existing fastener axes (e.g. from before Axis_12 was created)
//Active layer: appropriate fastener AXIS layer (iron nail - axis)
//Uses a Python script (below), the folder location of which must match the script text exactly.

_pipe
_pause
6
_Enter
_Enter
_Enter
_selLast
_-runPythonScript "C:\Users\[USER]\AppData\Roaming\McNeel\Rhinoceros\
[VER.NO]\scripts\ChangeObjsToHiddenLayer.py"

//Alias: CutCyl_30
//line-snap to draw Ã˜30 treenail/bolt cylinder only
//use on existing fastener axes (e.g. from before Axis_30 was created)
//Active layer: appropriate fastener AXIS layer (iron nail - axis)
//Uses a Python script (below), the folder location of which must match the script text exactly.

_pipe
_pause
15
_Enter
_Enter
_Enter
_selLast
_-runPythonScript "C:\Users\[USER]\AppData\Roaming\McNeel\Rhinoceros\
[VER.NO]\scripts\ChangeObjsToHiddenLayer.py"

//Alias: MinBoundingBox
//run script to create a true minimum (not native Rhino's axis-true) bounding box. Useful for storage volume calculations etc.
//Script made by Thomas Anagnostou and to be found here: https://www.food4rhino.com/en/resource/minimum-bounding-box - place in your desired Rhino script folder, typically: C:\Users\
[USER]\AppData\Roaming\McNeel\Rhinoceros\[VER.NO]\scripts
//Active layer: Other or appropriate custom layer
//Uses an RVB script, the folder location of which must match the script text exactly.

'_RunScript "MinBoundBox3d.rvb"

```
//Alias: ForPrint
//Finalise for 3D-printing by scaling to 1:10, rotating to Y-direction, and moving to Origo
//Check that Cplane is on Inboard Face or as desired

!_Scale _Enter 0.1 _Enter _-runPythonScript
"C:\Users\mht\AppData\Roaming\McNeel\Rhinoceros\7.0\scripts\Rotate90AroundWBBCtrZAxis.py" _Move
_Enter 0,0,0 _Enter _Export



//PYTHON:



//ExtendLInesByLengthFromMidPt.py
//By Pascal Golay - https://discourse.mcneel.com/t/extending-line-at-both-ends/158993/4
//Modified here so that extension length is hard-coded to 50

import rhinoscriptsyntax as rs
import scriptcontext as sc
import Rhino



def test():

    def line_filter(rhino_object, geometry, component_index):

        if isinstance(geometry, Rhino.Geometry.LineCurve):
            return True
        if geometry.IsLinear():
            return True

        return False

    while True:
        xLength = 50
        if "XLENGTH" in sc.sticky:
            xLength = sc.sticky["XLENGTH"]

        go = Rhino.Input.Custom.GetObject()
        go.SetCommandPrompt("Select lines to extend")

        go.GeometryFilter = Rhino.DocObjects.ObjectType.Curve
        go.SetCustomGeometryFilter(line_filter)

        go.AcceptNumber(True,False)

        opLen = Rhino.Input.Custom.OptionDouble(xLength)
        go.AddOptionDouble("ExtensionLength", opLen)

        rc = go.GetMultiple(1,0)
        if go.CommandResult()!=Rhino.Commands.Result.Success:
            return go.CommandResult()

        if rc==Rhino.Input.GetResult.Object:
```

```
            ids = [go.Object(i).ObjectId for i in range(go.ObjectCount)]
            break

        elif rc==Rhino.Input.GetResult.Number:
            xLength = go.Number()
            sc.sticky["XLENGTH"]= xLength
            continue
    count = 0
    for id in ids:
        geo= rs.coerceline(id)
        geo.Extend(xLength, xLength)
        sc.doc.Objects.Replace(id, geo)
        count += 1
    print(str(count) + " lines extended at each end by " + str(xLength))
    sc.doc.Views.Redraw()
test()



//ChangeObjsToHiddenLayer.py
//Based on script of the same name by Mitch 'Helvetosaur' (and should therefore relly be renamed, sorry) -
https://discourse.mcneel.com/t/changing-layer-layer-name-in-command/36926
//Modified here to set the desired target Layer name - therefore TIED TO THE VIR RHINO TEMPLATES - change
content of double quotes to your own (sub)layer names!

# coding=utf-8

import rhinoscriptsyntax as rs

def ChangeObjsToHiddenLayer():
    # find the name of the first layer in the Layer book:
    layers = rs.LayerNames(True)
    # create the name of the desired layer (everything in the double quotes is per the VIR Rhino templates, change as
desired):
    layer=layers[0]+"::SKÃ†REOBJEKTER::Model-sammenfÃ¸jninger"
    # rs.TextOut(layer)
    objs =rs.GetObjects("Select objects to change to hidden layer",preselect=True)
    # if the layer doesn't exist, then create it; place the object in this layer
    if objs:
        rs.EnableRedraw(False)
        if not rs.IsLayer(layer): rs.AddLayer(layer,visible=False)
        for obj in objs: rs.ObjectLayer(obj,layer)
ChangeObjsToHiddenLayer()



//Rotate90AroundWBBCtrZAxis.py
//Script by Mitch Heynick 09.06.2
//Rotates object 90 deg around World Z-axis

"""
Rotates selected objects a fixed degree value around the specified *world* axis.
Provide angle in degrees and axis vector as input arguments at bottom.
Script by Mitch Heynick 09.06.23
"""

import rhinoscriptsyntax as rs
```

```
import Rhino

def RotateAroundWBBCtrAxis(deg,axis):
    msg="Select objects to rotate {} deg.".format(deg)
    obj_ids=rs.GetObjects(msg,preselect=True)
    if obj_ids:
        rs.EnableRedraw(False)
        bb=rs.BoundingBox(obj_ids)
        if bb:
            xform=rs.XformRotation2(deg,axis,(bb[0]+bb[6])/2)
            rs.TransformObject(obj_ids,xform)

RotateAroundWBBCtrAxis(90,Rhino.Geometry.Vector3d.ZAxis)
```