

I. Thư viện AutoCad

Cài đặt thư viện làm việc với AutoCad: **pip install --upgrade pyautocad**

```
C:\Windows\system32>pip install --upgrade pyautocad
Collecting pyautocad
  Downloading pyautocad-0.2.0.zip (45 kB)
    ----- 45.6/45.6 KB 324.2 kB/s eta 0:00:00
  Preparing metadata (setup.py) ... done
Collecting comtypes>=1.1.1
  Downloading comtypes-1.1.11-py2.py3-none-any.whl (167 kB)
    ----- 167.4/167.4 KB 456.8 kB/s eta 0:00:00
Using legacy 'setup.py install' for pyautocad, since package 'wheel' is not installed.
Installing collected packages: comtypes, pyautocad
  Running setup.py install for pyautocad ... done
Successfully installed comtypes-1.1.11 pyautocad-0.2.0
WARNING: You are using pip version 22.0.4; however, version 22.1.1 is available.
You should consider upgrading via the 'C:\Users\PKH_SGTVT2\AppData\Local\Programs\Python\Python310\python.exe -m pip install --upgrade pip' command.
C:\Windows\system32>
```

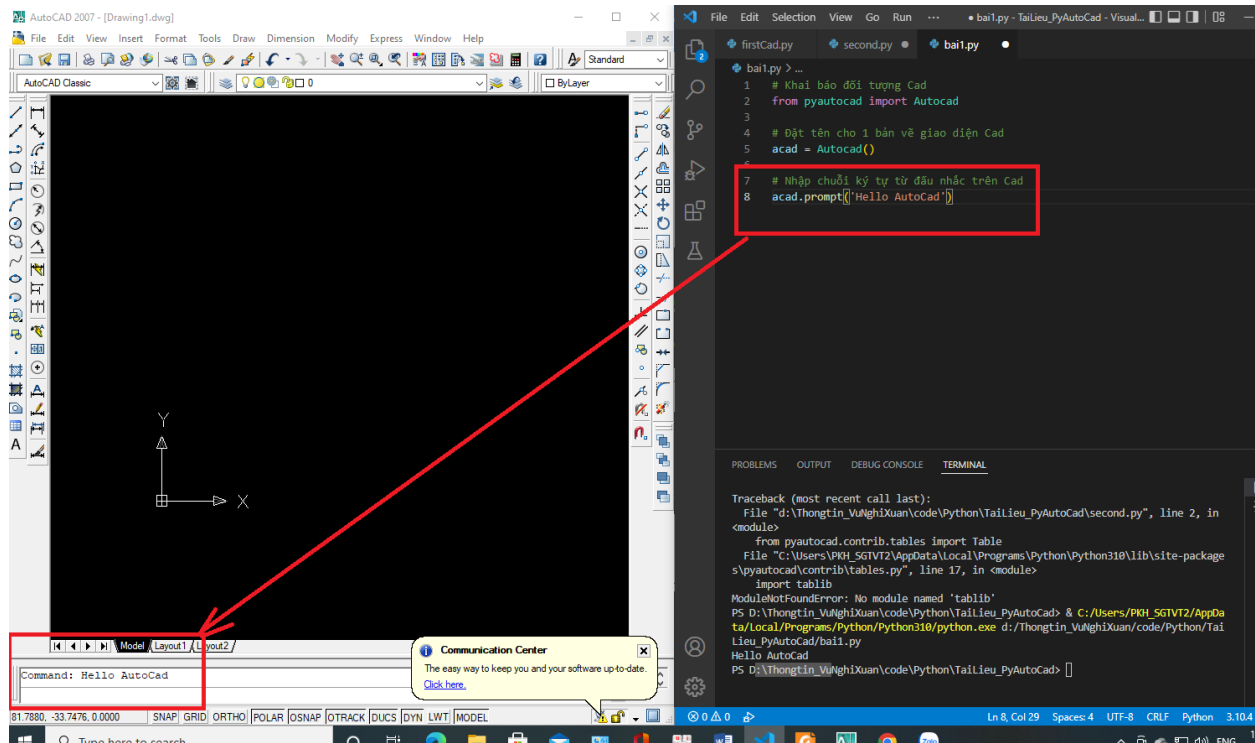
➤ **Cách khai báo đối tượng bản vẽ Cad và nhập chuỗi vào đầu nhắc bản vẽ Cad:**

```
# Khai báo đối tượng Cad
from pyautocad import Autocad

# Đặt tên cho 1 bản vẽ giao diện Cad và mở mới
acad = Autocad()

# Nhập chuỗi ký tự từ đầu nhắc trên Cad
acad.prompt('Hello AutoCad')
```

Đây là kết quả sau khi chạy code:



Tiếp theo hãy khảo sát một số đoạn code để lấy được tên bản vẽ, tên đường dẫn,...hãy tham khảo đoạn code sau:

```
# Khai báo đối tượng Cad
from pyautocad import Autocad

"1. Đặt tên (viết tắt) 01 bản vẽ Cad với tên là: acad";
acad = Autocad(create_if_not_exists = True)

"2. Nhập chuỗi ký tự từ đầu nhắc trên Cad";
# acad.prompt('Hello AutoCad')

"3. Cách xem tên bản vẽ";
nameDrw = acad.doc.name
print(f"Tên bản vẽ là: {acad.doc.name}")

"4. Cách xem tên bản vẽ";
print(f"Tên đường dẫn đầy đủ: {acad.doc.fullname}")

"5. Điều chỉnh phóng đại các đối tượng bằng ZoomAll";
acad.app.ZoomAll()
```

```
"6. Điều chỉnh đối tượng bằng ZoomExtents";  
acad.app.ZoomExtents()
```

➤ **Làm quen với cách tạo điểm, thay đổi tọa độ, tính khoảng cách 2 điểm:**

```
# Khai báo đối tượng Cad  
from pyautocad import Autocad  
from pyautocad import APoint  
  
"1. Đặt tên (viết tắt) 01 bản vẽ Cad với tên là: acad";  
acad = Autocad(create_if_not_exists = True)  
  
"2. Tạo điểm";  
P1 = APoint(0,0,0) # Có thể khai báo P1 = APoint(0,0)  
print(P1)  
">>>APoint(0.00, 0.00, 0.00)"  
  
"3. Thay đổi vị trí, tọa độ P1, với x=5, y=2, z=0";  
P1 = APoint(5,2,0)  
print(P1)  
">>>APoint(5.00, 2.00, 0.00)"  
  
"4. Lấy tọa độ của điểm P1";  
print(f"x= {P1.x}, y= {P1.y}, z= {P1.z}")  
">>>x= 5.0, y= 2.0, z= 0.0"  
  
"5. Lấy khoảng cách 2 điểm P1 & P2";  
P2 = APoint(0,0,0)  
print(f"Khoảng cách P1-P2 là: {P1.distance_to(P2)}")  
">>>Khoảng cách P1-P2 là: 5.385164807134504"  
  
"6. Đưa vị trí P1 về tọa độ ";  
P1 = APoint(0)  
print(P1)  
">>>APoint(0.00, 0.00, 0.00)"  
  
"7. Cộng 1 đơn vị cho điểm P2";  
P2 = P2 + 1 # Tương đương P2 +=1  
print(P2)  
">>>APoint(1.00, 1.00, 1.00)"  
  
"8. Trừ 1 đơn vị cho điểm P2";
```

```

P2 = P2 - 1 # Tương đương P2 -=1
print(P2)
">>>APoint(0.00, 0.00, 0.00)"

"9. Trừ 1 đơn vị cho điểm P2";
P2 = P2 - 1 # Tương đương P2 -=1
print(P2)
• ">>>APoint(0.00, 0.00, 0.00)"

```

➤ **Vẽ đường thẳng, đường tròn, thay đổi màu và đặt layer cho các đối tượng đường vẽ:**

```

"""Vì Google mãi mà chưa tìm ra cách để mở 1 file Cad đã lưu nên tạm thời sử dụng
đoạn code dưới đây"""
"-----"
import comtypes.client
import time
try:
    acad = comtypes.client.GetActiveObject("AutoCAD.Application")
except:
    acad = comtypes.client.CreateObject("AutoCAD.Application")
while not acad.GetAcadState().IsQuiescent :
    time.sleep(5)
acad.Visible = True
doc =
acad.Documents.Open("D:\\Thongtin_VuNghiXuan\\code\\Python\\TaiLieu_PyAutoCad\\Ba
i1_ObjCad.dwg")
"-----"

from pyautocad import Autocad, APoint
acad = Autocad(create_if_not_exists=True)
p1= APoint(10)
p2= APoint(20)

"1. Vẽ đường thẳng từ 2 điểm"
acad.model.AddLine(p1, p2)

"2. Vẽ đường thẳng từ 2 điểm"
l12 = acad.model.AddLine(p1, p2)

"3. Di chuyển đường l12 lùi về cách điểm P1 là : 5 đơn vị"
l12.Move(p1, APoint(-5))
print(p1, p2)

```

```
"4. Trả về màu hiện tại l12"
# print(l12.color)
">>>256"

"5. Đổi màu l12"
l12.color = 4

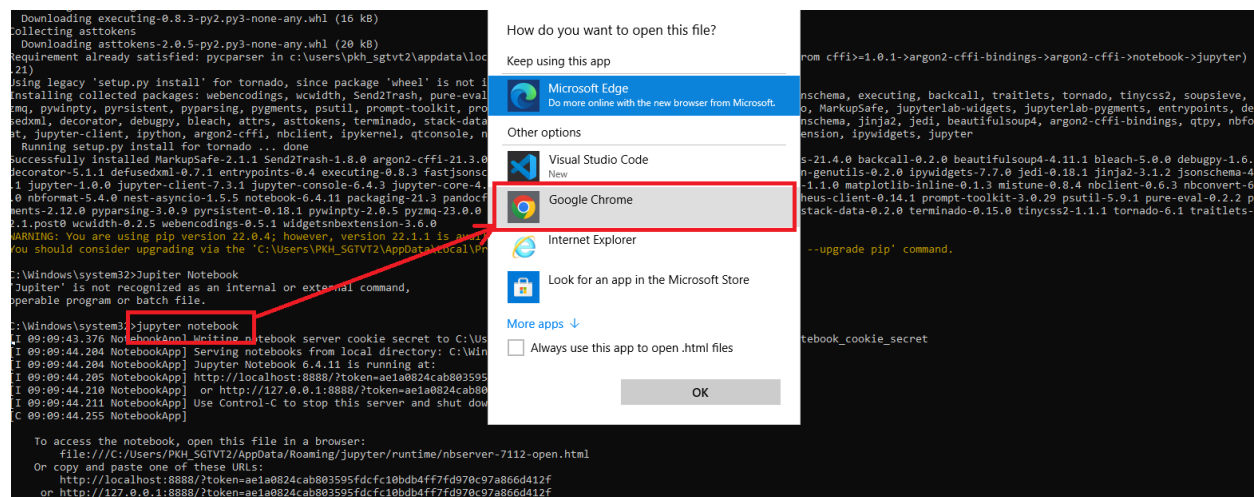
"6. Trả về giá trị layer cho line l12"
print(l12.layer)
">>>Layer1"
```

➤ Giới thiệu và cách cài đặt chương trình Jupyter Notebook:

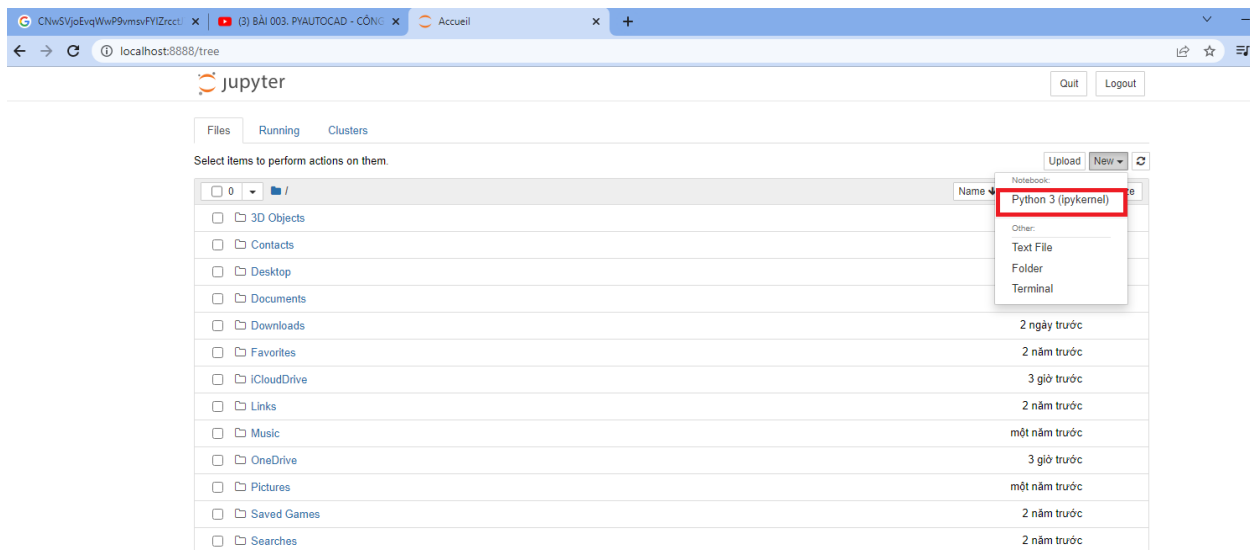
Jupyter Notebook là 1 chương trình soạn thảo code thông qua trình duyệt web giúp bạn có thể thực thi theo từng dòng code 1 cách thuận tiện (Bạn Google để tìm hiểu những các tính năng **jupyter notebook**). Nó rất hữu ích trong phân tích dữ liệu cũng như học lập trình Cad cho các bạn mới bắt đầu.

Gõ trong cài đặt: **pip install jupyter** trong cmd

Sau khi cài đặt xong, để mở trình soạn thảo code Jupyter Notebook bạn tiếp tục gõ dòng lệnh trong cmd như sau: **Jupyter Notebook** và chọn 1 trong các chương trình duyệt web để mở.

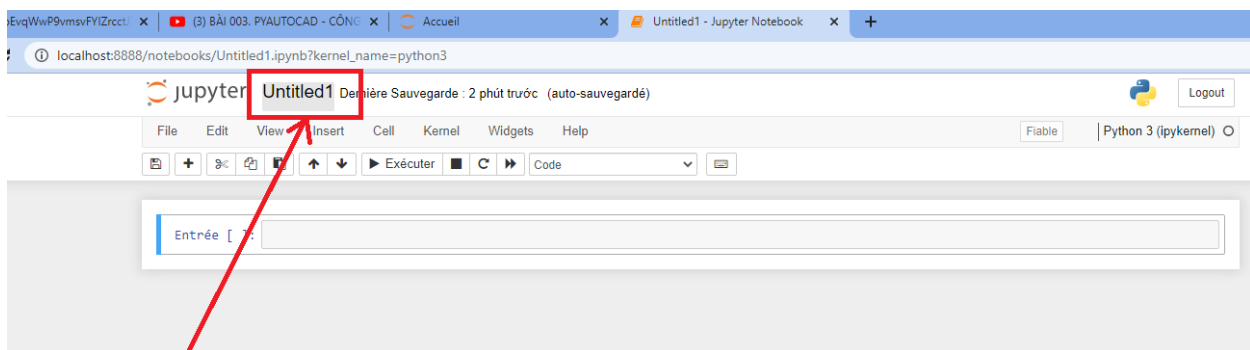


Bạn vào thẻ **New** để tạo 1 file trình soạn thảo mới như hình bên dưới:

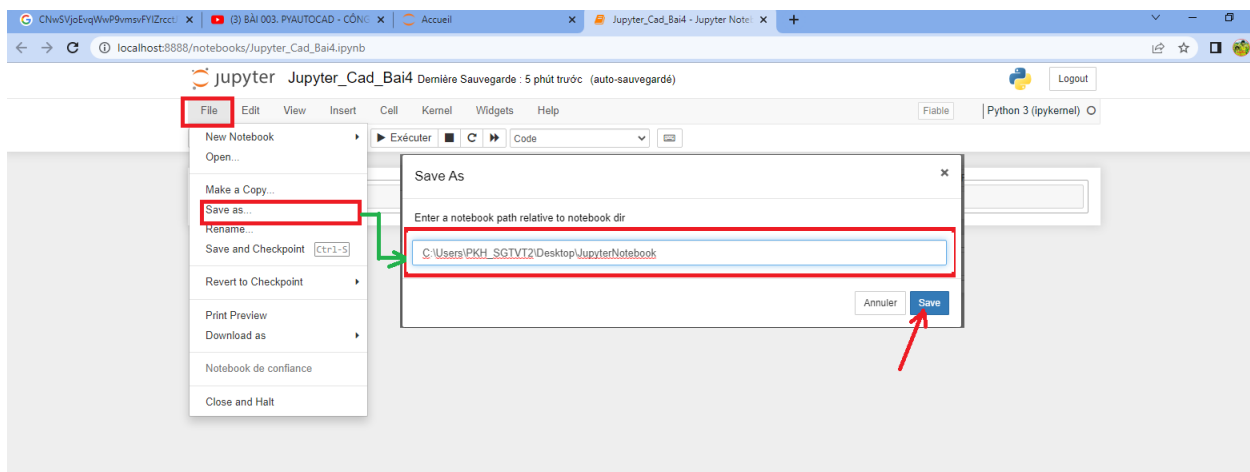


Sau khi màn hình mở trình soạn thảo bạn có thể:

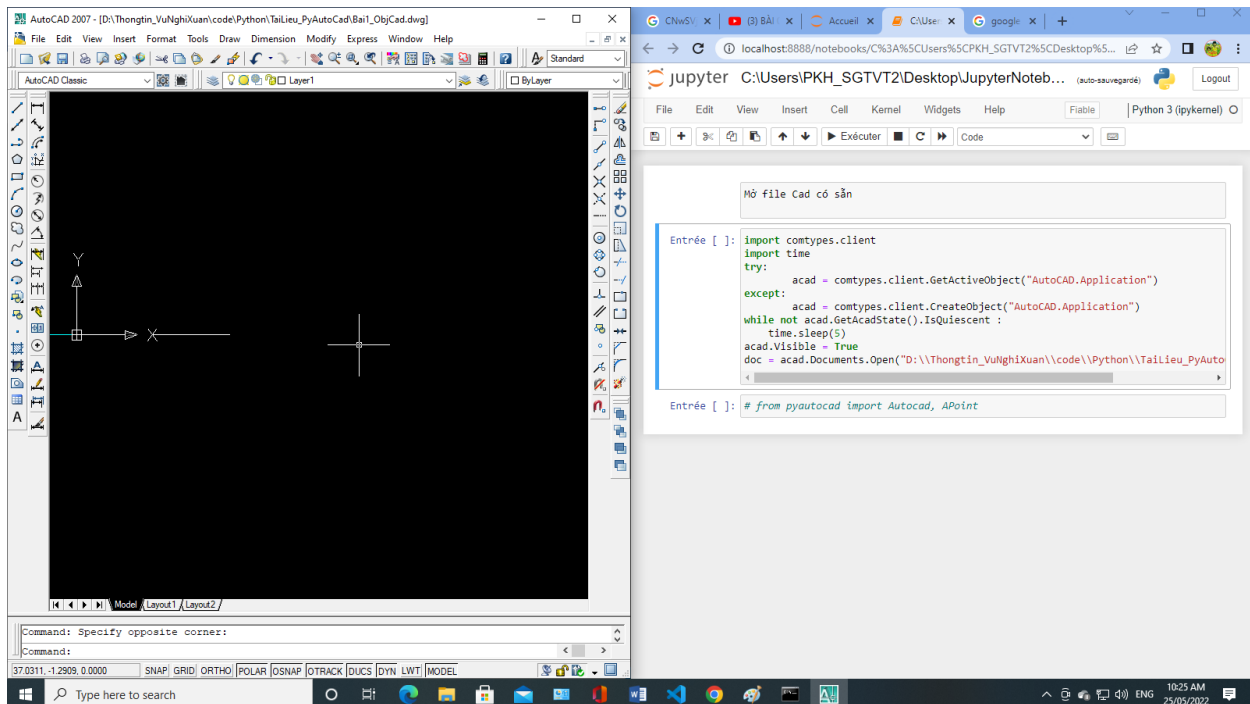
+ Nhấp vào vị trí **Untitled 1** thực hiện đổi tên file:



+ **Save as** về đường dẫn bạn cần lưu:



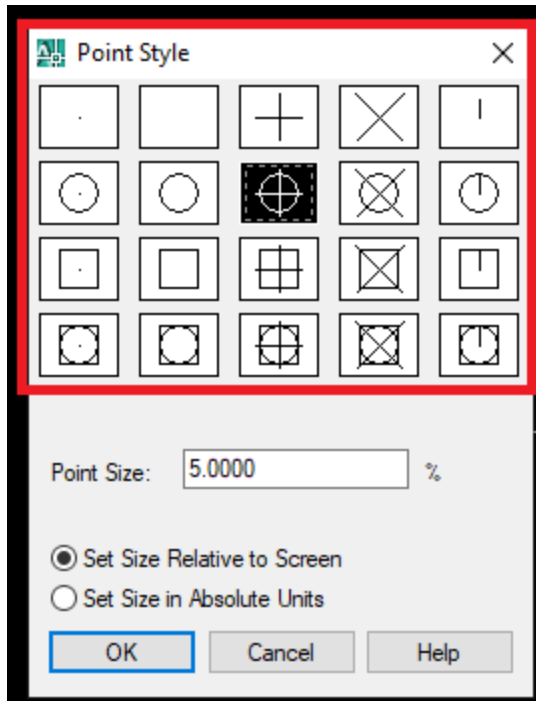
Chạy đoạn code đầu tiên để mở file Cad đã lưu trước đó và xem kết quả bên dưới



Trên đây là phần giới thiệu 1 trình soạn thảo mà nhiều bạn cũng hay sử dụng vì tính tiện lợi của nó. Do mình ko quen sử dụng Jupyter Notebook nên mình quay trở lại sử dụng VSC và tiếp tục bài học, còn các bạn muốn sử dụng trình soạn thảo nào cũng được.

➤ **Đối tượng Point và cách lấy thông (PDMODE và PDSIZE):**

Để thay đổi và định dạng Point trong bản vẽ Cad, bạn vào thẻ **Format** → **Point Style**

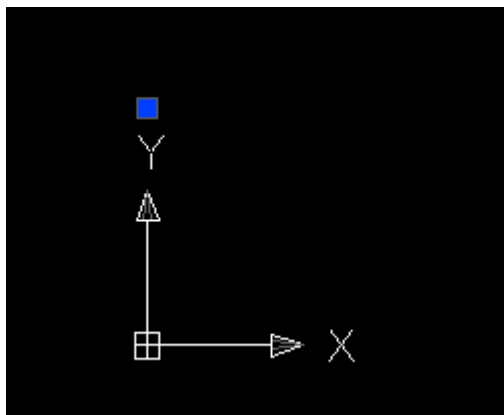


Thông thường biến **PDMODE** đã đặt chế độ về Point Object Display (tức là chế độ hiển thị đối tượng điểm). Trong code python bạn dùng hàm **GetVariable** như sau:

```
from pyautocad import Autocad, APoint
acad = Autocad(create_if_not_exists=True)
P1= APoint(0, 10)
P2= acad.model.AddPoint(P1)

print(f"Thông số PDMODE là: {acad.doc.GetVariable('PDMODE')}")
'>>>Thông số PDMODE là: 0'
```

Sau khi thực hiện đoạn code trong bản vẽ Cad chỉ nhìn thấy điểm P2 chỉ là 1 dấu chấm như hình:



Để thay đổi thông số này chúng ta tham khảo bảng các thông số PDMODE:

.		+	×	'
0	1	2	3	4
○	○	⊕	⊗	⊙
32	33	34	35	36
□	□	⊞	⊠	⊡
64	65	66	67	68
⊞	⊠	⊡	⊢	⊣
96	97	98	99	100

Thay code từ hàm **GetVariable** thành hàm **SetVariable** với thông số là 34:

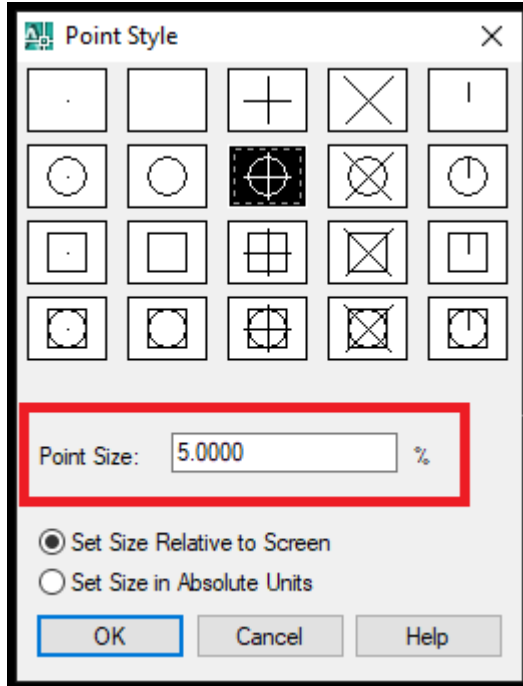
```
from pyautocad import Autocad, APoint
acad = Autocad(create_if_not_exists=True)
P1= APoint(0, 10)
P2= acad.model.AddPoint(P1)

print(f"Thông số PDMODE là: {acad.doc.GetVariable('PDMODE')}")
'>>>Thông số PDMODE là: 0';
```

```
acad.doc.SetVariable('PDMODE',34)
```

Sau khi thực thi code bạn nhìn vào bản vẽ Cad để thấy sự thay đổi này.

Để thay đổi PDSIZE, tức là size cỡ của điểm trong bản vẽ Cad



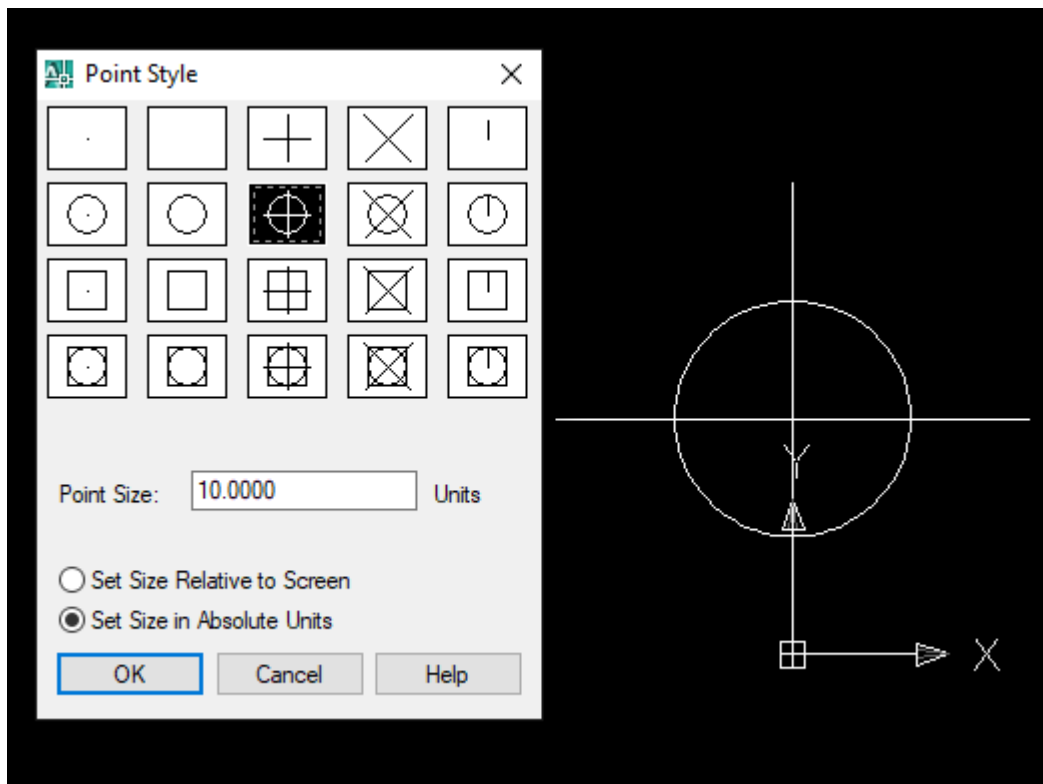
Ta thay đổi code như sau:

```
from pyautocad import Autocad, APoint
acad = Autocad(create_if_not_exists=True)
P1= APoint(0, 10)
P2= acad.model.AddPoint(P1)

# print(f"Thông số PDMODE là: {acad.doc.GetVariable('PDMODE')}")
'>>>Thông số PDMODE là: 0';

# acad.doc.SetVariable('PDMODE',34)
acad.doc.SetVariable('PDSIZE',10)
print (f"Thông số PDSIZE là: {acad.doc.GetVariable('PDSIZE')}")
">>>Thông số PDSIZE là: 10.0"
```

Kết quả sau khi thực hiện code:



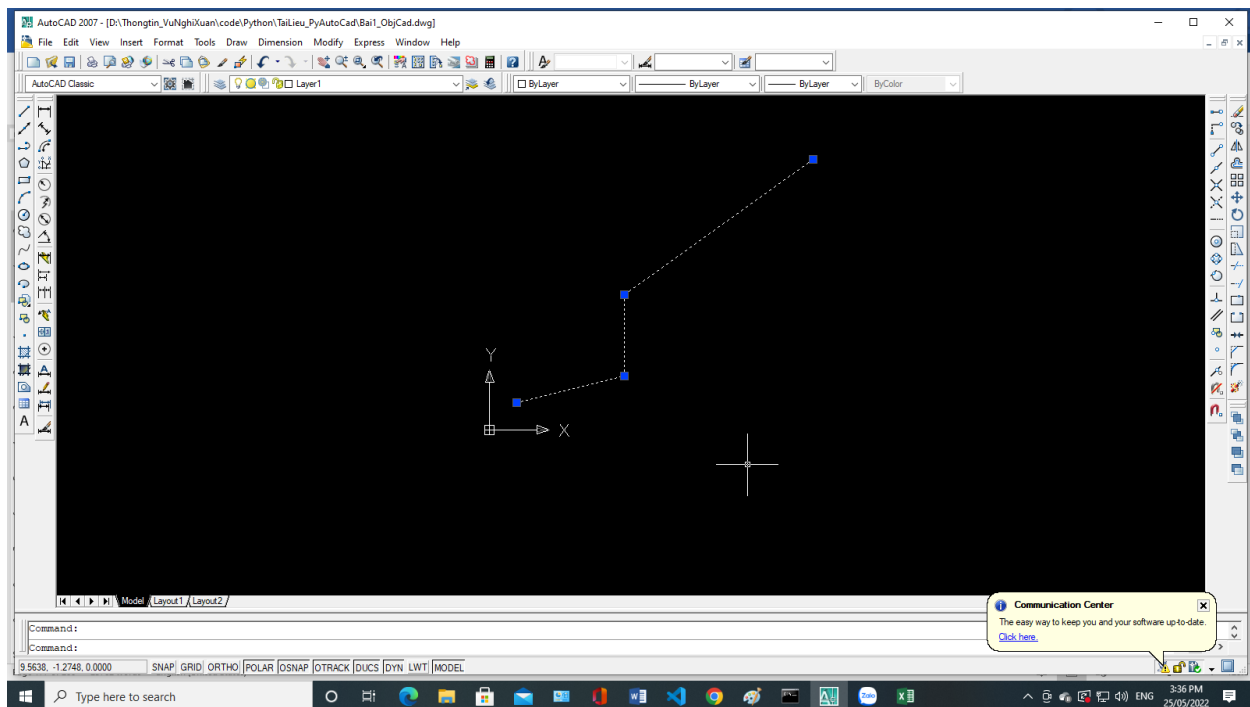
➤ Vẽ đường PolyLine:

Sau đây là code thực hiện đường gấp khúc (polyline) qua 4 điểm

```
from pyautocad import Autocad, aDouble
acad = Autocad(create_if_not_exists=True)

"1. Vẽ đường PolyLine"
P1 = aDouble(1,1,0,5,2,0, 5,5,0, 12,10,0)
acad.model.AddPolyline(P1)
```

Kết quả thực thi code:



➤ Vẽ đường tròn tâm O, text, Mtex:

```
from psutil import AccessDenied
from pyautocad import Autocad, APoint
acad = Autocad(create_if_not_exists=True)

Center_0 = APoint(0,0)

# Add điểm và thay đổi định dạng tâm
0 = acad.model.AddPoint(Center_0)
acad.doc.SetVariable('PDMODE',2)

DuongKinh = 50
sizeFont = 2.5
acad.model.AddCircle(Center_0, DuongKinh)

"Tạo Text"
text = acad.model.AddText("P1",Center_0, sizeFont)

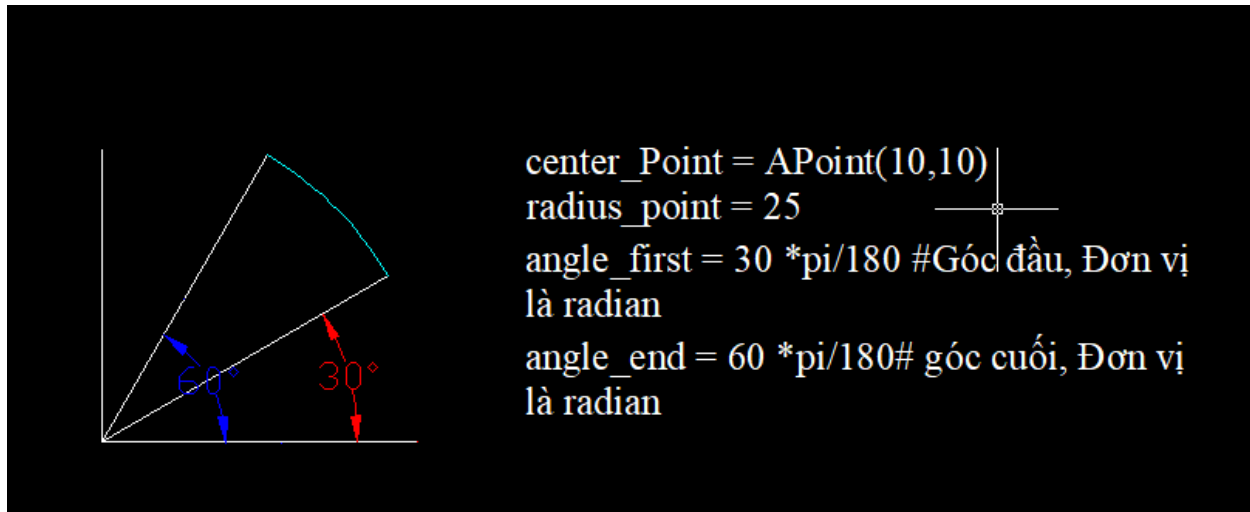
"Tạo MText"
pos_Mtext = APoint(0,60)
acad.model.AddPoint(pos_Mtext)
Crong_Mtext = 20
```

```
Mtext = acad.model.AddMText(pos_Mtext, Crong_Mtext, "Đây là đường tròn Tâm P1")
```

Các bạn tự thực thi code và xem kết quả trên Cad

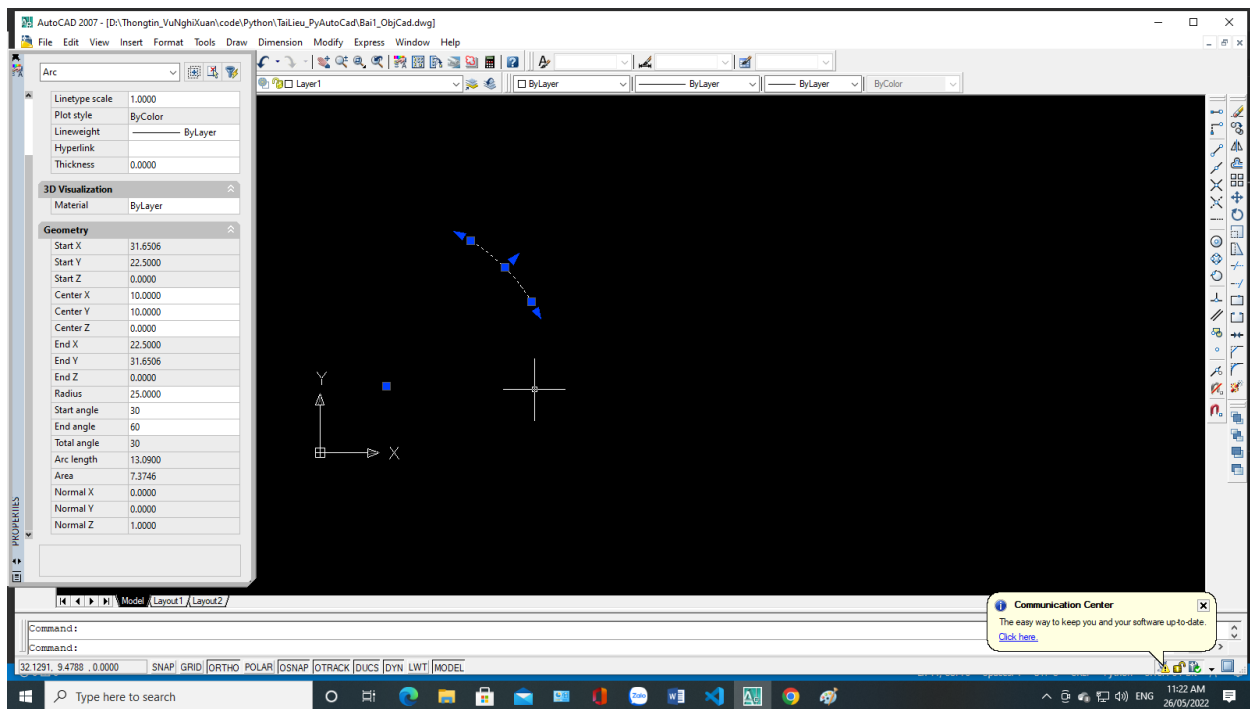
➤ **Vẽ cung tròn (Arc), Ê líp (Ellipse):**

Vẽ cung tròn với các thông số như hình vẽ

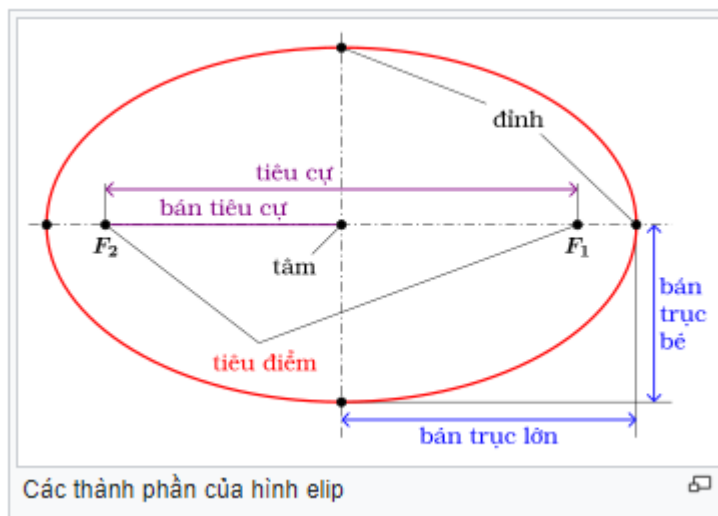


```
from pyautocad import Autocad, APoint  
acad = Autocad(create_if_not_exists=True)  
  
center_Point = APoint(10,10)  
radius_point = 25  
angle_first = 30 #Góc đầu, Đơn vị là radian  
angle_end = 60 # góc cuối, Đơn vị là radian  
  
acad.model.AddArc(center_Point, radius_point, angle_first, angle_end)
```

Kết quả thực hiện code vẽ cung tròn (Arc):



➤ Vẽ hình Elip:



Tham khảo code và thực hành vẽ đường Ellipse sau:

```
from pyautocad import Autocad, APoint
acad = Autocad(create_if_not_exists=True)
from math import pi
```

```

"1. Vẽ cung tròn"
# center_Point = APoint(10,10)
# radius_point = 25
# angle_first = 30 *pi/180 #Góc đầu, Đơn vị là radian
# angle_end = 60 *pi/180# góc cuối, Đơn vị là radian

# acad.model.AddArc(center_Point, radius_point, angle_first, angle_end)

"2. Vẽ Ellispse"
center_elip = APoint(1,1)
pointOfRadiusBig= APoint(5,8) # Điểm bán kính lớn Ellipse

ratio_smallRadius_bigRadius = 0.5 # Tỷ lệ bán kính nhỏ /bán kính lớn
acad.model.AddEllipse(center_elip, pointOfRadiusBig, ratio_smallRadius_bigRadius)
#ELLIPSE

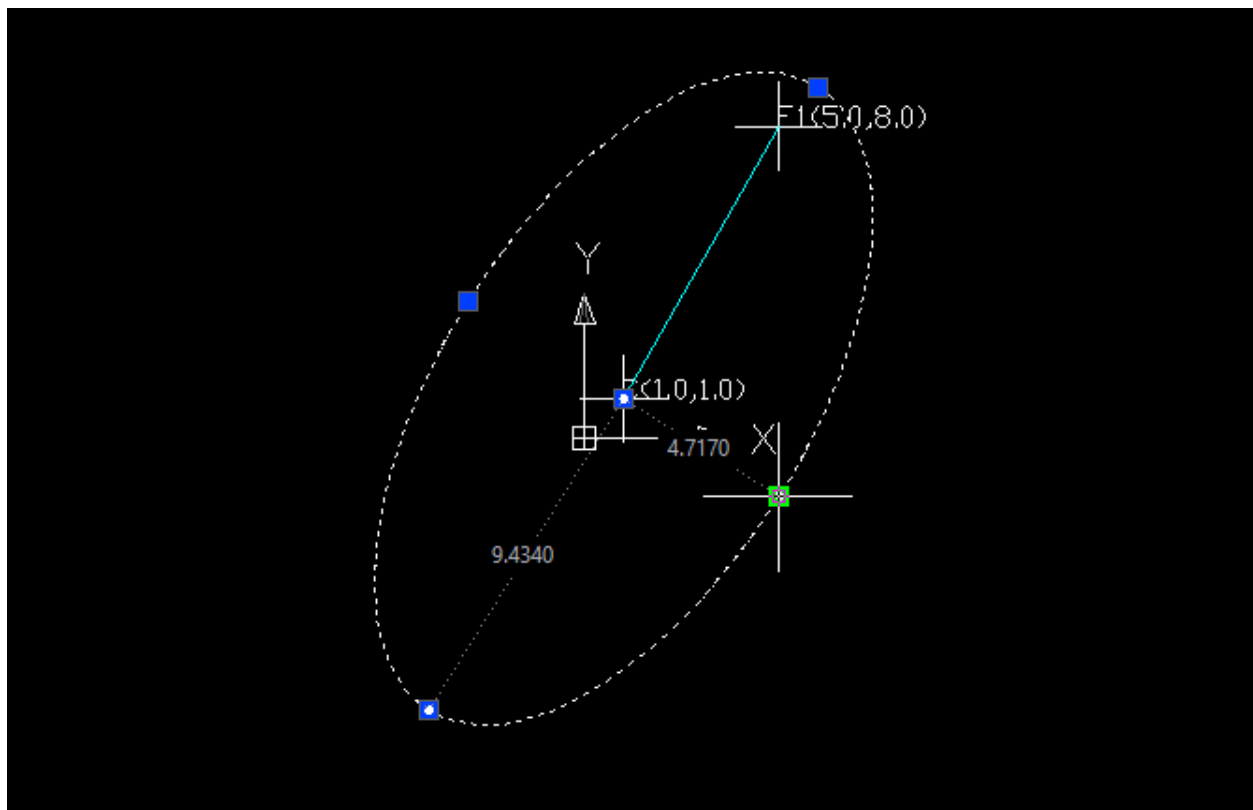
"2a. Vẽ tọa điểm và gán tọa độ"
sizeFont = 0.5
acad.model.AddPoint(center_elip)
acad.model.AddText(f"C({center_elip.x},{center_elip.y})",center_elip, sizeFont)

acad.model.AddPoint(pointOfRadiusBig)
acad.model.AddText(f"F1({pointOfRadiusBig.x},{pointOfRadiusBig.y})",pointOfRadiusBig, sizeFont)

"2b. Vẽ đường bán kính lớn"
bk = acad.model.AddLine(center_elip, pointOfRadiusBig)
bk.color = 4

```

Kết quả thực thi code:



➤ **Kích thước Dimesion và cách tính khoảng cách ghi kích thước cho đường Line bất kỳ:**

```
from pyautocad import Autocad, APoint
acad = Autocad(create_if_not_exists=True)
import math

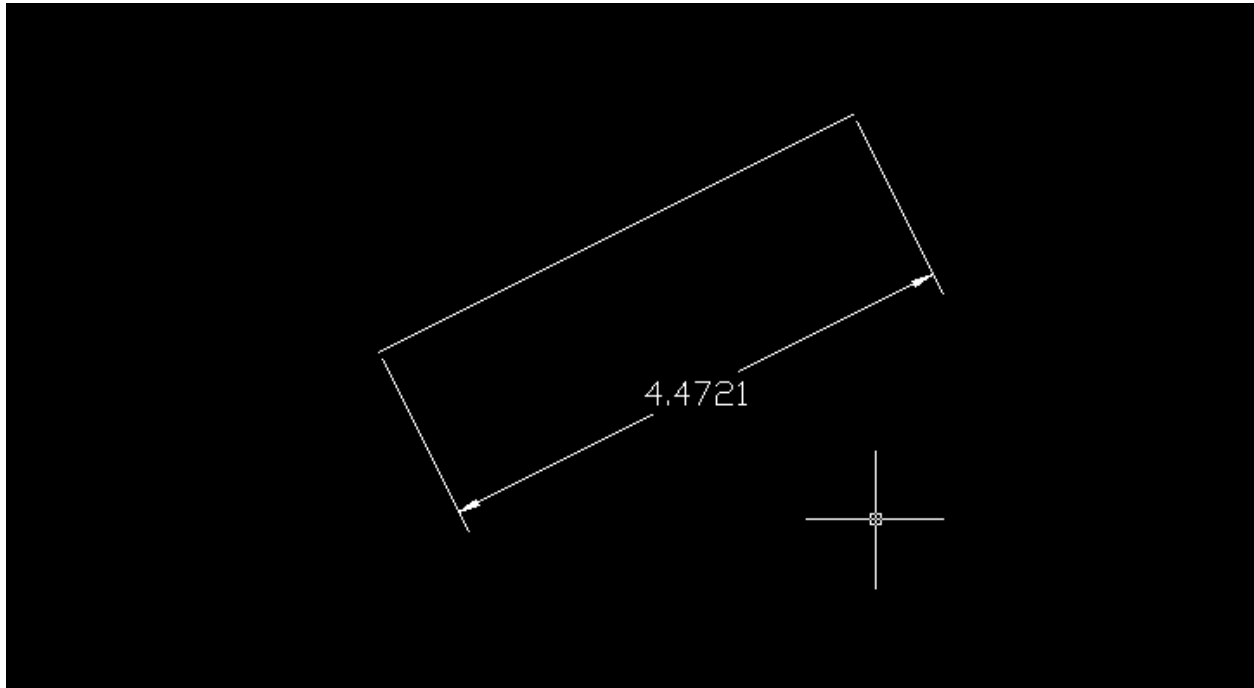
P1 = APoint(4,5)
P2 = APoint(8,7)

"Cách tính điểm đặt Text cho bất kỳ đường Line"
kc = 1 # Khoảng cách cách đường Line
goc = math.atan(P2.y/P2.x)
k = kc/math.cos(goc)

poss_Text = APoint(0,k) # Vị trí đặt text

Line = acad.model.AddLine(P1,P2)
acad.model.AddDimAligned(P1,P2, poss_Text)
```


Kết quả thực thi code:



➤ Ghi kích thước cho góc (DimAngular):

```
from pyautocad import Autocad, APoint, aDouble
acad = Autocad(create_if_not_exists=True)
import math

"1. Ghi kích thước đường thẳng (Dimersion Line)"
# P1 = APoint(4,5)
# P2 = APoint(8,7)

# "Cách tính điểm đặt Text cho bất kỳ đường Line"
# kc = 1 # Khoảng cách cách đường Line
# goc = math.atan(P2.y/P2.x)
# k = kc/math.cos(goc)

# poss_Text = APoint(0,k) # Vị trí đặt text

# Line = acad.model.AddLine(P1,P2)
# acad.model.AddDimAligned(P1,P2, poss_Text)
```

```

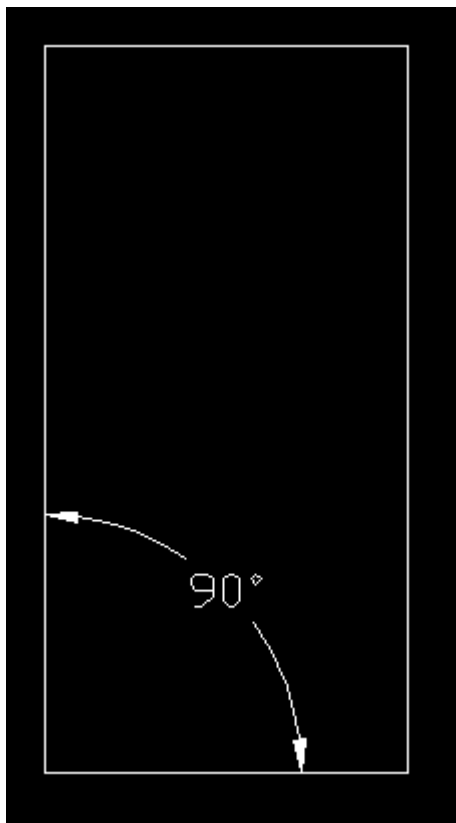
"2. Ghi kích thước góc (DimAngular)"
P1 = (10,10,0)
P2 = (12,10,0)
P3 = (12,14,0)
P4 = (10,14,0)

poline = aDouble(*P1, *P2, *P3, *P4, *P1)
# print(poline)
acad.model.AddPolyLine(poline)

"2.1 Vẽ DimAngular"
cen_poss = APoint(P1)
begin_poss = APoint(P2)
end_poss = APoint(P4)
text_poss = APoint(11,11,0) # vị trí text góc vuông 45 độ
acad.model.AddDimAngular(cen_poss, begin_poss, end_poss, text_poss)

```

Kết quả code:



➤ Ghi kích thước cung tròn:

```
from pyautocad import Autocad, APoint, aDouble
acad = Autocad(create_if_not_exists=True)

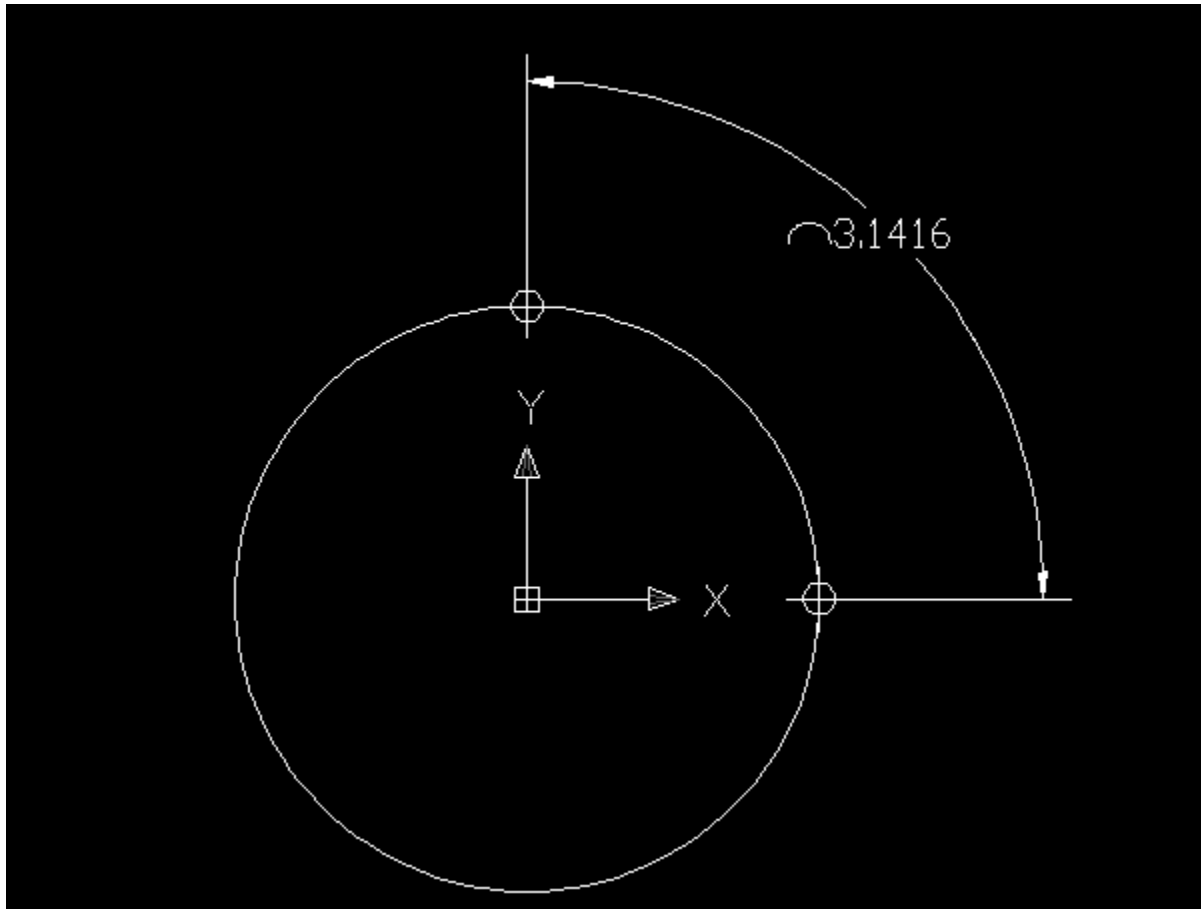
P = APoint(0,0)
A = APoint(2,0)
B = APoint(0,2)

"Add 2 điểm A, B vào bản vẽ"
acad.model.AddPoint(A)
acad.model.AddPoint(B)
acad.doc.SetVariable('PDMODE',34)

"Vẽ đường tròn"
r = 2
acad.model.AddCircle(P,r)

" Ghi kích thước chiều dài cung tròn"
text_poss = APoint(2.5,2.5) #2.5 >r
acad.model.AddDimArc(P, A, B, text_poss)
```

Kết quả chạy code:



➤ Cách ghi kích thước bán kính, đường kính đường tròn:

```
from pyautocad import Autocad, APoint, aDouble
acad = Autocad(create_if_not_exists=True)

P = APoint(0,0)
A = APoint(-2,0) #Vị trí đầu
B = APoint(2,0) #Vị trí kết thúc
C = APoint(0,2) # Vị trí đầu để ghi bán kính

"Add 2 điểm A, B vào bản vẽ"
acad.model.AddPoint(A)
acad.model.AddPoint(B)
acad.model.AddPoint(C)
acad.doc.SetVariable('PDMODE',34)
```

```

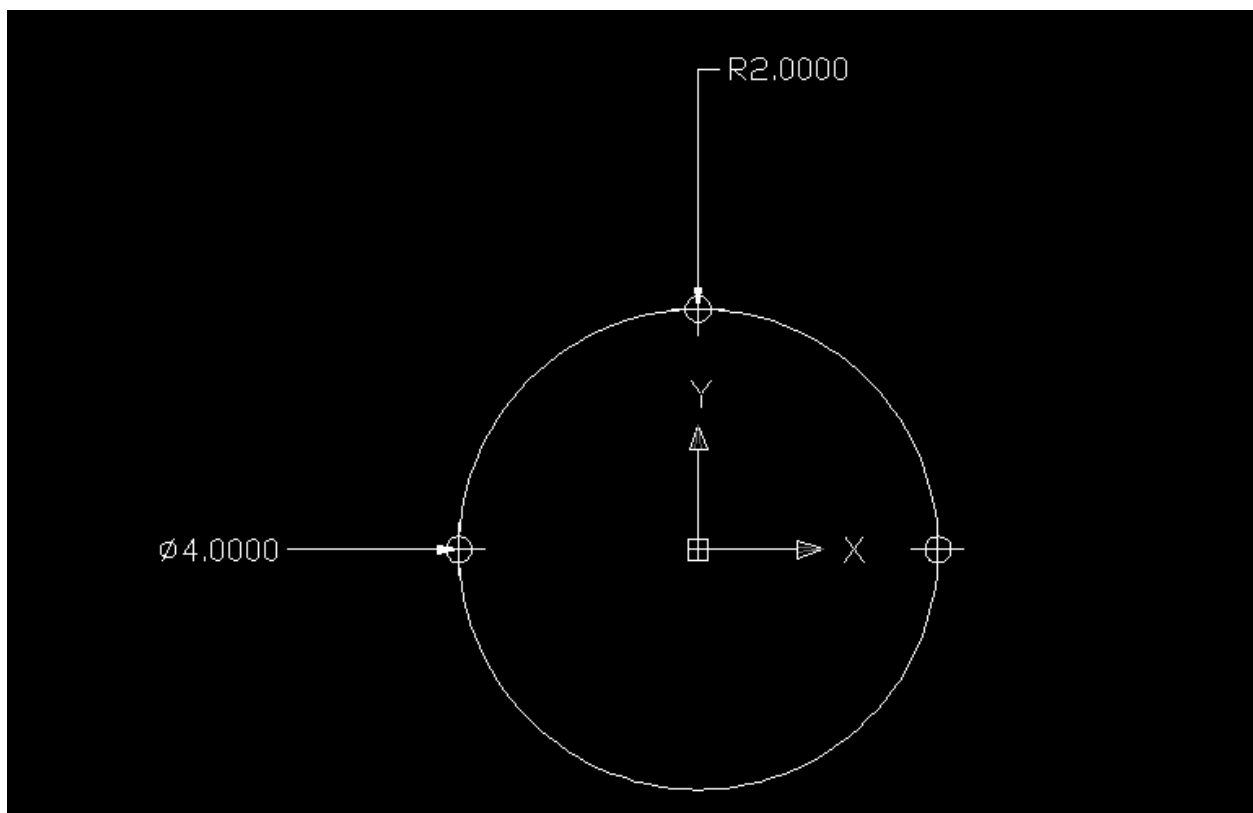
"Vẽ đường tròn"
r = 2
acad.model.AddCircle(P,r)

"Ghi kích thước Đường kính"
text_poss = 2 # kiểu int, có giá trị =r
acad.model.AddDimDiametric(A, B, text_poss)
# acad.model.AddDimDiametric(B, A, text_poss) # "Cách ghi ngược với chiều
kích thước"

"Ghi kích thước Bán kính"
text_poss_radius = 2 # kiểu int, có giá trị =r
acad.model.AddDimRadial(P, C, text_poss_radius)

```

Kết quả thực hiện code:



➤ Các phương thức cho đối tượng hình học (Copy, move, delete, rotate, mirror):

Chú ý: Câu lệnh **Copy** phải thực hiện song song với lệnh **Move** theo đoạn code bên dưới

```
from pyautocad import Autocad, APoint, aDouble
acad = Autocad(create_if_not_exists=True)
from math import pi

P = APoint(0,0)
A = APoint(-10,0) #Vị trí đầu
B = APoint(10,0) #Vị trí kết thúc
C = APoint(0,10) # Vị trí đầu để ghi bán kính

"Add 2 điểm A, B vào bản vẽ"
acad.model.AddPoint(A)
acad.model.AddPoint(B)
acad.model.AddPoint(C)
acad.doc.SetVariable('PDMODE',34)

"Vẽ đường tròn"
r = 10
circle_Obj = acad.model.AddCircle(P,r)

"1. Thực hiện lệnh Copy đối tượng"
circle_Obj_2 = circle_Obj.Copy()

"2. Thực hiện Move để dời ra khỏi vị trí chồng lấn đối tượng đã Copy trước đó"
circle_Obj_2.Move(P,B)

"3. Xóa đối tượng"
circle_Obj.Delete()

"4. Xoay đối tượng (+45*pi/180: ngược chiều Đồng hồ, dấu (-) cùng chiều"
DK_AB = acad.model.AddLine(A,B)
DK_AB.Rotate(A, 45*pi/180)
DK_AB.color = 4

"5. Lệnh mirror đối tượng hình học"
mir_AB = DK_AB.Mirror(A, B) #MIRROR
mir_AB.color = 6
```

```

"6. Lệnh mirror đối tượng String (chuỗi)"
x1 = 0 # Mirror ko lật ngược
x2 = 1 # lật ngược chữ
sizeFont = 1
VuNghixuan = acad.model.AddText("VuNghixuan",P, sizeFont)

"Mirror Text"
acad.doc.SetVariable ("MIRRTEXT", x2) #Thay đổi x2 để thấy sự thay đổi
VuNghixuan.Mirror(P,C)
VuNghixuan.Mirror(A,C)

```

➤ Tạo hàng loạt file Cad:

```

from pyautocad import Autocad, APoint
import os

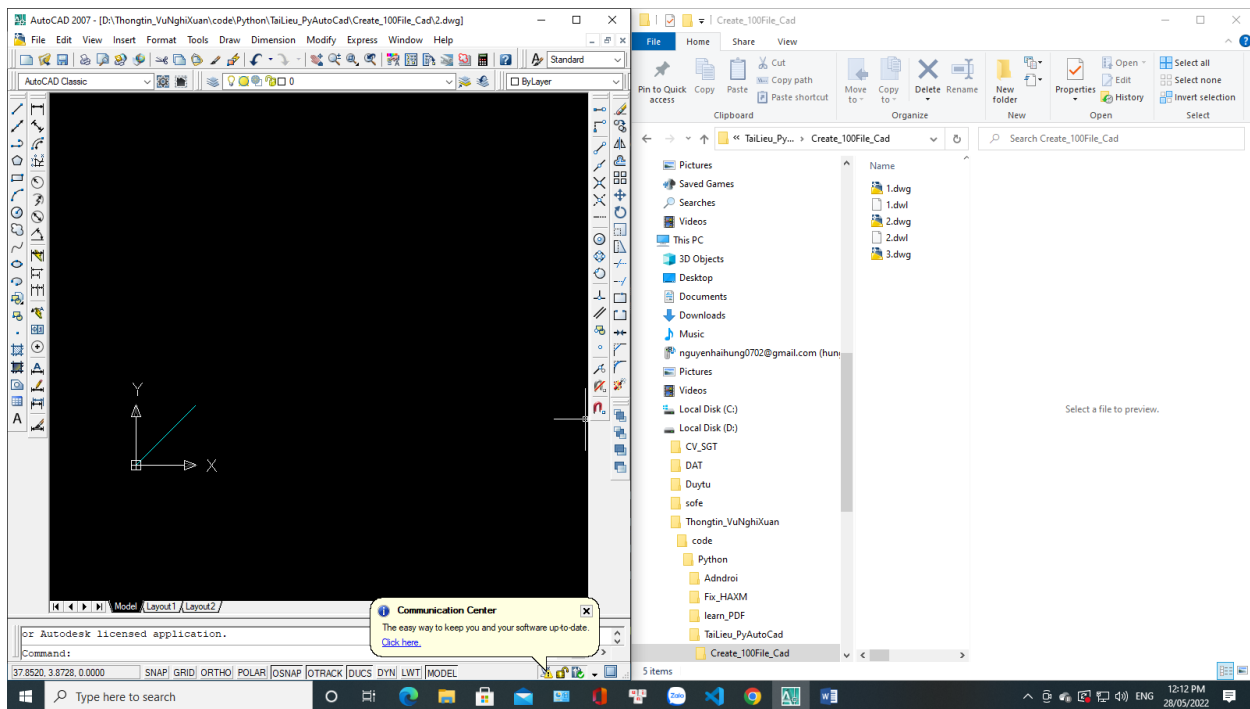
a = Autocad(create_if_not_exists = True, visible=True)
path_file =
r'D:\Thongtin_VuNghixuan\code\Python\TaiLieu_PyAutoCad\Create_100File_Cad'

"Tạo 2 điểm vẽ đường thẳng"
P1 = APoint(0,0)
P2 = APoint(5,5)
Line = a.model.AddLine(P1,P2)
Line.color = 4
for i in range(3):
    a.doc.saveas(path_file + f"\{str(i+1)}.dwg")

"""Taskkill.exe là một tệp thực thi Công cụ diệt tác vụ và không gây ra bất kỳ
mối đe dọa nào cho PC của bạn.
Windows là một loạt các hệ điều hành được Microsoft phát triển, tiếp thị và
bán"""
os.system ('TASKKILL / T / IM acad.exe')

```

Kết quả chạy code:



➤ Tìm tọa độ các đỉnh

```
from pyautocad import Autocad, aDouble, APoint
acad = Autocad(create_if_not_exists=True)

p1= APoint(10)
p2= APoint(20)

"1. Vẽ đường PolyLine"
P1 = aDouble(1,1,0,5,2,0, 5,5,0, 12,10,0)
acad.model.AddPolyline(P1)

"2. Lấy tọa độ các đỉnh do người dùng tự chọn"
pos_vertex = acad.get_selection()
for objs in pos_vertex:
    obj = objs.Coordinates

    print(obj)
```

Lưu ý: Lúc chạy code cần chọn đối tượng (chỉ 1 đối tượng)

Kết quả thực hiện code:


```
Select objects
(1.0, 1.0, 0.0, 5.0, 2.0, 0.0, 5.0, 5.0, 0.0, 12.0, 10.0, 0.0)
PS D:\Thongtin_VuNghiXuan\code\Python\TaiLieu_PyAutoCad> []
```

➤ Lấy danh sách Layer và tạo Layer mới:

```
from pyautocad import Autocad, aDouble, APoint
acad = Autocad(create_if_not_exists=True)
```

"1. Tổng số Layer trên bản vẽ"

```
layer_count = acad.doc.Layers.count
print(layer_count)
```

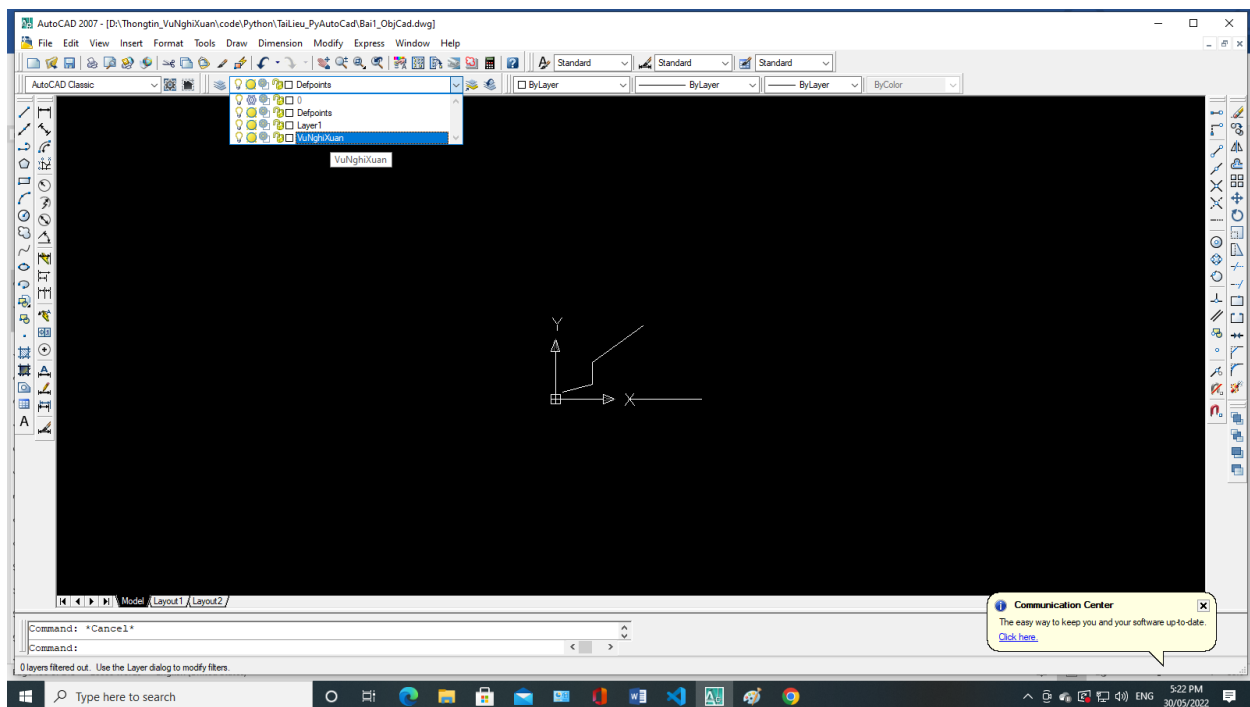
"2. Lấy tên danh sách Layer trên bản vẽ"

```
for num in range(layer_count):
    print(acad.doc.Layers.Item(num).Name)
```

"3. Cài đặt Layer mới"

```
layer_new = acad.doc.Layers.add("VuNghiXuan")
```

Kết quả chạy code:



➤ Truy xuất thông tin và thuộc tính các đối tượng Cad:

```
from pyautocad import Autocad, APoint
```

```

acad=Autocad(False,True)
objs=acad.model

"1. Số đối tượng tồn tại trong cad";
count=objs.count
print('Tổng số đối tượng: {count}')

'2. Liệt kê tên đối tượng'
# Cách 1:
for i in range(objs.count):
    # objs[i].color = 11
    # objs[i].layer = 'Layer1'
    print (i+1, objs[i].ObjectName)

# Cách 2:
for obj in acad.iter_objects():
    print(f"cách thứ 2: {obj.ObjectName}")

"3. Liệt kê các đối tượng Text và Line"
for obj in acad.iter_objects(['Text', 'Line']):
    print (obj.ObjectName)

"4. Lấy tọa độ các đối tượng PolyLine"
for item in acad.iter_objects("PolyLine"):
    print (item.Coordinates) #Coordinates

"5. Lấy tên đối tượng Text và tọa độ"
for text in acad.iter_objects('Text'):
    print (text.TextString, text.InsertionPoint)

"6. Lấy tên đối tượng Line"
for line in acad.iter_objects('Line'):
    p1 = APoint(line.StartPoint)
    p2 = APoint(line.EndPoint)
    print(p1, p2)
    "Chú ý sẽ bị lỗi khi gặp PolyLine"

"7. Lấy thuộc tính các đối tượng Line"
for line in acad.iter_objects('Line'):
    print(f"Layer: {line.Layer}")
    print(f"Linetype: {line.Linetype}")
    print(f"LinetypeScale: {line.LinetypeScale}")
    print(f"Lineweight: {line.Lineweight}")

```

```
print(f"Thickness: {line.Thickness}")
print(f"Material: {line.Material}")
```

➤ Liệt kê các đối tượng và thuộc tính (delete đối tượng):

```
from pyautocad import Autocad, APoint

acad=Autocad(create_if_not_exists=False)
objs=acad.model

"1. Số đối tượng tồn tại trong cad";
i_num = objs.count
print(f'Tổng số đối tượng: {i_num}')

"2. Liệt kê tên đối tượng"
# Cách 1:
for i in range(objs.count):
    # objs[i].color = 11
    # objs[i].layer = 'Layer1'
    print (i+1, objs[i].ObjectName)

# Cách 2:
for obj in acad.iter_objects():
    print(f"cách thứ 2: {obj.ObjectName}")

"3. Liệt kê các đối tượng Text và Line"
for obj in acad.iter_objects(['Text', 'Line']):
    print (obj.ObjectName)

"4. Lấy tọa độ các đối tượng PolyLine"
for item in acad.iter_objects("PolyLine"):
    print (item.Coordinates) #Coordinates

"5. Lấy tên đối tượng Text và tọa độ"
for text in acad.iter_objects('Text'):
    print (text.TextString, text.InsertionPoint)

"6. Lấy tên đối tượng Line"
for line in acad.iter_objects('Line'):
    p1 = APoint(line.StartPoint)
    p2 = APoint(line.EndPoint)
```

```

print(p1, p2)
"Chú ý sẽ bị lỗi khi gặp PolyLine"

"7. Lấy thuộc tính các đối tượng Line"
for line in acad.iter_objects('Line'):
    print(f"Layer: {line.Layer}")
    print(f"Linetype: {line.Linetype}")
    print(f"LinetypeScale: {line.LinetypeScale}")
    print(f"Lineweight: {line.Lineweight}")
    print(f"Thickness: {line.Thickness}")
    print(f"Material: {line.Material}")

"7. Xóa đối tượng"
"Cách 1:";
# for obj in
acad.iter_objects(object_name_or_list="AcDbCircle"):    #object_name_or_list="AcDb
Circle",
#     obj.Delete()
# Cách 1 trên bị lỗi khi có quá nhiều đối tượng

"Cách 2: Không lỗi như cách 1";
import win32com.client

acad = win32com.client.Dispatch("AutoCAD.Application")

acad.Visible = True
acadModel = acad.ActiveDocument.ModelSpace
for object in acadModel:
    if object.ObjectName == "AcDbCircle":
        object.Delete()

```