

## Chương 8

### LÀM VIỆC VỚI ĐỐI TƯỢNG SELECTIONSET

***Nội dung chương :***

- *Khai báo và khởi tạo một đối tượng SelectionSet*
- *Giới thiệu một số phương thức chọn đối tượng thường dùng*
- *Kết hợp bộ lọc đối tượng vào các phương thức chọn đối tượng*

## 8.1. Quy trình làm việc với đối tượng SelectionSet

Đối tượng **SelectionSet** được sử dụng để chứa các đối tượng được chọn trong bản vẽ. Mỗi đối tượng SelectionSet có nhiều phương thức để thêm đối tượng vào nó như AddItems, SelectOnScreen...

Mỗi bản vẽ Autocad đều có một tập đối tượng **SelectionSets** chứa các đối tượng SelectionSet trong bản vẽ. Người dùng có thể tạo ra nhiều đối tượng SelectionSet bằng cách dùng phương thức Add có trong tập đối tượng SelectionSets.

Đối tượng SelectionSet chỉ có tính chất tạm thời nghĩa là khi đóng bản vẽ lại thì mọi đối tượng SelectionSet sẽ là một tập rỗng.

Quá trình làm việc với đối tượng SelectionSet gồm các bước sau:

- Khai báo đối tượng SelectionSet
- Khởi tạo đối tượng SelectionSet.
- Thêm các đối tượng vào SelectionSet
- Thực hiện các thao tác cần thiết trên các đối tượng trong SelectionSet

## 8.2. Khai báo và khởi tạo đối tượng SelectionSet

### ▪ Cú pháp

```
Set SelectionSetObj = ThisDrawing.SelectionSets.Add (SelectionSetName)
```

### ▪ Giải thích

Tham số	Giải thích
SelectionSetObj	Kiểu acadSelectionSet – Biến chứa đối tượng SelectionSet được tạo
SelectionSetName	Kiểu String – Chuỗi xác định tên SelectionSet được tạo

- **Ví dụ**

```

Sub Ex_AddSelectionSet ()
    'This Example will create a SelectionSet
    On Error Resume Next
    Dim SelectionSetObj As AcadSelectionSet
    Set SelectionSetObj = ThisDrawing.SelectionSets("Kira")
    If Err <> 0 Then
        Err.Clear
        Set SelectionSetObj = ThisDrawing.SelectionSets.Add("Kira")
    Else
        SelectionSetObj.Clear
    End If
    MsgBox "Name of new SelectionSet was:" & SelectionSetObj.Name
End Sub

```

- **Giải thích dòng lệnh**

- Dòng mã lệnh số 1: Tắt các thông báo lỗi khi xuất hiện
- Dòng mã lệnh số 2: Khai báo đối tượng SelectionSet
- Dòng mã lệnh số 3: Nếu đối tượng SelectionSet có tên “Kira” tồn tại thì sẽ được gán cho đối tượng SelectionSetObj.
- Dòng mã lệnh số 4: Nếu đối tượng SelectionSet có tên “Kira” chưa tồn tại thì sẽ xuất hiện lỗi.
- Dòng mã lệnh số 5: Xóa lỗi xảy ra.
- Dòng mã lệnh số 6: Khởi tạo đối tượng SelectionSet có tên “Kira”
- Dòng mã lệnh số 7: Nếu đã tồn tại đối tượng SelectionSet có tên “Kira” thì sẽ xóa bỏ nội dung bên trong đối tượng này.

### 8.3. Thêm đối tượng vào trong một SelectionSet

Để thêm các đối tượng hình học vào trong một SelectionSet chúng ta sử dụng phương thức có sẵn trong đối tượng SelectionSet như **AddItems**, **Select**, **SelectAtPoint**, **SelectionOnScreen**, **SelectByPolygon...**

Tùy thuộc vào mỗi phương thức được sử dụng mà có các tham số tương ứng đi kèm khác nhau. Sau đây là một số phương thức thường được sử dụng.

### 8.3.1. Phương thức AddItems

Phương thức này dùng để thêm **TỪNG ĐỐI TƯỢNG** hình học vào trong đối tượng SelectionSet.

#### ▪ Cú pháp

SelectionSetObj.**AddItems** ArrayObj

#### ▪ Giải thích

Tham số	Giải thích
SelectionSetObj	Kiểu acadSelectionSet – Biến chứa đối tượng SelectionSet được tạo
ArrayObj	Kiểu Variant, là mảng chứa các đối tượng được thêm vào SelectionSet

#### ▪ Ví dụ

```
Sub Example_AddItems ()
    On Error Resume Next
    Dim SelectionSetObj As AcadSelectionSet
    Set SelectionSetObj = ThisDrawing.SelectionSets("Kira")
    If Err <> 0 Then
        Err.Clear
        Set SelectionSetObj = ThisDrawing.SelectionSets.Add("Kira")
    Else
        SelectionSetObj.Clear
    End If
    '-----
    Dim ArrayObj(0 To 1) As Variant
    Dim LineObj As AcadLine
    Dim StartPnt(0 To 2) As Double
    Dim EndPnt(0 To 2) As Double
    StartPnt(0) = 0: StartPnt(1) = 0: StartPnt(2) = 0
    EndPnt(0) = 2: EndPnt(1) = 2: EndPnt(2) = 0
    Set LineObj = ThisDrawing.ModelSpace.AddLine(StartPnt, EndPnt)
    Set ArrayObj(0) = LineObj
    '-----
    Dim CircObj As AcadCircle
    Dim CenterPnt(0 To 2) As Double
    Dim Radius As Double
    CenterPnt(0) = 0: CenterPnt(1) = 0: CenterPnt(2) = 0
    Radius = 5
    Set CircObj = ThisDrawing.ModelSpace.AddCircle(CenterPnt, Radius)
    Set ArrayObj(2) = CircObj
    SelectionSetObj.AddItems ArrayObj
    Application.ZoomAll
End Sub
```

### 8.3.2. Phương thức Select

Phương thức **Select** là một phương thức cơ bản của đối tượng **SelectionSet**. Với phương thức này thì người dùng sẽ có nhiều lựa chọn khác nhau để thêm đối tượng vào trong đối tượng **SelectionSet** tùy thuộc vào việc sử dụng các tham số của phương thức.

- **Cú pháp**

```
SelectionSetObj.Select Mode [, Point_1] [, Point_2] [, FilterType] [, FilterData]
```

- **Giải thích**

Tham số	Giải thích
SelectionSetObj	Kiểu acadSelectionSet – Biến chứa đối tượng SelectionSet được tạo
Mode	Kiểu InteGer - Tham số xác định chế độ chọn đối tượng.
Point_1	Tham số tùy chọn, kiểu Variant /Double - Mảng 3 phần tử chứa tọa độ điểm thứ nhất của cửa sổ lựa chọn, sử dụng kết hợp với tham số Point_2
Point_2	Tham số tùy chọn, kiểu Variant/ Double - Mảng 3 phần tử chứa tọa độ điểm thứ hai của cửa sổ lựa chọn, sử dụng kết hợp với tham số Point_1
FilterType, FilterData	Tham số tùy chọn xác định bộ lọc đối tượng (Xem chi tiết tại “Bộ lọc đối tượng”)

- **Giới thiệu tham số Mode**

Tham số **Mode** sẽ xác định cách thức lựa chọn đối tượng khi sử dụng phương thức **Select**.

*Bảng giá trị và ý nghĩa của tham số Mode*

Mode	Giá trị	Ý nghĩa
acSelectionSetWindow	0	Chọn tất cả các đối tượng nằm trong hình chữ nhật giới hạn bởi hai điểm Point_1, Point_2. Hai tham số Point_1, Point_2 bắt buộc phải được sử dụng
acSelectionSetCrossing	1	Chọn tất cả các đối tượng nằm trong hoặc bị cắt bởi hình chữ nhật giới hạn bởi hai điểm Point_1, Point_2. Hai tham số Point_1, Point_2 bắt buộc phải được sử dụng
acSelectionSetPrevious	3	Chọn các đối tượng đã được chọn gần nhất. Hai tham số Point_1, Point_2 được bỏ qua.
acSelectionSetLast	4	Chọn đối tượng cuối cùng được tạo ra. Hai tham số Point_1, Point_2 được bỏ qua
acSelectionSetAll	5	Chọn tất cả các đối tượng trong bản vẽ. Hai tham số Point_1, Point_2 được bỏ qua

▪ **Ví dụ**

```

Sub Example_SelectionSetWindow()
    On Error Resume Next
    Dim SelectionSetObj As AcadSelectionSet
    Set SelectionSetObj = ThisDrawing.SelectionSets("Kira")
    If Err <> 0 Then
        Err.Clear
        Set SelectionSetObj = ThisDrawing.SelectionSets.Add("Kira")
    Else
        SelectionSetObj.Clear
    End If

    -----
    Dim Mode As Integer
    Mode = acSelectionSetWindow
    Dim Point_1(0 To 2) As Double
    Dim Point_2(0 To 2) As Double
    Point_1(0) = 0: Point_1(1) = 0: Point_1(2) = 0
    Point_2(0) = 420: Point_2(1) = 297: Point_2(2) = 0
    SelectionSetObj.Select Mode, Point_1, Point_2
End Sub

```

### 8.3.3. Phương Thức SelectOnScreen

Phương thức này sẽ hiện thị dòng nhắc “Select Object:” tại dòng lệnh và cho phép người dùng chọn đối tượng trực tiếp trên màn hình bản vẽ bằng nhiều cách. Cách thao tác này giống như một số lệnh thông thường của AutoCad như Copy, Move, Delete...

Với phương thức này thì hai tham số Point\_1 và Point\_2 không được sử dụng, hai tham số xác định chế độ lọc FilterType và FilterData có thể được sử dụng.

#### ▪ Cú pháp

```
SelectionSetObj.SelectOnScreen [FilterType] [, FilterData]
```

#### ▪ Giải thích

Tham số	Giải thích
SelectionSetObj	Kiểu acadSelectionSet – Biến chứa đối tượng SelectionSet được tạo ra
FilterType, FilterData	Tham số tùy chọn xác định bộ lọc đối tượng (Xem chi tiết tại “Bộ lọc đối tượng”)

#### ▪ Ví dụ

```
Sub Example_SelectOnScreen ()
  On Error Resume Next
  Dim SelectionSetObj As AcadSelectionSet                                'Line 1
  Set SelectionSetObj = ThisDrawing.SelectionSets ("Kira")              'Line 2
  If Err <> 0 Then                                                       'Line 3
    Err.Clear                                                            'Line 4
    Set SelectionSetObj = ThisDrawing.SelectionSets.Add ("Kira")       'Line 5
  Else
    SelectionSetObj.Clear                                              'Line 6
  End If
  SelectionSetObj.SelectOnScreen                                       'Line 7
End Sub
```

### 8.3.4. Thao tác với các đối tượng trong SelectionSet

Mục đích chính của đối tượng SelectionSet là nhóm các đối tượng hình học lại để hiệu chỉnh các đối tượng trong đó một cách dễ dàng hơn. Do vậy để truy cập đến các đối tượng trong nằm trong **SelectionSet** ta sử dụng cấu trúc “**For Each...Next**”

#### ▪ Ví dụ

Ví dụ này sẽ yêu cầu người dùng chọn các đối tượng trên màn hình bản vẽ, sau đó đổi màu các đối tượng được chọn thành màu đỏ.

```
Sub Example_ChangeColor_Red()
    On Error Resume Next
    Dim ssetObj As AcadSelectionSet
    Set ssetObj = ThisDrawing.SelectionSets("MySelectionSet")
    If Err <> 0 Then
        Err.Clear
        Set ssetObj = ThisDrawing.SelectionSets.Add("MySelectionSet")
    Else
        ssetObj.Clear
    End If
    '-----
    Dim StrPrompt As String
    StrPrompt = vbCrLf & "Select Object to Change Color:"
    ThisDrawing.Utility.Prompt StrPrompt
    ssetObj.SelectOnScreen
    Dim ent As AcadEntity
    For Each ent In ssetObj
        ent.color = acRed
        ent.Update
    Next ent
End Sub
```

### 8.4. Bộ lọc cho đối tượng SelectionSet

Trong quá trình thao tác với đối tượng **SelectionSet**, đôi khi ta muốn giới hạn việc chọn đối tượng theo một điều kiện nào đó như chỉ chọn các đối tượng **Line, Circle...** hoặc đơn giản chọn các đối tượng thuộc một Layer nào đó hoặc đối tượng có màu nào đó....Để thực hiện điều này cần xác định “Kiểu bộ lọc” và “Đối tượng lọc”.Kiểu bộ lọc chính là tham số **FilterType** và Đối tượng lọc chính là **FilterData**.

- **FilterType** : Kiểu Integer, là mảng 1 chiều chứa mã Dxf xác định “Kiểu lọc đối tượng”.
- **FilterData** : Kiểu Variant, là mảng một chiều chứa “Đối tượng lọc”.

**Chú ý :** Quan hệ giữa **FilterType** và **FilterData** là quan hệ 1:1 nên số phần tử trong mảng **FilterType** và **FilterData** phải bằng nhau.



### Danh sách các mã Dxf thường được sử dụng

Mã Dxf	Ý nghĩa
0	Kiểu String – Thể hiện loại đối tượng như: Line, PolyLine, LWPolyline, Spline, Circle, Arc, Text, Mtext, Dimension...
1	Chuỗi văn bản chứa trong đối tượng Text, Mtext.
2	Kiểu String – Thể hiện tên của Block.
6	Kiểu String – Thể hiện LineType của các đối tượng.
7	Kiểu String – Tên TextStyle
8	Kiểu String – Thể hiện tên của Layer
10	Tọa độ chính của một đối tượng như: StartPoint của Line, điểm chèn của Text, CenterPoint của Circle...
62	Kiểu Integer – Xác định màu của đối tượng: 0=ByBlock, 1=Red, 2=Yellow....256=ByLayer.
67	Kiểu Integer – Xác định không gian (0 = ModelSpace ; 1 = PaperSpace)
-4	Các toán tử điều kiện (Sử dụng khi kết hợp nhiều điều kiện lọc).

#### 8.4.1. Lọc theo một điều kiện

Khi lọc theo một điều kiện thì hai tham số **FilterType** và **FilterData** là mảng một chiều 1 phần tử.

- **Ví dụ** Đoạn mã sau sẽ chọn duy nhất đối tượng Line trong bản vẽ sau đó sẽ đổi tất cả các Line được chọn này thành màu đỏ.

```

Sub Example_Filter_OneCondition()
On Error Resume Next
Dim SelectionSetObj As AcadSelectionSet
Set SelectionSetObj = ThisDrawing.SelectionSets("Kira")
If Err <> 0 Then
    Err.Clear
    Set SelectionSetObj = ThisDrawing.SelectionSets.Add("Kira")
Else
    SelectionSetObj.Clear
End If
'-----
Dim FilterType(0) As Integer
Dim FilterData(0) As Variant
FilterType(0) = 0: FilterData(0) = "Line"
SelectionSetObj.SelectOnScreen FilterType, FilterData
Dim Ent As AcadLine
For Each Ent In SelectionSetObj
    Ent.color = acRed
    Application.Update
Next Ent
End Sub

```

#### 8.4.2. Lọc theo nhiều điều kiện

Khi cần lọc đối tượng theo nhiều điều kiện kết hợp ta sử dụng “**Toán tử điều kiện**” vào trong bộ lọc, lúc này các điều kiện lọc sẽ được đặt giữa các “**Toán tử điều kiện**”.

Toán tử	Giá trị bắt đầu	Giá trị kết thúc	Số lượng điều kiện lọc	Ý nghĩa
AND	“<And”	“>And”	>=1	Đối tượng nào thỏa mãn tất cả các điều kiện lọc sẽ được chọn
OR	“<Or”	“>Or”	>=1	Đối tượng nào thỏa mãn một trong các điều kiện sẽ được chọn.
XOR	“<Xor”	“>Xor”	2	Đối tượng nào thỏa mãn một điều kiện và không thỏa mãn điều kiện còn lại sẽ được chọn.
NOT	“<Not”	“>Not”	1	Đối tượng nào không thỏa mãn điều kiện lọc sẽ được chọn

- **Ví dụ:**

Đoạn mã sau sẽ lọc theo nhiều điều kiện, những đối tượng nào là đường thẳng hoặc đường tròn sẽ được chọn, sau đó sẽ đổi màu chúng thành màu đỏ.

**Tóm lược điều kiện**

“<Or” (Line, Circle) “Or>”

**Triển khai dạng code**

```
Sub Example_Filter_MultiCondition()
    Dim ssetObj As AcadSelectionSet
    Set ssetObj = ThisDrawing.SelectionSets("MySelectionSet")
    If Err <> 0 Then
        Err.Clear
        Set ssetObj = ThisDrawing.SelectionSets.Add("MySelectionSet")
    Else
        ssetObj.Clear
    End If
    '-----
    Dim FilterType(0 To 3) As Integer
    Dim FilterData(0 To 3) As Variant
    FilterType(0) = -4: FilterData(0) = "<Or"
    FilterType(1) = 0: FilterData(1) = "Line"
    FilterType(2) = 0: FilterData(2) = "Circle"
    FilterType(3) = -4: FilterData(3) = "Or>"
    ssetObj.SelectOnScreen FilterType, FilterData
    Dim Ent As AcadEntity
    For Each Ent In ssetObj
        Ent.Highlight (True)
    Next Ent
End Sub
```

- **Ví dụ:**

Đoạn mã sau sẽ chọn những đối tượng nào là Text hoặc Mtext nhưng không thuộc Layer có tên “DefPoints”, sau đó đổi chúng về Layer có tên “Text\_3”

**Tóm tắt điều kiện**

“<And”(“<Or”(Text, Mtext)”Or>),”<Not”(Defpoints)”Not>”)”And>”

**Triển khai dạng code**

```

Sub Ex2_Filter_MultiCondition()
    Dim ssetObj As AcadSelectionSet
    Set ssetObj = ThisDrawing.SelectionSets("MySelectionSet")
    If Err <> 0 Then
        Err.Clear
        Set ssetObj = ThisDrawing.SelectionSets.Add("MySelectionSet")
    Else
        ssetObj.Clear
    End If
    '-----
    Dim FilterType(0 To 8) As Integer
    Dim FilterData(0 To 8) As Variant
    FilterType(0) = -4: FilterData(0) = "<And"
    FilterType(1) = -4: FilterData(1) = "<Or"
    FilterType(2) = 0: FilterData(2) = "Text"
    FilterType(3) = 0: FilterData(3) = "Mtext"
    FilterType(4) = -4: FilterData(4) = "Or>"
    FilterType(5) = -4: FilterData(5) = "<Not"
    FilterType(6) = 8: FilterData(6) = "Defpoints"
    FilterType(7) = -4: FilterData(7) = "Not>"
    FilterType(8) = -4: FilterData(8) = "And>"
    ssetObj.SelectOnScreen FilterType, FilterData
    '-----
    Dim Ent As AcadEntity
    For Each Ent In ssetObj
        Ent.Layer = "Text_3"
    Next Ent
End Sub

```

## 8.5. Một số phương thức khác của đối tượng SelectionSet

### 8.5.1. Phương thức RemoveItems

Phương thức này sẽ loại bỏ một hoặc nhiều đối tượng ra khỏi **SelectionSet**. Các đối tượng này sẽ không nằm trong **SelectionSet** nữa NHƯNG vẫn tồn tại trong bản vẽ.

#### ▪ Cú pháp

SelectionSetObj.**RemoveItems** ArrayObj

#### ▪ Giải thích

Tham số	Giải thích
SelectionSetObj	Kiểu acadSelectionSet – Biến chứa đối tượng SelectionSet được tạo ra
ArrayObj	Kiểu Variant - mảng chứa các đối tượng được thêm vào SelectionSet

- **Ví dụ:**

Đoạn code sau sẽ thực hiện việc tạo đối Circle và đối tượng Ellipse trong ModelSpace, sau đó sẽ thêm tất cả các đối tượng trong ModelSpace vào SelectionSetObj (Kể cả 2 đối tượng Circle và Ellipse vừa mới tạo) tiếp theo đó sẽ Remove 2 đối tượng Circle và Ellipse này ra khỏi SelectionSet.

```

Sub Ex_RemoveItems ()
    On Error Resume Next                                     'Line 1
    Dim SelectionSetObj As AcadSelectionSet                 'Line 2
    Set SelectionSetObj = ThisDrawing.SelectionSets("Kira") 'Line 3
    If Err <> 0 Then                                       'Line 4
        Err.Clear                                         'Line 5
        Set SelectionSetObj = ThisDrawing.SelectionSets.Add("Kira") 'Line 6
    Else
        SelectionSetObj.Clear                             'Line 7
    End If
    '-----
    ' Create a circle object in model space
    Dim CircObj As AcadCircle                             'Line 8
    Dim CenterPt(0 To 2) As Double                       'Line 9
    Dim Radius As Double                                 'Line 10
    CenterPt(0) = 20: CenterPt(1) = 30: CenterPt(2) = 0 'Line 11
    Radius = 3                                           'Line 12
    Set CircObj = ThisDrawing.ModelSpace.AddCircle(CenterPt, Radius) 'Line 13
    ' Create an Ellipse object in model space
    Dim EllObj As AcadEllipse                             'Line 14
    Dim MajAxis(0 To 2) As Double                       'Line 15
    Dim Center(0 To 2) As Double                       'Line 16
    Dim RadRatio As Double                              'Line 17
    Center(0) = 5: Center(1) = 5: Center(2) = 0        'Line 18
    MajAxis(0) = 10: MajAxis(1) = 20: MajAxis(2) = 0   'Line 19
    RadRatio = 0.3                                       'Line 20
    Set EllObj = ThisDrawing.ModelSpace.AddEllipse(Center, MajAxis, RadRatio) 'Line 21
    ReDim Ssobjs(0 To ThisDrawing.ModelSpace.Count - 1) As AcadEntity 'Line 22
    Dim i As Integer                                     'Line 23
    For i = 0 To ThisDrawing.ModelSpace.Count - 1      'Line 24
        Set Ssobjs(i) = ThisDrawing.ModelSpace.Item(i) 'Line 25
    Next
    SelectionSetObj.AddItem Ssobjs                      'Line 26
    ' Remove two of the objects from the selection set
    Dim ArrayObj(0 To 1) As AcadEntity                 'Line 27
    Set ArrayObj(0) = EllObj                           'Line 28
    Set ArrayObj(1) = CircObj                          'Line 29
    SelectionSetObj.RemoveItems ArrayObj               'Line 30
End Sub

```

### 8.5.2. Phương thức Clear

Khi phương thức **Clear** được thực thi nó sẽ loại bỏ tất cả các đối tượng ra khỏi **SelectionSet** NHƯNG bản thân đối tượng **SelectionSet** và các đối tượng đó vẫn tồn tại trong bản vẽ. Đối tượng **SelectionSet** lúc này trở thành một tập rỗng.

- **Cú pháp**

```
SelectionSetObj.Clear
```

### 8.5.3. Phương thức Eraser

Khi phương thức **Eraser** được thực thi nó sẽ loại bỏ tất cả các đối tượng ra khỏi **SelectionSet** đồng thời xóa luôn các đối tượng này khỏi bản vẽ NHƯNG đối tượng **SelectionSet** vẫn còn tồn tại. Đối tượng **SelectionSet** lúc này là một tập rỗng.

- **Cú pháp**

```
SelectionSetObj.Eraser
```

### 8.5.4. Phương thức Delete

Khi phương thức **Delete** được thực thi nó sẽ xóa đối tượng **SelectionSet** ra khỏi bản vẽ NHƯNG các đối tượng nằm trong **SelectionSet** vẫn còn tồn tại trong bản vẽ.

- **Cú pháp**

```
SelectionSetObj.Delete
```

## Bài Tập chủ đề: Làm việc với đối tượng SelectionSet

**Bài 1:**

Khai báo và khởi tạo đối tượng SelectionSetObj có tên “PTSCMC”, đồng thời sử dụng bộ lọc 1 điều kiện để chọn tất cả các Block có tên do người dùng nhập vào.

**Bài 2:**

Khai báo và khởi tạo đối tượng SelectionSet có tên “Design\_Team”, đồng thời sử dụng bộ lọc nhiều điều kiện kết hợp để chọn đối tượng Line có màu Xanh Lá và không thuộc Layer có tên “Defpoints”. Sau đó thực hiện thao tác Delete tất cả các Line vuông góc với trục Ox.

**Bài 3:**

Thông thường các kích thước nằm rải rác ở các Layer khác nhau. Hãy viết chương trình tạo ra một đối tượng SelectionSet có tên “Dimension” và chuyển chúng sang một Layer mới. Chương trình sẽ hỏi người sử dụng tên Layer mới này.