

### 25.2.3 Klassen Point und PointF

Die Klasse *Point* (gb.clipper) beschreibt *einen* Punkt mit *ganzzahligen* Koordinaten. In diesem Kapitel werden Eigenschaften und Methoden der Klasse Point sowie einige Beispiele für den Einsatz der Klasse vorgestellt.

#### 25.2.3.1 Eigenschaften

Die Klasse *Point* verfügt über diese beiden Eigenschaften:

Eigenschaft	Beschreibung
X As Integer	Setzt die x-Koordinate des Punktes oder gibt den Wert zurück.
Y As Integer	Setzt die y-Koordinate des Punktes oder gibt den Wert zurück.

Tabelle 25.2.3.1.1 : Eigenschaften der Klasse Point

#### 25.2.3.2 Methoden

Für die Klasse *Point* werden hier die zwei Methoden beschrieben.

Methode	Beschreibung
Copy( ) As Point	Gibt eine Kopie des aktuellen Punktes zurück.
InRect( Rectangle As Rect ) As Boolean	Gibt True zurück, wenn der Punkt im angegebenen Rechteck liegt.

Tabelle 25.2.3.2.1 : Ausgewählte Methoden der Klasse Point

#### 25.2.3.3 Beispiele

Die Klasse *Point* ist erstellbar. Sie können die Klasse wie eine (statische) Funktion benutzen:

```
Dim PointP, PointQ As Point
PointP = New Point() ' P(0|0)
PointQ = New Point(44,-22)
```

Im folgenden Beispiel werden Punkte mit zufälligen Koordinaten (aus einem eingeschränkten Bereich) und einer Zufallsfarbe erzeugt und auf ein Picture gezeichnet, das dann in einer DrawArea angezeigt wird:

```
Public Sub SetPoint(X As Integer, Y As Integer, cColor As Integer)
    Paint.AntiAlias = False
    Paint.FillRect(X, Y, 1, 1, cColor)
    Paint.AntiAlias = True
End ' SetPoint(..)

Public Sub ScriptPointCloud()
    Dim i As Integer
    Dim PointP As Point ' Klasse 'Point' in gb.clipper

    GenerateNewPicture()
    SetPictureBorder()

    Paint.Begin(hPicture)
    Paint.Translate(xTranslate, yTranslate)
    Paint.Scale(xScale, yScale) ' +y ▲
    Paint.AntiAlias = False
    DrawCoordinateSystem() ' +y ▲
    For i = 1 To 20000
        ' Punkte mit zufälligen Koordinaten und einer Zufallsfarbe
        PointP = New Point(Rnd(10, 550 - 5), Rnd(10, 260))
        SetPoint(PointP.X, PointP.Y, Color.RGB(Rnd(0, 255), Rnd(0, 255), Rnd(0, 255)))
    Next
    Paint.AntiAlias = True
    Paint.End
End ' ScriptPointCloud()
```

Wenn Sie genau zählen, müssten Sie auf 20000 Punkte kommen:

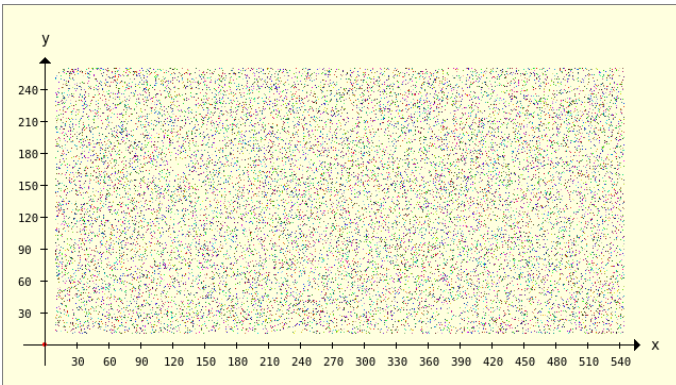


Abbildung 25.2.3.3.1: Punktwolke

Das abschließende Beispiel demonstriert, wie Sie feststellen können, ob ein vorgegebener Punkt in einem bestimmten Rechteck liegt.

```
Dim PointP As Point
Dim RectRed As Rect

PointP = New Point(260, 150)
RectRed = New Rect(60, 60, 400, 180)

Print "P in RectRed? --> "; PointP.InRect(RectRed) ' Ergebnis: TRUE
```

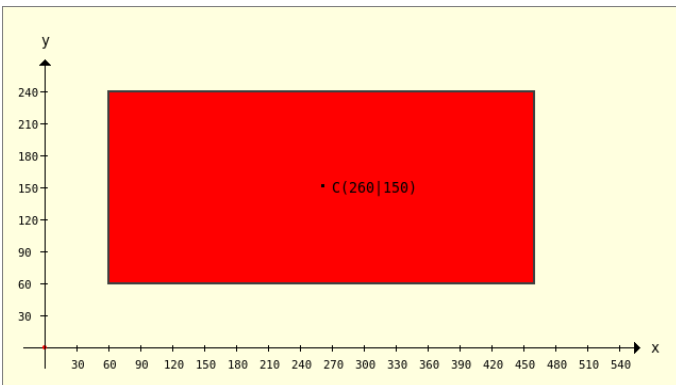


Abbildung 25.2.3.3.2: Gegenseitige Lage von Punkt und Rechteck

### 25.2.3.4 Klasse PointF

Während für die Klasse *Point* nur ganze Zahlen als Koordinaten zulässig sind, können Sie für die Klasse *PointF* für die Koordinaten eines Punktes reelle Zahlen verwenden.