

در ادامه پست [پیاده سازی پروژه نقاشی \(Paint\) به صورت شی گرا #5](#) ، در این پست به تشریح کلاس دایره و بیضی می‌پردازیم.

ابتدا به تشریح کلاس ترسیم بیضی (Ellipse) می‌پردازیم.

```
using System.Drawing;

namespace PWS.ObjectOrientedPaint.Models
{
    /// <summary>
    /// Ellipse Draw
    /// </summary>
    public class Ellipse : Shape
    {
        #region Constructors (2)

        /// <summary>
        /// Initializes a new instance of the <see cref="Ellipse" /> class.
        /// </summary>
        /// <param name="startPoint">The start point.</param>
        /// <param name="endPoint">The end point.</param>
        /// <param name="zIndex">Index of the z.</param>
        /// <param name="foreColor">Color of the fore.</param>
        /// <param name="thickness">The thickness.</param>
        /// <param name="isFill">if set to <c>true</c> [is fill].</param>
        /// <param name="backgroundColor">Color of the background.</param>
        public Ellipse(PointF startPoint, PointF endPoint, int zIndex, Color foreColor, byte thickness,
            bool isFill, Color backgroundColor)
            : base(startPoint, endPoint, zIndex, foreColor, thickness, isFill, backgroundColor)
        {
            ShapeType = ShapeType.Ellipse;
        }

        /// <summary>
        /// Initializes a new instance of the <see cref="Ellipse" /> class.
        /// </summary>
        public Ellipse()
        {
            ShapeType = ShapeType.Ellipse;
        }

        #endregion Constructors

        #region Methods (1)

        // Public Methods (1)

        /// <summary>
        /// Draws the specified g.
        /// </summary>
        /// <param name="g">The g.</param>
        public override void Draw(Graphics g)
        {
            if (IsFill)
            {
                g.FillEllipse(BackgroundBrush, StartPoint.X, StartPoint.Y, Width, Height);
                g.DrawEllipse(Pen, StartPoint.X, StartPoint.Y, Width, Height);
            }
            base.Draw(g);
        }

        #endregion Methods
    }
}
```

این کلاس از شی Shape ارث برده و دارای دو سازنده ساده می‌باشد که نوع شی ترسیمی را مشخص می‌کنند، در متد Draw نیز با توجه به توپر یا توخالی بودن شی ترسیم آن انجام می‌شود، در این کلاس باید متد **HasPointInShape** بازنویسی (override) شود، در این متد باید تعیین شود که یک نقطه در داخل بیضی قرار گرفته است یا خیر که متاسفانه فرمول بیضی خاطرمد نبود. البته به

صورت پیش فرض نقطه با توجه به چهارگوشی که بیضی را احاطه می کند سنجیده می شود.

کلاس دایره (Circle) از کلاس بالا (Ellipse) ارث بری دارد که کد آن را در زیر مشاهده می نمایید.

```
using System;
using System.Drawing;

namespace PWS.ObjectOrientedPaint.Models
{
    /// <summary>
    /// Circle
    /// </summary>
    public class Circle : Ellipse
    {
        #region Constructors (2)

        /// <summary>
        /// Initializes a new instance of the <see cref="Circle" /> class.
        /// </summary>
        /// <param name="startPoint">The start point.</param>
        /// <param name="endPoint">The end point.</param>
        /// <param name="zIndex">Index of the z.</param>
        /// <param name="foreColor">Color of the fore.</param>
        /// <param name="thickness">The thickness.</param>
        /// <param name="isFill">if set to <c>true</c> [is fill].</param>
        /// <param name="backgroundColor">Color of the background.</param>
        public Circle(PointF startPoint, PointF endPoint, int zIndex, Color foreColor, byte thickness,
            bool isFill, Color backgroundColor)
        {
            float x = 0, y = 0;
            float width = Math.Abs(endPoint.X - startPoint.X);
            float height = Math.Abs(endPoint.Y - startPoint.Y);
            if (startPoint.X <= endPoint.X && startPoint.Y <= endPoint.Y)
            {
                x = startPoint.X;
                y = startPoint.Y;
            }
            else if (startPoint.X >= endPoint.X && startPoint.Y >= endPoint.Y)
            {
                x = endPoint.X;
                y = endPoint.Y;
            }
            else if (startPoint.X >= endPoint.X && startPoint.Y <= endPoint.Y)
            {
                x = endPoint.X;
                y = startPoint.Y;
            }
            else if (startPoint.X <= endPoint.X && startPoint.Y >= endPoint.Y)
            {
                x = startPoint.X;
                y = endPoint.Y;
            }
            StartPoint = new PointF(x, y);
            var side = Math.Max(width, height);
            EndPoint = new PointF(x + side, y + side);
            ShapeType = ShapeType.Circle;
            Zindex = zIndex;
            ForeColor = foreColor;
            Thickness = thickness;
            BackgroundColor = backgroundColor;
            IsFill = isFill;
        }

        /// <summary>
        /// Initializes a new instance of the <see cref="Circle" /> class.
        /// </summary>
        public Circle()
        {
            ShapeType = ShapeType.Circle;
        }

        #endregion Constructors

        #region Methods (1)

        // Public Methods (1)

        /// <summary>
        /// Points the in sahpe.

```

```

/// </summary>
/// <param name="point">The point.</param>
/// <param name="tolerance">The tolerance.</param>
/// <returns>
/// <c>true</c> if [has point in sahpe] [the specified point]; otherwise, <c>false</c>.
/// </returns>
public override bool HasPointInSahpe(PointF point, byte tolerance = 5)
{
    float width = Math.Abs(EndPoint.X+tolerance - StartPoint.X-tolerance);
    float height = Math.Abs(EndPoint.Y+tolerance - StartPoint.Y-tolerance);
    float diagonal = Math.Max(height, width);
    float raduis = diagonal / 2;
    float dx = Math.Abs(point.X - (X + Width / 2));
    float dy = Math.Abs(point.Y - (Y + height / 2));
    return (dx + dy <= raduis);
}

#endregion Methods
}
}

```

این کلاس شامل دو سازنده می‌باشد، که در سازنده اول با توجه به نقاط ابتدا و انتهای ترسیم شکل مقدار طول و عرض مستطیل احاطه کننده دایره محاسبه شده و با توجه به آنها بزرگترین ضلع به عنوان قطر دایره در نظر گرفته می‌شود و EndPoint شکل مورد نظر تعیین می‌شود.

در متد **HasPointInShape** با استفاده از فرمول دایره تعیین می‌شود که آیا نقطه پارامتر ورودی متد در داخل دایره واقع شده است یا خیر (جهت انتخاب شکل برای جابجایی یا تغییر اندازه). در پست‌های بعد به پیاده سازی اینترفیس نرم افزار خواهیم پرداخت.

موفق و موید باشید

در ادامه مطالب قبل:

- [پیاده سازی پروژه نقاشی \(Paint\) به صورت شی گرا #1](#)
- [پیاده سازی پروژه نقاشی \(Paint\) به صورت شی گرا #2](#)
- [پیاده سازی پروژه نقاشی \(Paint\) به صورت شی گرا #3](#)
- [پیاده سازی پروژه نقاشی \(Paint\) به صورت شی گرا #4](#)
- [پیاده سازی پروژه نقاشی \(Paint\) به صورت شی گرا #5](#)

نظرات خوانندگان

نویسنده: بتیسا
تاریخ: ۱۳۹۱/۱۲/۰۷ ۹:۸

با سلام برای پیدا کردن نقطه در بیضی من چند لینک پیدا کردم امیدوارم که به کارتون بیاد

لینک اول از [ویکی پدیا](#)

لینک دوم از [stackoverflow](#)

لینک سوم [mathforum](#)

لینک چهارم [mathopenref](#)