

# xcharts- -



Settings

Background

Image

BarChart001\_Blue

Image Type

Simple

Image Color

Auto Color

Border Style

Title

Text

Area Line

Sub Text

Item Gap

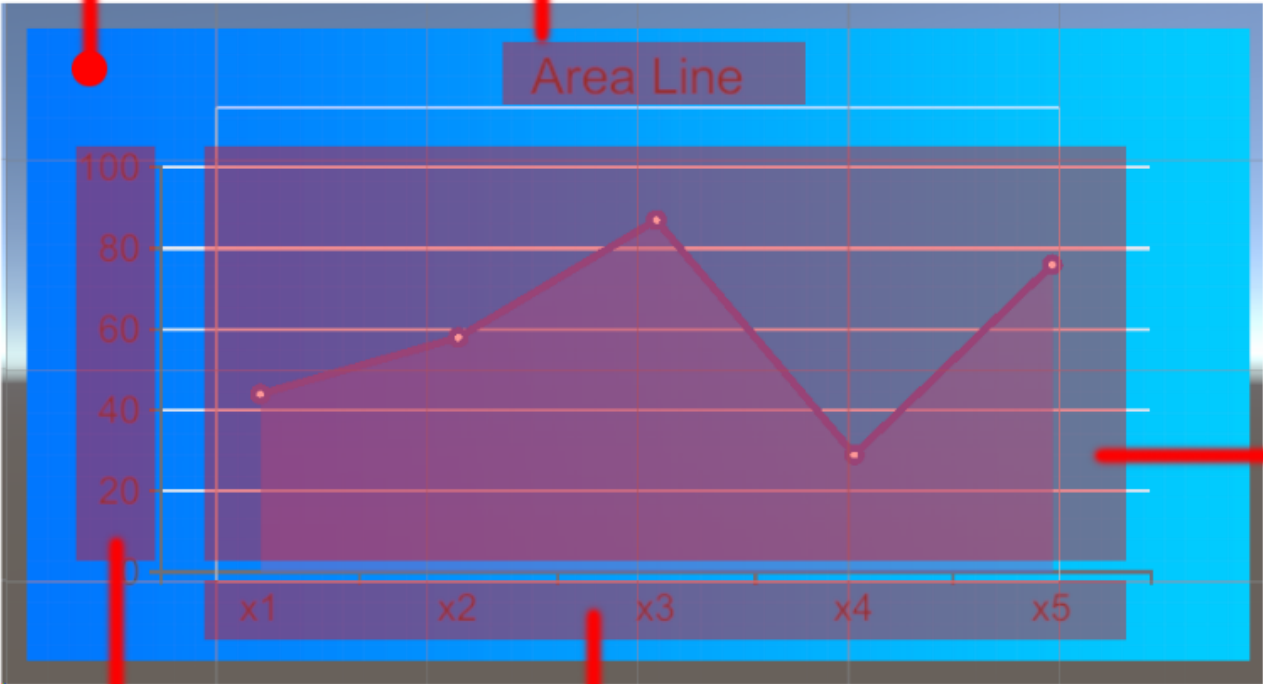
0

Location

Top Center

Label Style

Sub Label Style

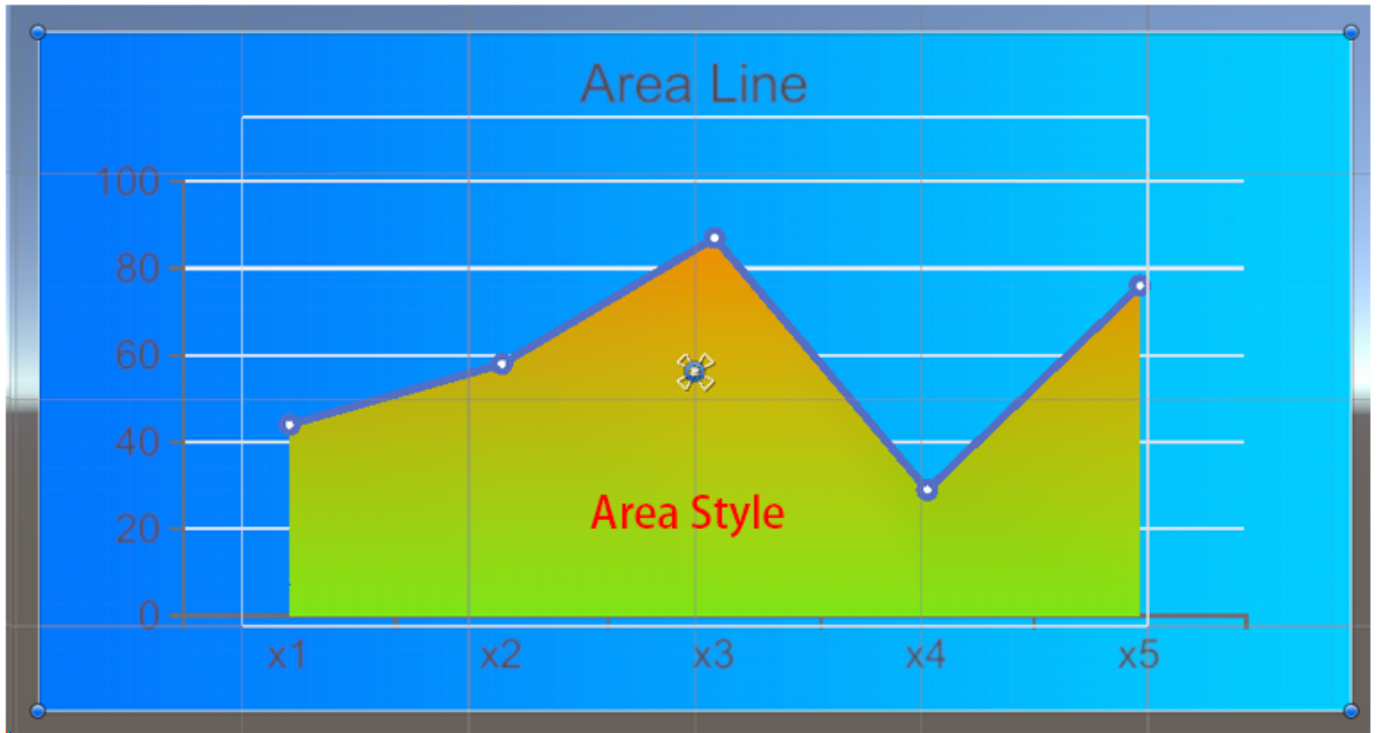


Y Axis

X Axis

Serie

Area Style



▼ Serie 0: Line

Serie Name

serie0

Coord System

GridCoord

State

Normal

Stack

X Axis Index

0

Y Axis Index

0

Line Type

Normal

Clip

☐

► More

► Symbol

☒

► Line Style

☒

► Item Style

☒

► Animation

☒

▼ AreaStyle 0

Origin

Auto

Color

To Color

Opacity

1

To Top

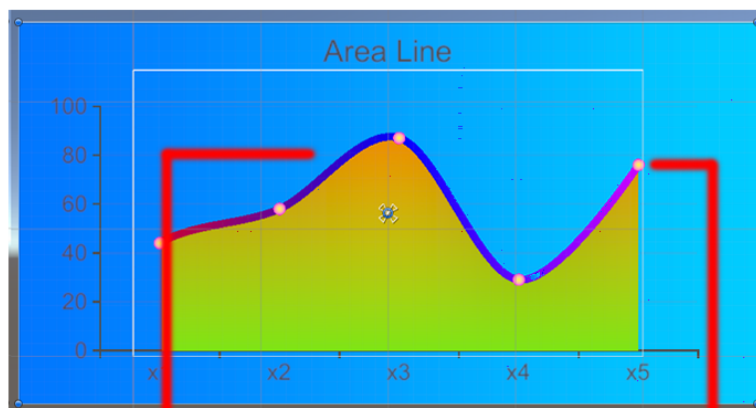
☒

Inner Fill

☐

► Data





## 平滑曲线

▼ Serie 0: Line

✓	
Serie Name	产值
Coord System	GridCoord
State	Normal
Stack	
X Axis Index	0
Y Axis Index	0
Line Type	Smooth
Smooth Limit	<input type="checkbox"/>
Clip	<input type="checkbox"/>

## 曲线上的点的样式

▼ Symbol

✓	
Type	Circle
Gap	0
Size Type	Custom
Size	0
Start Index	0
Interval	0
Force Show Last	<input type="checkbox"/>
Repeat	<input type="checkbox"/>
Line Style	✓

## 曲线上点的颜色

▼ Item Style

✓	
Color	
Color 0	
To Color	
To Color 2	
Mark Color	
Background Color	
Background Width	0
Center Color	
Center Gap	0
Border Width	0
Border Gap	0
Border Color	
Border Color 0	
Border To Color	
Opacity	1
Item Marker	
Item Formatter	
Numeric Formatter	
Corner Radius	4

## 线条颜色

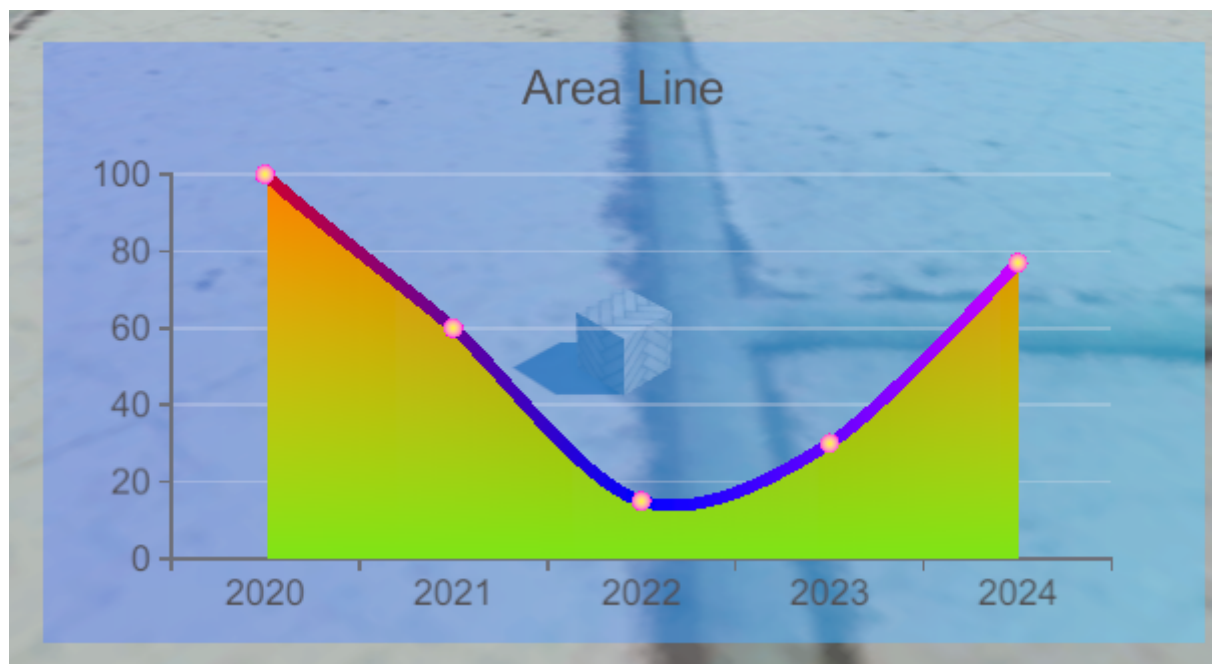
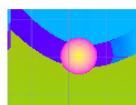
▼ Line Style

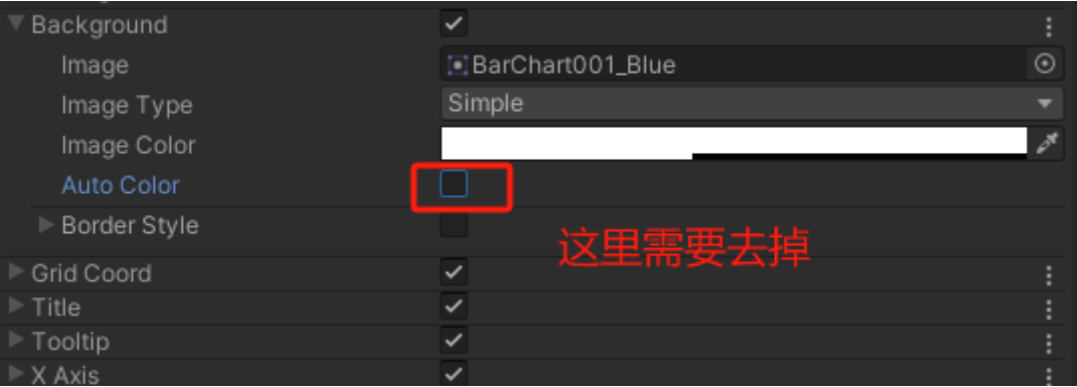
✓	
Type	Solid
Color	
To Color	
To Color 2	
Width	1
Length	0
Opacity	<input type="checkbox"/>
Dash Length	4
Dot Length	2
Gap Length	2

## 轴的辅助颜色

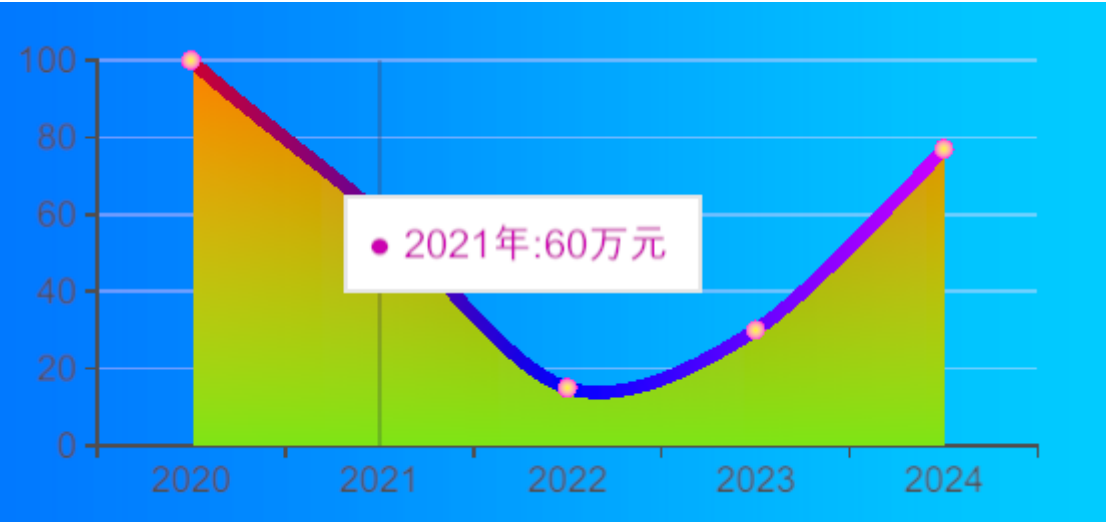
▼ Split Line

✓	
Interval	0
Distance	0
Auto Color	<input type="checkbox"/>
Show Start Line	✓
Show End Line	✓
Show Z Line	✓
Line Style	None
Type	
Color	
To Color	
To Color 2	





{.}{b}={c}



```
/// || \n {i}title  
/// {a}{b}{c}{d}{e}{f}{g}<br/>  
/// {.}index[0]serie<br/>  
/// {a}index[0]serie[0]name<br/>  
/// {b}index[0]serie[0]serieData[name]X<br/>  
/// {c}index[0]serie[y]dimesion[1]<br/>  
/// {d}index[0]serie[y]dimesion[1]<br/>  
/// {e}index[0]serie[0]serieData[name]<br/>  
/// {h}index[0]serie[0]serieData<br/>  
/// {f}<br/>  
/// {g}<br/>  
/// {y}value[y]<br/>  
/// {.1}index[1]serie<br/>  
/// {a1}{b1}{c1}[1]index[1]serie<br/>  
/// {c1:2}[1]serie[0]index[2]<br/>  
/// {c1:2-2}[1]serie[0]<br/>
```

```

/// {d1: 2: f2}f2numericFormatter<br/>
/// {d: 0. ##} 0. ## f2 '100.00%' <br/>
/// {a}: {c}" {a1}: {c1: f1}" {a1}: {c1: 0: f1}" {a1}: {c1: 1-1: f1}"

```



```

public class LoadData : MonoBehaviour
{
    private Color initialColor = Color.white;
    private Color targetColor = Color.white;
    private Color currentColor = Color.white;
    public float updateInterval = 1.0f;
    private XCharts.Runtime.LineChart lineChart;
    [Title("")]
    public GameObject TargetObject;

    // Start is called before the first frame update
    void Start()
    {
        if (TargetObject != null)
        {
            // tag
            lineChart = TargetObject.GetComponent<XCharts.Runtime.LineChart>();
            lineChart.ClearData();
            lineChart.AddXAxisData("2020");
            lineChart.AddXAxisData("2021");
            lineChart.AddXAxisData("2022");
            lineChart.AddXAxisData("2023");
            lineChart.AddXAxisData("2024");

            lineChart.AddData(0, 0, 100);
            lineChart.AddData(0, 1, 60);
            lineChart.AddData(0, 2, 15);
            lineChart.AddData(0, 3, 30);
            lineChart.AddData(0, 4, 77);

            // 
            initialColor = new Color(1.0f, 0.0f, 1.0f); // 
            currentColor = initialColor;

```

```

        targetColor = new Color(1.0f, 1.0f, 1.0f); // 白
    }

}

// Update is called once per frame
void Update()
{
    if (lineChart != null)
    {
        // 第一seriesのitem styleのbackground color
        var serie = lineChart.GetSerie(0);
        if (serie != null)
        {
            var itemStyle = serie.itemStyle;
            float time = Mathf.PingPong(Time.time / updateInterval, 1.0f);
            currentColor = Color.Lerp(initialColor, targetColor, time);
            itemStyle.backgroundColor = currentColor;
            serie.itemStyle = itemStyle;
        }
    }
}
}

```

実行 #10

実行日時 15 10 2025 10:01:51

実行日時 16 10 2025 15:06:27