SPRINT-3

TEAM ID	PNT2022TMID10960
PROJECT NAME	INDUSTRY - SPECIFIC INTELLIGENT FIRE MANAGEMENT SYSTEM
IBM ID	IBM-Project-6081-1658823192

PROGRAM:

```
{
        type: 'gauge',
        center: ['50%', '60%'],
        startAngle: 200,
        endAngle: -20,
        min: 0,
        max: 60,
        itemStyle: {
          color: '#FD7347'
        },
        progress: {
          show: true,
          width: 8
        },
        pointer: {
          show: false
        },
        axisLine: {
          show: false
        },
        axisTick: {
          show: false
```

```
    },
    splitLine: {
      show: false
    },
    axisLabel: {
      show: false
    },
    detail: {
      show: false
    },
    data: [
     {
       value: 30
     }
    ]
   }
 ]
};
setInterval(function () {
  const random = Math.floor(Math.random() * 5) + 30;
  myChart.setOption({
    series: [
     {
       data: [
        {
          value: random
        }
       ]
     },
     {
       data: [
```

```
          {
            value: random
          }
        ]
      }
    ]
  });
 }, 2000);

 if (option && typeof option === 'object') {
  myChart.setOption(option);
 }

 window.addEventListener('resize', myChart.resize);
}

function n2oGasCensorChart() {

 var dom = document.getElementById('n2o-chart-container');
 var myChart = echarts.init(dom, 'null', {
  renderer: 'canvas',
  useDirtyRect: false
 });
 var app = {};

 var option;
 option = {
  series: [
   {
    type: 'gauge',
    axisLine: {
```

```
  lineStyle: {
    width: 30,
    color: [
      [0.3, '#67e0e3'],
      [0.7, 'orange'],
      [1, '#fd666d']
    ]
  }
},
pointer: {
  itemStyle: {
    color: 'auto'
  }
},
axisTick: {
  distance: -30,
  length: 8,
  lineStyle: {
    color: '#fff',
    width: 2
  }
},
splitLine: {
  distance: -30,
  length: 30,
  lineStyle: {
    color: '#fff',
    width: 4
  }
},
axisLabel: {
```

```
        color: 'auto',
        distance: 40
      },
      detail: {
        valueAnimation: true,
        formatter: '{value} °C',
        color: 'auto',
        fontSize: 14
      },
      data: [
        {
          value: 85
        }
      ]
    }
  ]
};
setInterval(function () {
  myChart.setOption({
    series: [
      {
        data: [
          {
            value: Math.floor(Math.random() * 5) + 80
          }
        ]
      }
    ]
  });
}, 2000);
```

```
  if (option && typeof option === 'object') {
    myChart.setOption(option);
  }

  window.addEventListener('resize', myChart.resize);
}

function cmoGasCensorChart() {

  var dom = document.getElementById('cmo-chart-container');
  var myChart = echarts.init(dom, 'null', {
    renderer: 'canvas',
    useDirtyRect: false
  });
  var app = {};

  var option;
  option = {
    series: [
      {
        type: 'gauge',
        axisLine: {
          lineStyle: {
            width: 30,
            color: [
              [0.3, '#67e0e3'],
              [0.7, 'orange'],
              [1, '#fd666d']
            ]
          }
        },
```

```
pointer: {
  itemStyle: {
    color: 'auto'
  }
},
axisTick: {
  distance: -30,
  length: 8,
  lineStyle: {
    color: '#fff',
    width: 2
  }
},
splitLine: {
  distance: -30,
  length: 30,
  lineStyle: {
    color: '#fff',
    width: 4
  }
},
axisLabel: {
  color: 'auto',
  distance: 40
},
detail: {
  valueAnimation: true,
  formatter: '{value} °C',
  color: 'auto',
  fontSize: 14
},
```

```
        data: [
          {
            value: 30
          }
        ]
      }
    ]
  };
  setInterval(function () {
    myChart.setOption({
      series: [
        {
          data: [
            {
              value: Math.floor(Math.random() * 5) + 50
            }
          ]
        }
      ]
    });
  }, 2000);

  if (option && typeof option === 'object') {
    myChart.setOption(option);
  }

  window.addEventListener('resize', myChart.resize);
}

function co2GasCensorChart() {
```

```
var dom = document.getElementById('co2-chart-container');
var myChart = echarts.init(dom, 'null', {
  renderer: 'canvas',
  useDirtyRect: false
});
var app = {};

var option;
option = {
  series: [
    {
      type: 'gauge',
      axisLine: {
        lineStyle: {
          width: 30,
          color: [
            [0.3, '#67e0e3'],
            [0.7, 'orange'],
            [1, '#fd666d']
          ]
        }
      },
      pointer: {
        itemStyle: {
          color: 'auto'
        }
      },
      axisTick: {
        distance: -30,
        length: 8,
        lineStyle: {
```

```
        color: '#fff',
        width: 2
      }
    },
    splitLine: {
      distance: -30,
      length: 30,
      lineStyle: {
        color: '#fff',
        width: 4
      }
    },
    axisLabel: {
      distance: 40,
      fontSize: 10
    },
    detail: {
      valueAnimation: true,
      formatter: '{value} °C',
      color: 'auto',
      fontSize: 14
    },
    data: [
      {
        value: 78
      }
    ]
  }
 ]
};
setInterval(function () {
```

```
      myChart.setOption({
        series: [
          {
            data: [
              {
                value: Math.floor(Math.random() * 10) + 75
              }
            ]
          }
        ]
      });
    }, 2000);

    if (option && typeof option === 'object') {
      myChart.setOption(option);
    }

    window.addEventListener('resize', myChart.resize);
  }

  temperatureChart();

  function gasCensor() {
    n2oGasCensorChart();
    cmoGasCensorChart();
    co2GasCensorChart();
  }

  gasCensor();

</script>
```

</body>

</html>