













```
<template >
```

```
| <canvas @mousemove="updateMousePosition" ref="canvas"></canvas>
```

```
</template>
```

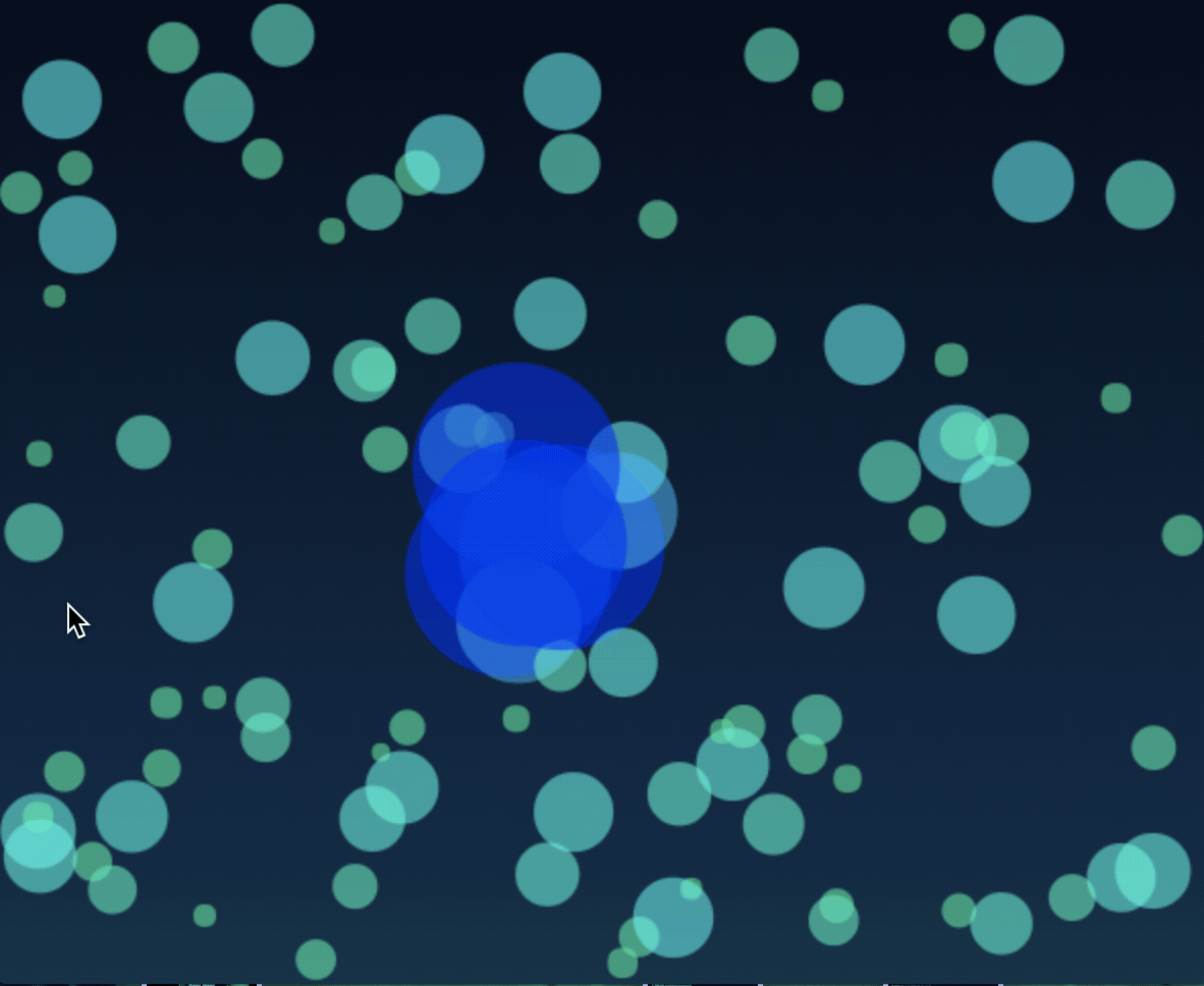


```
updateMousePosition (e) {  
  this.$set(this.mouse, 'x', e.x)  
  this.$set(this.mouse, 'y', e.y)  
}
```



```
data () {  
  return {  
    circles: [],  
    mouse: {  
      x: null,  
      y: null  
    }  
  }  
},
```







```
mounted () {  
  this.$refs.canvas.width = window.innerWidth  
  this.$refs.canvas.height = window.innerHeight  
  
  for (let i = 0; i < this.numberOfCircles; i++) {  
    let radius = this.randomNumber(5, 20)  
    let diameter = radius * 2;  
  
    let circleObj = {  
      radius: radius,  
      x: this.randomNumber(diameter, window.innerWidth - diameter),  
      y: this.randomNumber(diameter, window.innerHeight - diameter),  
      dx: this.randomNumber(-1, 1),  
      dy: this.randomNumber(-1, 1),  
      color: this.mainColor  
    }  
  
    this.circles.push(  
      new CreateCircle(circleObj, this.canvasContext, this.mouse)  
    )  
  }  
  
  this.animate()  
}
```