# Vue项目性能优化方案

1、v-for 正确设置key值  
2、封装复用的模块（http请求）、组件（ui库）  
3、路由懒加载：component：() => import('./xxx.vue')  
4、productionSourceMap: false  
5、启用gzip压缩，打包体积更小  
6、keep-alive 缓存不活跃组件  
7、插件CDN方式引入，减小项目体积  
8、图片使用CDN地址，图片懒加载

# 放大镜模板

<template>

  <div class="spec-preview">

    <img src="../assets/xiaoxin4.png" />

    <div class="event" @mousemove="moveMask"></div>

    <div class="big">

      <img src="../assets/xiaoxin4.png" ref="bigImg" />

    </div>

    <div class="mask" ref="maskRef"></div>

  </div>

</template>

<script>

export default {

  name: 'Magnifier',

  methods: {

    moveMask(e) {

      // 获取对象

      const bigImg = this.$refs.bigImg

      const mask = this.$refs.maskRef

      // 获取鼠标位置

      let x = e.offsetX - mask.offsetWidth / 2

      let y = e.offsetY - mask.offsetHeight / 2

      // 设置约束区域

      if (x < 0) x = 0

      if (x > mask.offsetWidth) x = mask.offsetWidth

      if (y < 0) y = 0

      if (y > mask.offsetHeight) y = mask.offsetHeight

      // 遮罩层跟随移动

      mask.style.left = x + 'px'

      mask.style.top = y + 'px'

      // 大图片移动

      bigImg.style.left = -2 \* x + 'px'

      bigImg.style.top = -2 \* y + 'px'

    },

  },

}

</script>

<style lang='less' scoped>

.spec-preview {

  position: relative;

  width: 400px;

  height: 400px;

  border: 1px solid #ccc;

  img {

    width: 100%;

    height: 100%;

  }

  .event {

    position: absolute;

    top: 0;

    left: 0;

    z-index: 998;

    width: 100%;

    height: 100%;

  }

  .mask {

    position: absolute;

    left: 0;

    top: 0;

    display: none;

    width: 50%;

    height: 50%;

    background-color: rgba(0, 255, 0, 0.3);

  }

  .big {

    position: absolute;

    top: -1px;

    left: 100%;

    width: 100%;

    height: 100%;

    border: 1px solid #aaa;

    z-index: 998;

    display: none;

    background: white;

    overflow: hidden;

    img {

      width: 200%;

      max-width: 200%;

      height: 200%;

      position: absolute;

      left: 0;

      top: 0;

    }

  }

  .event:hover ~ .mask,

  .event:hover ~ .big {

    display: block;

  }

}

</style>

# 分页

<template>

  <div class="box">

    <select v-model="size">

      <option value="5">5条数据</option>

      <option value="10">10条数据</option>

      <option value="15">15条数据</option>

    </select>

    {{startAndEndNum.start}}--{{startAndEndNum.end}}

    <button @click="$emit('changePageNo',pageNo-1)" :disabled="pageNo===1">上一页</button>

    <button v-if="startAndEndNum.start > 1" @click="$emit('changePageNo',1)">1</button>

    <button v-if="startAndEndNum.start > 2">...</button>

    <button v-for="item,i in startAndEndNum.end" :class="{active:pageNo===item}" @click="$emit('changePageNo',item)" v-if="item>=startAndEndNum.start"

      :key="i">{{item}}</button>

    <button v-if="startAndEndNum.end < maxPageNum - 1">...</button>

    <button v-if="startAndEndNum.end < maxPageNum" @click="$emit('changePageNo',maxPageNum)">{{maxPageNum}}</button>

    <button @click="$emit('changePageNo',pageNo+1)" :disabled="pageNo===maxPageNum">下一页</button>

    <input type="text" @keyup.enter="gotoPage" class="goto">

  </div>

</template>

<script>

export default {

  name: '',

  props: ['pageNo', 'pageSize', 'total', 'continues'],

  data() {

    return {

      size: this.pageSize,

    }

  },

  methods: {

    gotoPage(e) {

      let page = e.target.value

      if (isNaN(page) || page < 1 || page > this.maxPageNum) return (e.target.value = '')

      this.$emit('changePageNo', parseInt(page))

      e.target.value = ''

    },

  },

  computed: {

    // 最大页数

    maxPageNum() {

      return Math.ceil(this.total / this.size)

    },

    // 获取连续分页按钮的起始值和结束值

    startAndEndNum() {

      let start = 0,

        end = 0

      // 连续分页数大于页面总数

      if (this.continues > this.maxPageNum) {

        start = 1

        end = this.continues

      }

      // 连续分页数小于页面总数

      if (this.continues < this.maxPageNum) {

        start = this.pageNo - parseInt(this.continues / 2)

        end = this.pageNo + parseInt(this.continues / 2)

        // start 小于1

        if (start < 1) {

          start = 1

          end = this.continues

        }

        // end 大于最大页数

        if (end > this.maxPageNum) {

          end = this.maxPageNum

          start = this.maxPageNum - this.continues + 1

        }

      }

      return {

        start,

        end,

      }

    },

  },

}

</script>

<style lang='less' scoped>

.active {

  background-color: skyblue !important;

}

.box {

  margin: 40px auto;

  text-align: center;

  button {

    margin-left: 10px;

    padding: 0 10px;

    min-width: 30px;

    height: 30px;

    text-align: center;

    line-height: 30px;

    border: none;

    background-color: #ddd;

    cursor: pointer;

    user-select: none;

  }

  .goto {

    margin-left: 10px;

    width: 30px;

    height: 30px;

    border: 1px solid #ddd;

    outline: none;

  }

}

</style>