```html
<div class="search">
    <div class="container">
        <div class="header1"></div>
        <!--导航栏-->
      <!--复用组件-->
      <NavigationMenu
    :activeIndex="activeIndex"
    :backgroundColor="backgroundColor"
    :textColor="textColor"
    :activeTextColor="activeTextColor"
    />
    </div>
    <div style="height: 650px; display: flex" class="cab">
  <el-scrollbar style="height:100%">

    <!--页面主体-->
    <div>
        <!--导航栏与页面-->
        <div class="header1"></div>

        <!-- 上半部分：大矩形卡片 -->
        <el-card class="top-card" :style="{ height: '23rem'}">

          <!-- 所在地选择 -->
          <el-row>
              <div class="brand-container4">
          <el-cascader
          class="location-input"
            v-model="selectedCities"
            :options="options"
            size="mini"
            @change="handleCityChange"
            placeholder="请选择省份和城市"
            clearable
        ></el-cascader>


          <div class="search-container">
                <!-- 搜索框 -->
                <el-input v-model="searchKeyword" size="mini" placeholder="" class="custom-input2
                <el-button type="primary" icon="el-icon-search" class="button1" @click="search">搜
            </div>
                </div>
        </el-row>

        <!-- 品牌选择 -->
        <el-row>
          <div class="brand-container0">
            <label class="bold-text">品牌</label>
            <ul class="brand-list">
```

```html
        <li class="item" :class="{ 'hasChecked': selectedBrand === '' }" @click="selectBrand(
        <li class="item" :class="{ 'hasChecked': selectedBrand === '大众' }" @click="selectBra
        <li class="item" :class="{ 'hasChecked': selectedBrand === '本田' }" @click="selectBra
        <li class="item" :class="{ 'hasChecked': selectedBrand === '奔驰' }" @click="selectBra
        <li class="item" :class="{ 'hasChecked': selectedBrand === '宝马' }" @click="selectBra
        <li class="item" :class="{ 'hasChecked': selectedBrand === '丰田' }" @click="selectBra
        <li class="item" :class="{ 'hasChecked': selectedBrand === '奥迪' }" @click="selectBra
        <li class="item" :class="{ 'hasChecked': selectedBrand === '别克' }" @click="selectBra
        <li class="item" :class="{ 'hasChecked': selectedBrand === '福特' }" @click="selectBra
        <li class="item" :class="{ 'hasChecked': selectedBrand === '吉利' }" @click="selectBra
        <li class="item" :class="{ 'hasChecked': selectedBrand === '哈弗' }" @click="selectBra
        <li class="item" :class="{ 'hasChecked': selectedBrand === '日系' }" @click="selectBra
        <li class="item" :class="{ 'hasChecked': selectedBrand === '比亚迪' }" @click="selectB
        <li class="item" :class="{ 'hasChecked': selectedBrand === '长安' }" @click="selectBra
        <li class="item" :class="{ 'hasChecked': selectedBrand === '马自达' }" @click="selectB
        <li class="item" :class="{ 'hasChecked': selectedBrand === '红星汽车' }" @click="selec
        <li class="item" :class="{ 'hasChecked': selectedBrand === '红旗' }" @click="selectBra
        <li class="item" :class="{ 'hasChecked': selectedBrand === 'Jeep' }" @click="selectBr
        <li class="item" :class="{ 'hasChecked': selectedBrand === '捷豹' }" @click="selectBra
        <li class="item" :class="{ 'hasChecked': selectedBrand === 'MINI' }" @click="selectBr

        <!-- 其他品牌选项... -->
      </ul>
      <el-dropdown trigger="click">
        <li class="item brand-more" slot="reference" @click="showMoreBrands = !showMoreBrands">
          <span class="more-txt"></span> 更多
          <i class="el-icon-arrow-down el-icon--right"></i>
        </li>
        <el-dropdown-menu slot="dropdown">
          <!-- 更多下拉菜单选项... -->
        </el-dropdown-menu>
      </el-dropdown>
    </div>
  </el-row>

  <!-- 车系选择 -->
  <el-row>
    <div class="brand-container1">
      <label class="bold-text">车系</label>
      <ul class="brand-list">
        <li class="item" :class="{ 'hasChecked': selectedModel === '' }" @click="selectModel(
        <li class="item" v-for="model in modelList.slice(0, 15)" :key="model" :class="{ 'hasC
        <el-dropdown v-if="showMoreModel">
          <li class="item hasChecked" slot="reference" @click="showMoreModel = !showMoreModel
            <i class="el-icon-arrow-down el-icon--right"></i>
          </li>
          <el-dropdown-menu slot="dropdown">
            <el-dropdown-item v-for="model in modelList.slice(15)" :key="model" @click="selec
          </el-dropdown-menu>
        </el-dropdown>
      </ul>
```

```html
        </div>
      </el-row>
        <!-- 价格选择 -->
        <el-row>
          <div class="brand-container2">
            <label class="bold-text">价格</label>
            <ul class="brand-list">
              <li class="item" :class="{ hasChecked: selectedPrice === '' }">
                <span @click="selectPrice('')">不限</span>
              </li>
              <li class="item" :class="{ hasChecked: selectedPrice === '3万以下' }">
                <span @click="selectPrice('3万以下')">3万以下</span>
              </li>
              <li class="item" :class="{ hasChecked: selectedPrice === '3-5万' }">
                <span @click="selectPrice('3-5万')">3-5万</span>
              </li>
              <li class="item" :class="{ hasChecked: selectedPrice === '5-8万' }">
                <span @click="selectPrice('5-8万')">5-8万</span>
              </li>
              <li class="item" :class="{ hasChecked: selectedPrice === '8-10万' }">
                <span @click="selectPrice('8-10万')">8-10万</span>
              </li>
              <li class="item" :class="{ hasChecked: selectedPrice === '10-15万' }">
                <span @click="selectPrice('10-15万')">10-15万</span>
              </li>
              <li class="item" :class="{ hasChecked: selectedPrice === '15-20万' }">
                <span @click="selectPrice('15-20万')">15-20万</span>
              </li>
              <li class="item" :class="{ hasChecked: selectedPrice === '20-30万' }">
                <span @click="selectPrice('20-30万')">20-30万</span>
              </li>
              <li class="item" :class="{ hasChecked: selectedPrice === '30-50万' }">
                <span @click="selectPrice('30-50万')">30-50万</span>
              </li>
              <li class="item" :class="{ hasChecked: selectedPrice === '50-100万' }">
                <span @click="selectPrice('50-100万')">50-100万</span>
              </li>
              <li class="item" :class="{ hasChecked: selectedPrice === '100万以上' }">
                <span @click="selectPrice('100万以上')">100万以上</span>
              </li>
            </ul>
            <div class="input-row">
            <el-form :inline="true" ref="search_data" :model="search_data">
              <el-form-item label="">
                <el-input v-model.number="search_data.minPrice" size="mini" placeholder="最小" cla
                <span> - </span>
                <el-input v-model.number="search_data.maxPrice" size="mini" placeholder="最大" cla
              </el-form-item>
              <el-form-item>
                <el-button type="primary" icon="pricesearch" @click="onScreeoutMoney()" class="bu
              </el-form-item>
```

```html
        </el-form>
    </div>
    </div>
</el-row>



<!-- 其他选择 -->
<el-row>
    <div class="brand-container2">
        <lable class="bold-text">其他</lable>
        <!-- 转手次数下拉菜单 -->
        <el-dropdown trigger="click" @command="handlenum_of_transferSelect">
          <span class="brand-more">
            {{ num_of_transferText ? num_of_transferText : '不限' }}<i class="el-icon-arro
          </span>
          <el-dropdown-menu slot="dropdown">
            <el-dropdown-item command="">不限</el-dropdown-item>
            <el-dropdown-item command="0">0</el-dropdown-item>
            <el-dropdown-item command="1">1</el-dropdown-item>
            <el-dropdown-item command="2">2</el-dropdown-item>
            <el-dropdown-item command="3">3</el-dropdown-item>
            <el-dropdown-item command="4">4</el-dropdown-item>
            <el-dropdown-item command="5次及以上">5次及以上</el-dropdown-item>
          </el-dropdown-menu>
        </el-dropdown>

        <!-- 车龄下拉菜单 -->
        <el-dropdown trigger="click" @command="handleAgeSelect">
          <span class="brand-more">
            {{ ageText ? ageText : '不限' }}<i class="el-icon-arrow-down el-icon--right">←
          </span>
          <el-dropdown-menu slot="dropdown">
            <el-dropdown-item command="">不限</el-dropdown-item>
            <el-dropdown-item command="1">一年</el-dropdown-item>
            <el-dropdown-item command="2">两年</el-dropdown-item>
            <el-dropdown-item command="3">三年</el-dropdown-item>
            <el-dropdown-item command="4">四年</el-dropdown-item>
            <el-dropdown-item command="5">五年及以上</el-dropdown-item>
          </el-dropdown-menu>
        </el-dropdown>

        <!-- 变速箱下拉菜单 -->
        <el-dropdown trigger="click" @command="handleTransmissionSelect">
          <span class="brand-more">
            {{ transmissionText }}<i class="el-icon-arrow-down el-icon--right"></i>
          </span>
          <el-dropdown-menu slot="dropdown">
            <el-dropdown-item command="自动">自动</el-dropdown-item>
            <el-dropdown-item command="手动">手动</el-dropdown-item>
          </el-dropdown-menu>
        </el-dropdown>
```

```html
    </el-dropdown>

    <!-- 里程下拉菜单 -->
    <el-dropdown trigger="click" @command="handleMileageSelect">
      <span class="brand-more">
        {{ mileageText }}<i class="el-icon-arrow-down el-icon--right"></i>
      </span>
      <el-dropdown-menu slot="dropdown">
        <el-dropdown-item command="1万公里以下">1万公里以下</el-dropdown-item>
        <el-dropdown-item command="1-3万公里">1-3万公里</el-dropdown-item>
        <el-dropdown-item command="3-5万公里">3-5万公里</el-dropdown-item>
        <el-dropdown-item command="5-8万公里">5-8万公里</el-dropdown-item>
        <el-dropdown-item command="8-10万公里">8-10万公里</el-dropdown-item>
        <el-dropdown-item command="10万公里以上">10万公里以上</el-dropdown-item>
      </el-dropdown-menu>
    </el-dropdown>

    <!-- 排量下拉菜单 -->
    <el-dropdown trigger="click" @command="handleEmissionsSelect">
      <span class="brand-more">
        {{ emissionsText }}<i class="el-icon-arrow-down el-icon--right"></i>
      </span>
      <el-dropdown-menu slot="dropdown">
        <el-dropdown-item command="1.0L及以下">1.0L及以下</el-dropdown-item>
        <el-dropdown-item command="1.1L-1.6L">1.1L-1.6L</el-dropdown-item>
        <el-dropdown-item command="1.7L-2.0L">1.7L-2.0L</el-dropdown-item>
        <el-dropdown-item command="2.1L-2.5L">2.1L-2.5L</el-dropdown-item>
        <el-dropdown-item command="2.6L-3.0L">2.6L-3.0L</el-dropdown-item>
        <el-dropdown-item command="3.1L-4.0L">3.1L-4.0L</el-dropdown-item>
        <el-dropdown-item command="4.0L以上">4.0L以上</el-dropdown-item>
      </el-dropdown-menu>
    </el-dropdown>

    <!-- 排放标准下拉菜单 -->
    <el-dropdown trigger="click" @command="handleEmissionsstandardsSelect">
      <span class="brand-more">
        {{ emissionsstandardsText }}<i class="el-icon-arrow-down el-icon--right"></i>
      </span>
      <el-dropdown-menu slot="dropdown">
        <el-dropdown-item command="国V(国VI)">国V/国VI</el-dropdown-item>
        <el-dropdown-item command="欧III">欧III</el-dropdown-item>
        <el-dropdown-item command="欧V+OBD">欧V+OBD</el-dropdown-item>
        <el-dropdown-item command="国IV/国V">国IV/国V</el-dropdown-item>
        <el-dropdown-item command="国IV+OBD">国IV+OBD</el-dropdown-item>
        <el-dropdown-item command="国III+OBD">国III+OBD</el-dropdown-item>
        <el-dropdown-item command="国III">国III</el-dropdown-item>
        <el-dropdown-item command="国II">国II</el-dropdown-item>
        <el-dropdown-item command="欧II">欧II</el-dropdown-item>
        <el-dropdown-item command="国IV">国IV</el-dropdown-item>
        <el-dropdown-item command="国IV(国V)+OBD">国IV/国V+OBD</el-dropdown-item>
        <el-dropdown-item command="欧IV">欧IV</el-dropdown-item>
```

```html
      <el-dropdown-item command="欧IV+OBD">欧IV+OBD</el-dropdown-item>
      <el-dropdown-item command="欧VI">欧VI</el-dropdown-item>
      <el-dropdown-item command="国IV(国V)">国IV/国V</el-dropdown-item>
      <el-dropdown-item command="欧V">欧V</el-dropdown-item>
      <el-dropdown-item command="国V+OBD">国V+OBD</el-dropdown-item>
      <el-dropdown-item command="国VI">国VI</el-dropdown-item>
      <el-dropdown-item command="国V">国V</el-dropdown-item>
    </el-dropdown-menu>
</el-dropdown>

<!-- 颜色下拉菜单 -->
<el-dropdown trigger="click" @command="handleColorSelect">
  <span class="brand-more">
    {{ colorText }}<i class="el-icon-arrow-down el-icon--right"></i>
  </span>
  <el-dropdown-menu slot="dropdown">
    <el-dropdown-item command="黑色">黑色</el-dropdown-item>
    <el-dropdown-item command="白色">白色</el-dropdown-item>
    <el-dropdown-item command="银/灰色">银/灰色</el-dropdown-item>
    <el-dropdown-item command="红/紫色">红/紫色</el-dropdown-item>
    <el-dropdown-item command="黄/橙色">黄/橙色</el-dropdown-item>
    <el-dropdown-item command="蓝色">蓝色</el-dropdown-item>
    <el-dropdown-item command="绿色">绿色</el-dropdown-item>
    <el-dropdown-item command="粉色">粉色</el-dropdown-item>
    <el-dropdown-item command="香槟/棕色">香槟/棕色</el-dropdown-item>
    <el-dropdown-item command="其他">其他</el-dropdown-item>
  </el-dropdown-menu>
</el-dropdown>

<!-- 驱动下拉菜单 -->
<el-dropdown trigger="click" @command="handleDriveSelect">
  <span class="brand-more">
    {{ driveText }}<i class="el-icon-arrow-down el-icon--right"></i>
  </span>
  <el-dropdown-menu slot="dropdown">
    <el-dropdown-item command="前驱">前驱</el-dropdown-item>
    <el-dropdown-item command="后驱">后驱</el-dropdown-item>
    <el-dropdown-item command="四驱">四驱</el-dropdown-item>
  </el-dropdown-menu>
</el-dropdown>

<!-- 燃料类型下拉菜单 -->
<el-dropdown trigger="click" @command="handleFuelSelect">
  <span class="brand-more">
    {{ fuelText }}<i class="el-icon-arrow-down el-icon--right"></i>
  </span>
  <el-dropdown-menu slot="dropdown">
    <el-dropdown-item command="插电混动">插电混动</el-dropdown-item>
```

```html
                <el-dropdown-item command="增程式">增程式</el-dropdown-item>
                <el-dropdown-item command="汽油">汽油</el-dropdown-item>
                <el-dropdown-item command="柴油">柴油</el-dropdown-item>
                <el-dropdown-item command="电动">电动</el-dropdown-item>
                <el-dropdown-item command="油电混合">油电混合</el-dropdown-item>
                <el-dropdown-item command="油改气">油改气</el-dropdown-item>
              </el-dropdown-menu>
            </el-dropdown>

      </div>
  </el-row>


  <!-- 筛选排序 -->
  <el-row>
    <div class="brand-container">
      <ul class="brand-list">
        <!-- <li class="item" :class="{ hasChecked: selectedBrand === '综合排序' }"><span @c
        <li class="item" :class="{ hasChecked: selectedfun === '最新上架' }"><span @click="se
        <li class="item" :class="{ hasChecked: selectedfun === '价格最高' }"><span @click="se
        <li class="item" :class="{ hasChecked: selectedfun === '价格最低' }"><span @click="se
        <li class="item" :class="{ hasChecked: selectedfun === '车龄最短' }"><span @click="se
        <li class="item" :class="{ hasChecked: selectedfun === '里程最少' }"><span @click="se
        <button class="bold-text">清空筛选</button>
      </ul>

    </div>
  </el-row>


  <!-- 已选择 -->
  <el-row>
      <div class="brand-container">
        <lable class="bold-text-yixuanze">当前选择：</lable>
        <ul class="brand-list3">
          <span v-if="selectedBrand !== ''" class="selected-option">
            品牌：{{ selectedBrand }}
            <i class="el-icon-close" @click="clearSelectedBrand"></i>
          </span>
          <span v-if="selectedModel !== ''" class="selected-option">
            车系：{{ selectedModel }}
            <i class="el-icon-close" @click="clearSelectedModel"></i>
          </span>
          <span v-if="selectedPrice !== ''" class="selected-option">
            价格：{{ selectedPrice }}
            <i class="el-icon-close" @click="clearSelectedPrice"></i>
          </span>
          <span v-if="selectedNum_of_transfer!== ''" class="selected-option">
            转手次数：{{ selectedNum_of_transfer }}
            <i class="el-icon-close" @click="clearSelectedNum_of_transfer"></i>
          </span>
```

```html
          <span v-if="selectedAge!== ''" class="selected-option">
            车龄: {{ selectedAge }}
            <i class="el-icon-close" @click="clearSelectedAge"></i>
          </span>
          <span v-if="selectedTransmission!== ''" class="selected-option">
            变速箱: {{ selectedTransmission }}
            <i class="el-icon-close" @click="clearSelectedTransmission"></i>
          </span>
          <span v-if="selectedMileage!== ''" class="selected-option">
            里程: {{ selectedMileage }}
            <i class="el-icon-close" @click="clearSelectedMileage"></i>
          </span>
          <span v-if="selectedEmission!== ''" class="selected-option">
            排量: {{ selectedEmission }}
            <i class="el-icon-close" @click="clearSelectedEmission"></i>
          </span>
          <span v-if="selectedEmissionsstandards!== ''" class="selected-option">
            排量: {{ selectedEmissionsstandards }}
            <i class="el-icon-close" @click="clearselectedEmissionsstandards"></i>
          </span>
          <span v-if="selectedColor!== ''" class="selected-option">
            颜色: {{ selectedColor }}
            <i class="el-icon-close" @click="clearselectedColor"></i>
          </span>
          <span v-if="selectedDrive!== ''" class="selected-option">
            驱动: {{ selectedDrive }}
            <i class="el-icon-close" @click="clearselectedDrive"></i>
          </span>
          <span v-if="selectedFuel!== ''" class="selected-option">
            燃料类型: {{ selectedFuel }}
            <i class="el-icon-close" @click="clearselectedFuel"></i>
          </span>
        </ul>
        </div>
    </el-row>


<div> </div>

</el-card>

<div class="header2"></div>

<!-- 下半部分 -->
<el-row>
      <div class="card-container">
          <el-card v-for="item in tableData" :key="item.id" class="card-item">
            <img class="card-img" :src="item.img_url[0]" alt="image" >
            <div class="card-content">
                <h4 class="title">{{ item.title }}</h4>
                <div class="car-info">
```

```html
                        <div class="extra-info">
                            <p>{{ item.license_time }}</p>  <!-- 上牌时间 -->
                            <span class="separator2"></span>
                            <p>{{ item.mileage }}</p>   <!-- 里程 -->
                            <span class="separator2"></span>
                            <p>{{ item.location }}</p>   <!-- 所在地 -->
                        </div>
                        </div>
                        <div class="price-container">
                          <p class="price">￥{{ item.price }}万</p></div>
                          <div class="button-container">
                            <el-button type="text" class="buttondetails" @click="onEditProfile(it
                            查看详情
                            </el-button>
                          </div>
                      </div>
                    </el-card>
                </div>
            <el-row>
            <el-col :span="24">
                <div class="pagination">
                    <el-pagination
                    v-if="paginations.total > 0"
                    :page-sizes="paginations.page_sizes"
                    :page-size="paginations.page_size"
                    :layout="paginations.layout"
                    :total="paginations.total"
                    :current-page.sync="paginations.page_index"
                    @current-change="handleCurrentChange"
                    @size-change="handleSizeChange"
                    ></el-pagination>
                </div>
                </el-col>
            </el-row>
        </el-row>

    </div>
    </el-scrollbar>
    </div>
  </div>
</template>



<script>
import NavigationMenu from '@/components/NavigationMenu.vue';
export default {
  name: 'search3',
  components : {
    NavigationMenu
  },
```

```javascript
    data() {
    return {
        tableData: [], // 车辆信息数组
        currentPage: 1, // 当前页码
        totalPages: 0, // 总页数


        defaultModelList: [
            '奥迪A6L',
            '宝马5系',
            '宝马3系',
            '奔驰C级',
            '奔驰E级',
            '高尔夫',
            '迈腾',
            '速腾',
            '朗逸',
            '雅阁',
            '轩逸',
            '哈弗H6',
            '福克斯'
        ],
        allTableData: [],
        num_of_transferText: '',
        searchKeyword: '',
        sortDirection: '', // 记录当前排序方式: 'asc' (升序) 或 'desc' (降序)
        selectedfun:'',
        selectedBrand: '', // 记录用户选择的品牌
        selectedModel: '', // 记录用户选择的车系
        modelList: [], // 当前选中品牌对应的车系列表
        showMoreBrands: false,
        showMoreModel: false,
        filterTableData:[],
        selectedPrice: "",
        lowerPrice: "",
        higherPrice: "",
        isValid: true,
        areatext:'全国',
        num_of_transferText: '转手次数',
        ageText: '车龄',
        transmissionText: '变速箱',
        mileageText:'里程',
        emissionsText:'排量',
        emissionsstandardsText:'排放标准',
        colorText:'颜色',
        countryText:'国别',
        driveText:'驱动',
        seatText:'座位数',
        fuelText:'燃料类型',
        selectedNum_of_transfer: '', // 用户选择的转手次数
```

```javascript
selectedAge: '', // 用户选择的车龄
selectedTransmission: '', // 用户选择的变速箱
selectedMileage:'',
selectedEmission:'',
selectedEmissionsstandards:'',
selectedColor:'',
selectedDrive:'',
selectedFuel:'',
form: {
  img_url:"",
  title: "",
  license_time: "",
  mileage: "",
  price: "",
  location: "",
  id: "",
},
paginations: {
  page_index: 1,
  total: 0,
  page_size: 20,
  page_sizes: [20,28, 36, 48,80],
  layout: "total, sizes, prev, pager, next, jumper",
},
search_data: {
  minPrice: null,
  maxPrice: null,
},
selectedCities: [], // 用于存储选中的省份和城市的值
filterOptions: {location:''},
options: [
  {
    value: 'anhui',
    label: '安徽',
    children: [
      { value: 'hefei', label: '合肥' },
      { value: 'wuhu', label: '芜湖' },
      { value: 'bangbu', label: '蚌埠' },
      { value: 'huainan', label: '淮南' },
      { value: 'maanshan', label: '马鞍山' },
      { value: 'huaibei', label: '淮北' },
      { value: 'tongling', label: '铜陵' },
      { value: 'anqing', label: '安庆' },
      { value: 'huangshan', label: '黄山' },
      { value: 'chuzhou', label: '滁州' },
      { value: 'fuyang', label: '阜阳' },
      { value: 'suzhou', label: '宿州' },
      { value: 'liuan', label: '六安' },
      { value: 'bozhou', label: '亳州' },
      { value: 'chizhou', label: '池州' },
      { value: 'xuancheng', label: '宣城' }
```

```
    ]
  },
  {
    value: 'beijing',
    label: '北京',
    children: [
      { value: 'beijing', label: '北京' }
    ]
  },
  {
    value: 'chongqing',
    label: '重庆',
    children: [
      { value: 'chongqing', label: '重庆' }
    ]
  },
  {
    value: 'fujian',
    label: '福建',
    children: [
      { value: 'fuzhou', label: '福州' },
      { value: 'xiamen', label: '厦门' },
      { value: 'putian', label: '莆田' },
      { value: 'sanming', label: '三明' },
      { value: 'quanzhou', label: '泉州' },
      { value: 'zhangzhou', label: '漳州' },
      { value: 'nanping', label: '南平' },
      { value: 'longyan', label: '龙岩' },
      { value: 'ningde', label: '宁德' }
    ]
  },
  {
    value: 'guangdong',
    label: '广东',
    children: [
      { value: 'guangzhou', label: '广州' },
      { value: 'shaoguan', label: '韶关' },
      { value: 'shenzhen', label: '深圳' },
      { value: 'zhuhai', label: '珠海' },
      { value: 'shantou', label: '汕头' },
      { value: 'foshan', label: '佛山' },
      { value: 'jiangmen', label: '江门' },
      { value: 'zhanjiang', label: '湛江' },
      { value: 'maoming', label: '茂名' },
      { value: 'zhaoqing', label: '肇庆' },
      { value: 'huizhou', label: '惠州' },
      { value: 'meizhou', label: '梅州' },
      { value: 'shanwei', label: '汕尾' },
      { value: 'heyuan', label: '河源' },
      { value: 'yangjiang', label: '阳江' },
      { value: 'qingyuan', label: '清远' },
```

```
        { value: 'dongguan', label: '东莞' },
        { value: 'zhongshan', label: '中山' },
        { value: 'chaozhou', label: '潮州' },
        { value: 'jieyang', label: '揭阳' },
        { value: 'yunfu', label: '云浮' }
      ]
    },
    {
      value: 'guangxi',
      label: '广西',
      children: [
        { value: 'nanning', label: '南宁' },
        { value: 'liuzhou', label: '柳州' },
        { value: 'guilin', label: '桂林' },
        { value: 'wuzhou', label: '梧州' },
        { value: 'beihai', label: '北海' },
        { value: 'fangchenggang', label: '防城港' },
        { value: 'qinzhou', label: '钦州' },
        { value: 'guigang', label: '贵港' },
        { value: 'yu_lin', label: '玉林' },
        { value: 'baise', label: '百色' },
        { value: 'hezhou', label: '贺州' },
        { value: 'hechi', label: '河池' },
        { value: 'laibin', label: '来宾' },
        { value: 'chongzuo', label: '崇左' }
      ]
    },
    {
      value: 'guizhou',
      label: '贵州',
      children: [
        { value: 'guiyang', label: '贵阳' },
        { value: 'liupanshui', label: '六盘水' },
        { value: 'zunyi', label: '遵义' },
        { value: 'anshun', label: '安顺' },
        { value: 'bijie', label: '毕节' },
        { value: 'tongren', label: '铜仁' },
        { value: 'qianxinan', label: '黔西南' },
        { value: 'qiandongnan', label: '黔东南' },
        { value: 'qiannan', label: '黔南' }
      ]
    },
    {
      value: 'gansu',
      label: '甘肃',
      children: [
        { value: 'lanzhou', label: '兰州' },
        { value: 'jiayuguan', label: '嘉峪关' },
        { value: 'jinchang', label: '金昌' },
        { value: 'baiyin', label: '白银' },
        { value: 'tianshui', label: '天水' },
```

```
        { value: 'wuwei', label: '武威' },
        { value: 'zhangye', label: '张掖' },
        { value: 'pingliang', label: '平凉' },
        { value: 'jiuquan', label: '酒泉' },
        { value: 'qingyang', label: '庆阳' },
        { value: 'dingxi', label: '定西' },
        { value: 'longnan', label: '陇南' },
        { value: 'linxia', label: '临夏' },
        { value: 'gannan', label: '甘南' }
      ]
  },
  {
    value: 'hainan', label: '海南',
    children: [
        { value: 'haikou', label: '海口' },
        { value: 'sanya', label: '三亚' },
        { value: 'sansha', label: '三沙' },
        { value: 'danzhou', label: '儋州' },
        { value: 'wuzhishan', label: '五指山' },
        { value: 'qionghai', label: '琼海' },
        { value: 'wenchang', label: '文昌' },
        { value: 'wanning', label: '万宁' },
        { value: 'dongfang', label: '东方' },
        { value: 'dingan', label: '定安' },
        { value: 'tunchang', label: '屯昌' },
        { value: 'chengmai', label: '澄迈' },
        { value: 'lingao', label: '临高' },
        { value: 'baisha', label: '白沙' },
        { value: 'changjiang', label: '昌江' },
        { value: 'ledong', label: '乐东' },
        { value: 'lingshui', label: '陵水' },
        { value: 'baoting', label: '保亭' },
        { value: 'qiongzhong', label: '琼中' }
      ]
  },
  {
    value: 'henan', label: '河南',
    children: [
        { value: 'zhengzhou', label: '郑州' },
        { value: 'kaifeng', label: '开封' },
        { value: 'luoyang', label: '洛阳' },
        { value: 'pingdingshan', label: '平顶山' },
        { value: 'anyang', label: '安阳' },
        { value: 'hebi', label: '鹤壁' },
        { value: 'xinxiang', label: '新乡' },
        { value: 'jiaozuo', label: '焦作' },
        { value: 'puyang', label: '濮阳' },
        { value: 'xuchang', label: '许昌' },
        { value: 'luohe', label: '漯河' },
        { value: 'sanmenxia', label: '三门峡' },
        { value: 'nanyang', label: '南阳' },
```

```
          { value: 'shangqiu', label: '商丘' },
          { value: 'xinyang', label: '信阳' },
          { value: 'zhoukou', label: '周口' },
          { value: 'zhumadian', label: '驻马店' },
          { value: 'jiyuan', label: '济源' }
        ]
      },
      {
        value: 'hubei', label: '湖北',
        children: [
          { value: 'wuhan', label: '武汉' },
          { value: 'huangshi', label: '黄石' },
          { value: 'shiyan', label: '十堰' },
          { value: 'yichang', label: '宜昌' },
          { value: 'xiangyang', label: '襄阳' },
          { value: 'ezhou', label: '鄂州' },
          { value: 'jingmen', label: '荆门' },
          { value: 'xiaogan', label: '孝感' },
          { value: 'jingzhou', label: '荆州' },
          { value: 'huanggang', label: '黄冈' },
          { value: 'xianning', label: '咸宁' },
          { value: 'suizhou', label: '随州' },
          { value: 'enshi', label: '恩施' },
          { value: 'xiantao', label: '仙桃' },
          { value: 'qianjiang', label: '潜江' },
          { value: 'tianmen', label: '天门' },
          { value: 'shennongjia', label: '神农架' }
        ]
      },
      {
        value: 'hunan', label: '湖南',
        children: [
          { value: 'changsha', label: '长沙' },
          { value: 'zhuzhou', label: '株洲' },
          { value: 'xiangtan', label: '湘潭' },
          { value: 'hengyang', label: '衡阳' },
          { value: 'shaoyang', label: '邵阳' },
          { value: 'yueyang', label: '岳阳' },
          { value: 'changde', label: '常德' },
          { value: 'zhangjiajie', label: '张家界' },
          { value: 'yiyang', label: '益阳' },
          { value: 'chenzhou', label: '郴州' },
          { value: 'yongzhou', label: '永州' },
          { value: 'huaihua', label: '怀化' },
          { value: 'loudi', label: '娄底' },
          { value: 'xiangxi', label: '湘西' }
        ]
      },
      {
        value: 'hebei', label: '河北',
        children: [
```

```
        {value: 'shijiazhuang', label: '石家庄'},
        {value: 'tangshan', label: '唐山'},
        {value: 'qinhuangdao', label: '秦皇岛'},
        {value: 'handan', label: '邯郸'},
        {value: 'xingtai', label: '邢台'},
        {value: 'baoding', label: '保定'},
        {value: 'zhangjiakou', label: '张家口'},
        {value: 'chengde', label: '承德'},
        {value: 'cangzhou', label: '沧州'},
        {value: 'langfang', label: '廊坊'},
        {value: 'hengshui', label: '衡水'}
      ]
    },
    {
      value: 'heilongjiang', label: '黑龙江',
      children: [
        {value: 'haerbin', label: '哈尔滨'},
        {value: 'qiqihaer', label: '齐齐哈尔'},
        {value: 'jixi', label: '鸡西'},
        {value: 'hegang', label: '鹤岗'},
        {value: 'shuangyashan', label: '双鸭山'},
        {value: 'daqing', label: '大庆'},
        {value: 'yichun', label: '伊春'},
        {value: 'jiamusi', label: '佳木斯'},
        {value: 'qitaihe', label: '七台河'},
        {value: 'mudanjiang', label: '牡丹江'},
        {value: 'heihe', label: '黑河'},
        {value: 'suihua', label: '绥化'},
        {value: 'daxinganling', label: '大兴安岭'}
      ]
    },
    {
      value: 'jiangsu', label: '江苏',
      children: [
        {value: 'nanjing', label: '南京'},
        {value: 'wuxi', label: '无锡'},
        {value: 'xuzhou', label: '徐州'},
        {value: 'changzhou', label: '常州'},
        {value: 'suzhou', label: '苏州'},
        {value: 'nantong', label: '南通'},
        {value: 'lianyungang', label: '连云港'},
        {value: 'huaian', label: '淮安'},
        {value: 'yancheng', label: '盐城'},
        {value: 'yangzhou', label: '扬州'},
        {value: 'zhenjiang', label: '镇江'},
        {value: 'tai_zhou', label: '泰州'},
        {value: 'suqian', label: '宿迁'}
      ]
    },
    {
      value: 'jiangxi', label: '江西',
```

```
    children: [
      {value: 'nanchang', label: '南昌'},
      {value: 'jingdezhen', label: '景德镇'},
      {value: 'ping_xiang', label: '萍乡'},
      {value: 'jiujiang', label: '九江'},
      {value: 'xinyu', label: '新余'},
      {value: 'yingtan', label: '鹰潭'},
      {value: 'ganzhou', label: '赣州'},
      {value: 'jian', label: '吉安'},
      {value: 'yi_chun', label: '宜春'},
      {value: 'fu_zhou', label: '抚州'},
      {value: 'shangrao', label: '上饶'}
    ]
  },
  {
    value: 'jilin', label: '吉林',
    children: [
      {value: 'changchun', label: '长春'},
      {value: 'jilinshi', label: '吉林'},
      {value: 'siping', label: '四平'},
      {value: 'liaoyuan', label: '辽源'},
      {value: 'tonghua', label: '通化'},
      {value: 'baishan', label: '白山'},
      {value: 'songyuan', label: '松原'},
      {value: 'baicheng', label: '白城'},
      {value: 'yanbian', label: '延边'}
    ]
  },
  {
    value: 'liaoning', label: '辽宁',
    children: [
      {value: 'shenyang', label: '沈阳'},
      {value: 'dalian', label: '大连'},
      {value: 'anshan', label: '鞍山'},
      {value: 'fushun', label: '抚顺'},
      {value: 'benxi', label: '本溪'},
      {value: 'dandong', label: '丹东'},
      {value: 'jinzhou', label: '锦州'},
      {value: 'yingkou', label: '营口'},
      {value: 'fuxin', label: '阜新'},
      {value: 'liaoyang', label: '辽阳'},
      {value: 'panjin', label: '盘锦'},
      {value: 'tieling', label: '铁岭'},
      {value: 'chaoyang', label: '朝阳'},
      {value: 'huludao', label: '葫芦岛'}
    ]
  },
  {
    value: 'neimenggu', label: '内蒙古',
    children: [
      {value: 'huhehaote', label: '呼和浩特'},
```

```
        {value: 'baotou', label: '包头'},
        {value: 'wuhai', label: '乌海'},
        {value: 'chifeng', label: '赤峰'},
        {value: 'tongliao', label: '通辽'},
        {value: 'eerduosi', label: '鄂尔多斯'},
        {value: 'hulunbeier', label: '呼伦贝尔'},
        {value: 'bayannaoer', label: '巴彦淖尔'},
        {value: 'wulanchabu', label: '乌兰察布'},
        {value: 'xinganmeng', label: '兴安盟'},
        {value: 'xilinguolemeng', label: '锡林郭勒盟'},
        {value: 'alashanmeng', label: '阿拉善盟'}
    ]
  },
  {
    value: 'ningxia', label: '宁夏',
    children: [
        {value: 'yinchuan', label: '银川'},
        {value: 'shizuishan', label: '石嘴山'},
        {value: 'wuzhong', label: '吴忠'},
        {value: 'guyuan', label: '固原'},
        {value: 'zhongwei', label: '中卫'}
    ]
  },
  {
    value: 'qinghai', label: '青海',
    children: [
        {value: 'xining', label: '西宁'},
        {value: 'haidong', label: '海东'},
        {value: 'haibei', label: '海北'},
        {value: 'huangnan', label: '黄南'},
        {value: 'hai_nan', label: '海南'},
        {value: 'guoluo', label: '果洛'},
        {value: 'yushu', label: '玉树'},
        {value: 'haixi', label: '海西'}
    ]
  },
  {
    value: 'shanxi',
    label: '陕西',
    children: [
        { value: 'xian', label: '西安' },
        { value: 'tongchuan', label: '铜川' },
        { value: 'baoji', label: '宝鸡' },
        { value: 'xianyang', label: '咸阳' },
        { value: 'weinan', label: '渭南' },
        { value: 'yanan', label: '延安' },
        { value: 'hanzhong', label: '汉中' },
        { value: 'yulin', label: '榆林' },
        { value: 'ankang', label: '安康' },
        { value: 'shangluo', label: '商洛' },
        { value: 'xixianxinqu', label: '西咸新区' }
```

```
    ]
  },
  {
    value: 'sichuan',
    label: '四川',
    children: [
      { value: 'chengdu', label: '成都' },
      { value: 'zigong', label: '自贡' },
      { value: 'panzhihua', label: '攀枝花' },
      { value: 'luzhou', label: '泸州' },
      { value: 'deyang', label: '德阳' },
      { value: 'mianyang', label: '绵阳' },
      { value: 'guangyuan', label: '广元' },
      { value: 'suining', label: '遂宁' },
      { value: 'neijiang', label: '内江' },
      { value: 'leshan', label: '乐山' },
      { value: 'nanchong', label: '南充' },
      { value: 'meishan', label: '眉山' },
      { value: 'yibin', label: '宜宾' },
      { value: 'guangan', label: '广安' },
      { value: 'dazhou', label: '达州' },
      { value: 'yaan', label: '雅安' },
      { value: 'bazhong', label: '巴中' },
      { value: 'ziyang', label: '资阳' },
      { value: 'aba', label: '阿坝' },
      { value: 'ganzi', label: '甘孜' },
      { value: 'liangshan', label: '凉山' }
    ]
  },
  {
    value: 'shanghai',
    label: '上海',
    children: [
      { value: 'shanghai', label: '上海' }
    ]
  },
  {
    value: 'shanxi',
    label: '山西',
    children: [
      { value: 'taiyuan', label: '太原' },
      { value: 'datong', label: '大同' },
      { value: 'yangquan', label: '阳泉' },
      { value: 'zhangzhi', label: '长治' },
      { value: 'jincheng', label: '晋城' },
      { value: 'shuozhou', label: '朔州' },
      { value: 'jinzhong', label: '晋中' },
      { value: 'yuncheng', label: '运城' },
      { value: 'xinzhou', label: '忻州' },
      { value: 'linfen', label: '临汾' },
      { value: 'lvliang', label: '吕梁' }
```

```
      ]
    },
    {
      value: 'shandong',
      label: '山东',
      children: [
        { value: 'jinan', label: '济南' },
        { value: 'qingdao', label: '青岛' },
        { value: 'zibo', label: '淄博' },
        { value: 'zaozhuang', label: '枣庄' },
        { value: 'dongying', label: '东营' },
        { value: 'yantai', label: '烟台' },
        { value: 'weifang', label: '潍坊' },
        { value: 'jining', label: '济宁' },
        { value: 'taian', label: '泰安' },
        { value: 'weihai', label: '威海' },
        { value: 'rizhao', label: '日照' },
        { value: 'laiwu', label: '莱芜' },
        { value: 'linyi', label: '临沂' },
        { value: 'dezhou', label: '德州' },
        { value: 'liaocheng', label: '聊城' },
        { value: 'binzhou', label: '滨州' },
        { value: 'heze', label: '菏泽' }
      ]
    },
    {
      value: 'tianjin',
      label: '天津',
      children: [
        { value: 'tianjin', label: '天津' }
      ]
    },
    {
      value: 'xinjiang',
      label: '新疆',
      children: [
        { value: 'wulumuqi', label: '乌鲁木齐' },
        { value: 'kelamayi', label: '克拉玛依' },
        { value: 'turpan', label: '吐鲁番' },
        { value: 'hami', label: '哈密' },
        { value: 'changji', label: '昌吉' },
        { value: 'boertala', label: '博尔塔拉' },
        { value: 'bayinguoleng', label: '巴音郭楞' },
        { value: 'akesu', label: '阿克苏' },
        { value: 'kezilesu', label: '克孜勒苏' },
        { value: 'kashen', label: '喀什' },
        { value: 'hetian', label: '和田' },
        { value: 'yili', label: '伊犁' },
        { value: 'tacheng', label: '塔城' },
        { value: 'aletai', label: '阿勒泰' },
        { value: 'shihezi', label: '石河子' },
```

```
      { value: 'aral', label: '阿拉尔' },
      { value: 'tumxuk', label: '图木舒克' },
      { value: 'wujiaqu', label: '五家渠' },
      { value: 'beitun', label: '北屯' },
      { value: 'tiemenguan', label: '铁门关' },
      { value: 'shuanghe', label: '双河' },
      { value: 'kokdala', label: '可克达拉' },
      { value: 'kunyu', label: '昆玉' }
    ]
  },
  {
    value: 'xizang',
    label: '西藏',
    children: [
      { value: 'lasa', label: '拉萨' },
      { value: 'rikaze', label: '日喀则' },
      { value: 'qamdo', label: '昌都' },
      { value: 'nyingchi', label: '林芝' },
      { value: 'shannan', label: '山南' },
      { value: 'naqu', label: '那曲' },
      { value: 'ali', label: '阿里' }
    ]
  },
  {
    value: 'yunnan',
    label: '云南',
    children: [
      { value: 'kunming', label: '昆明' },
      { value: 'qujing', label: '曲靖' },
      { value: 'yuxi', label: '玉溪' },
      { value: 'baoshan', label: '保山' },
      { value: 'zhaotong', label: '昭通' },
      { value: 'lijiang', label: '丽江' },
      { value: 'puer', label: '普洱' },
      { value: 'lincang', label: '临沧' },
      { value: 'chuxiong', label: '楚雄' },
      { value: 'honghe', label: '红河' },
      { value: 'wenshan', label: '文山' },
      { value: 'xishuangbanna', label: '西双版纳' },
      { value: 'dali', label: '大理' },
      { value: 'dehong', label: '德宏' },
      { value: 'nujiang', label: '怒江' },
      { value: 'diqing', label: '迪庆' }
    ]
  },
  {
    value: 'zhejiang',
    label: '浙江',
    children: [
      { value: 'hangzhou', label: '杭州' },
      { value: 'ningbo', label: '宁波' },
```

```
                { value: 'wenzhou', label: '温州' },
                { value: 'jiaxing', label: '嘉兴' },
                { value: 'huzhou', label: '湖州' },
                { value: 'shaoxing', label: '绍兴' },
                { value: 'jinhua', label: '金华' },
                { value: 'quzhou', label: '衢州' },
                { value: 'zhoushan', label: '舟山' },
                { value: 'taizhou', label: '台州' },
                { value: 'lishui', label: '丽水' },
                { value: 'zhoushanxinqu', label: '舟山群岛新区' }
            ]
        }
    ],
    };
},

mounted() {
    this.modelList = this.defaultModelList;
    // 其他初始化操作
},

    created() {
        this.getProfile();
    },
    methods: {
        //提取排量的数字部分
        extractDisplacement(displacement) {
            const value = parseFloat(displacement);
            return isNaN(value) ? 0 : value;
        },

        selectfun(fun) {
            this.selectedfun = fun;
            switch (fun) {
                case '最新上架':
                    this.sortDirection = ''; // 取消排序
                    this.sortByUpdateTime();
                    break;
                case '价格最高':
                    this.sortDirection = 'desc'; // 设置排序方式为降序
                    this.sortByPrice();
                    break;
                case '价格最低':
                    this.sortDirection = 'asc'; // 设置排序方式为升序
                    this.sortByPrice();
                    break;
                case '车龄最短':
                    this.sortDirection = '';
                    this.sortByAge();
                    break;
                case '里程最少':
```

```javascript
        this.sortDirection = '';
        this.sortByMileage();
        break;
      // 添加其他筛选条件的处理
      // ...
    }
  },
  //价格高低排序
  sortByPrice() {
    // 根据排序方式进行升序或降序排序
    this.allTableData.sort((a, b) => {
      if (this.sortDirection === 'asc') {
        return a.price - b.price; // 价格升序排列
      }
      if (this.sortDirection === 'desc') {
        return b.price - a.price; // 价格降序排列
      }
      return 0;
    });

    this.setPaginations();
  },

  //车龄排序
  sortByAge() {
    this.allTableData.sort((a, b) => {
      // 首先判断是否为未上牌车辆
      if (a.license_time === '未上牌' || b.license_time === '未上牌') {
        if (a.license_time === b.license_time) {
          return 0;
        } else if (a.license_time === '未上牌') {
          return 1; // 未上牌车辆放在最后
        } else {
          return -1; // 未上牌车辆放在最后
        }
      }

      const ageA = this.calculateAge(a.license_time);
      const ageB = this.calculateAge(b.license_time);
      return ageA - ageB;
    });
    this.setPaginations();
  },
  //里程排序
  sortByMileage() {
    this.allTableData.sort((a, b) => {
      const mileageA = this.parseMileage(a.mileage);
      const mileageB = this.parseMileage(b.mileage);
      return mileageA - mileageB;
    });
    this.setPaginations();
```

```
    },
    //计算车龄
    calculateAge(licenseTime) {
      const currentDate = new Date();
      const year = parseInt(licenseTime.substr(0, 4));
      const month = parseInt(licenseTime.substr(5, 2)) - 1; // 月份需要减1，因为JavaScript中的月份是从0开始的
      const licenseDate = new Date(year, month);

      const ageInYears = currentDate.getFullYear() - licenseDate.getFullYear();
      if (currentDate.getMonth() < licenseDate.getMonth() ||
        (currentDate.getMonth() === licenseDate.getMonth() && currentDate.getDate() < licenseDate.getDa
          return ageInYears - 1;
      }
      return ageInYears;
    },


    //计算里程
    parseMileage(mileage) {
      return parseFloat(mileage.replace(/万公里/g, ''));
    },
    //最新上架
    sortByUpdateTime() {
      this.allTableData.sort((a, b) => {
        const currentTime = new Date().getTime();
        const timeA = new Date(a.update_time).getTime();
        const timeB = new Date(b.update_time).getTime();
        const diffA = currentTime - timeA; // 当前时间减去上架时间的差距
        const diffB = currentTime - timeB;
        return diffB - diffA; // 按照差距排序，差距越小表示越新上架
      });
      this.setPaginations();
    },


    // 输入搜索功能
    search() {
const keyword = this.searchKeyword.trim(); // 去掉输入关键词两端的空格

if (keyword !== '') {
  const regex = new RegExp(keyword, 'i'); // 创建不区分大小写的正则表达式

  this.tableData = this.allTableData.filter((item) => {
    // 在每个字段中进行匹配
    return (
      regex.test(item.title) ||
      regex.test(item.brand) ||
      regex.test(item.model) ||
      regex.test(item.license_time) ||
      regex.test(item.mileage) ||
      regex.test(item.transmission) ||
      regex.test(item.emission_standards) ||
      regex.test(item.displacement) ||
```

```javascript
          regex.test(item.num_of_transfer) ||
          regex.test(item.location) ||
          regex.test(item.engine) ||
          regex.test(item.color) ||
          regex.test(item.fuel_standard) ||
          regex.test(item.drive_model) ||
          regex.test(item.fuel_type) ||
          regex.test(item.pure_elec_range) ||
          regex.test(item.fast_charge_time) ||
          regex.test(item.battery_capacity) ||
          regex.test(item.slow_charge_time) ||
          (typeof item.update_time === 'string' && regex.test(item.update_time)) ||
          (typeof item.price === 'number' && regex.test(item.price.toString()))
        );
      });
    } else {
      this.tableData = this.allTableData; // 若关键词为空，则展示全部数据
    }

    this.setPaginations(); // 更新分页数据
    },



    selectModel(model) {
      this.selectedModel = model;

      if (model === '') {
        // 不限车系，显示原始数据
        this.tableData = this.allTableData;
      } else {
        this.tableData = this.allTableData.filter(item => item.model === model);
      }

    },
      selectBrand(brand) {
        this.selectedBrand = brand;
        this.selectedModel ='';
        this.selectedfun = ''; // 重置排序方式
        this.sortDirection = ''; // 重置排序方式
        this.showMoreModel = false;
        this.showMoreBrands = false;

        // 根据选择的品牌更新车辆数据
        if (brand === '') {
          // 不限品牌，显示原始数据
          this.tableData = this.allTableData;
        } else {
          this.tableData = this.allTableData.filter(item => item.brand === brand);
        }
        // 根据选择的品牌更新车系选项
```

```javascript
    if (brand === '') {
      // 不限品牌，显示原始选项
      this.modelList = ['奥迪A6L', '宝马5系', '宝马3系', '奔驰C级', '奔驰E级', '高尔夫', '迈腾', '速腾', '朗逸'
    } else if (brand === '大众') {
      // 大众品牌对应的车系列表
      this.modelList = ['速腾','高尔夫','大众POLO','朗逸','迈腾','宝来','帕萨特','凌渡','途观L','途观','大众CC'
    } else if (brand === '本田') {
      // 本田品牌对应的车系列表
      this.modelList = ['享域','雅阁', 'CR-V', '思域', '奥德赛', '艾力绅', 'Inspire', '里程', '飞度', '锋范',
    } else if (brand === '奔驰') {
      this.modelList = ['C级', 'E级', 'S级', 'GLC', 'GLE', 'GLA', 'GLB', 'GLS', 'A级', 'B级', 'CLA级', 'C
    } else if (brand === '宝马') {
      this.modelList = ['3系', '5系', '7系', 'X1', 'X3', 'X5', 'X6', 'X7', 'i3', 'i8']
    } else if (brand === '丰田') {
      this.modelList = ['卡罗拉', '雷凌', '汉兰达', '普拉多', '凯美瑞', '柯斯达', '锐志', '兰德酷路泽', 'RAV4荣
    } else if (brand === '奥迪') {
      this.modelList = ['A3', 'A4', 'A5', 'A6', 'A7', 'A8', 'Q2', 'Q3', 'Q5', 'Q7', 'Q8', 'TT', 'R8', '
    } else if (brand === '别克') {
      this.modelList = ['昂科威', '君越', '英朗', 'GL8', 'VELITE 7', '昂科拉', '威朗', 'GL6', '探岳', 'ENCOR
    } else if (brand === '福特') {
      this.modelList = ['福克斯', '蒙迪欧', '翼虎', '锐界', '探险者', '翼搏', '全顺', '金牛座', '撼路者', 'Must
    } else if (brand === '吉利') {
      this.modelList = ['博越', '星越', '远景X3', '缤瑞', '远景X5', '帝豪', '博瑞', '远景L', '嘉际', 'ICON',
    } else if (brand === '哈弗') {
      this.modelList = ['H6', 'H9', 'F7', 'H2', 'F5', 'H4', 'H5', 'H2s', 'H1', 'M6', 'F7x']
    } else if (brand === '日系') {
      this.modelList = ['雅阁', 'RAV4', '思域', '凯美瑞', 'CR-V', '皇冠', 'RAV4荣放', '汉兰达', '逍客', '锐志
    } else if (brand === '比亚迪') {
      this.modelList = ['元EV', '唐', '宋Pro', '汉']
    } else if (brand === '长安') {
      this.modelList = ['悦翔', '逸动', '奔奔', 'CS35', 'CS75', '和悦', '悦豪', '睿骋', 'UNI-T', '逸动DT']
    } else if (brand === '马自达') {
      this.modelList = ['马自达3', '马自达6', 'CX-4', 'CX-5', 'CX-8', 'CX-9', '阿特兹', '马自达2']
    } else if (brand === '红星汽车') {
      this.modelList = ['红旗E-HS3', '红旗E-HS9']
    } else if (brand === '红旗') {
      this.modelList = ['红旗E-HS3', '红旗E-HS9']
    } else if (brand === 'Jeep') {
      this.modelList = ['大切诺基', '自由光', '指南者', '牧马人', '自由侠']
    } else if (brand === '捷豹') {
      this.modelList = ['E-PACE', 'F-PACE', 'F-TYPE']
    } else if (brand === 'MINI') {
      this.modelList = ['MINI 3门', 'MINI 5门', 'MINI CLUBMAN', 'MINI COUNTRYMAN']
    }

    // 判断品牌列表的长度，如果超过15个则触发"更多"下拉菜单
    if (this.modelList.length > 15) {
      this.showMoreModel = true;
    }
  },
```

```javascript
getProfile() {
  // 获取表格数据
  this.$axios("/api/cars").then((res) => {
    this.allTableData = res.data;
    this.tableTableData = res.data;
    this.filterTableData = res.data;
    // 设置分页数据
    this.setPaginations();
  });
},

selectPrice(price) {
  if (price === '不限') {
    this.selectedPrice = '';
    // 不限价格，显示原始数据
    this.tableData = this.allTableData;
    return;
  }
  this.selectedPrice = price;
  switch (price) {
    case '3万以下':
      this.search_data.minPrice = 0;
      this.search_data.maxPrice = 3;
      break;
    case '3-5万':
      this.search_data.minPrice = 3;
      this.search_data.maxPrice = 5;
      break;
    case '5-8万':
      this.search_data.minPrice = 5;
      this.search_data.maxPrice = 8;
      break;
    case '8-10万':
      this.search_data.minPrice = 8;
      this.search_data.maxPrice = 10;
      break;
    case '10-15万':
      this.search_data.minPrice = 10;
      this.search_data.maxPrice = 15;
      break;
    case '15-20万':
      this.search_data.minPrice = 15;
      this.search_data.maxPrice = 20;
      break;
    case '20-30万':
      this.search_data.minPrice = 20;
      this.search_data.maxPrice = 30;
      break;
    case '30-50万':
      this.search_data.minPrice = 30;
```

```javascript
        this.search_data.maxPrice = 50;
        break;
      case '50-100万':
        this.search_data.minPrice = 50;
        this.search_data.maxPrice = 100;
        break;
      case '100万以上':
        this.search_data.minPrice = 100;
        this.search_data.maxPrice = 1000000000000000000000;
        break;

    }
    this.onScreeoutMoney();
  },
  onScreeoutMoney() {
    if (this.search_data.maxPrice !== null && this.search_data.minPrice !== null && this.search_data.mi
      // 最大值小于最小值，给出错误提示或执行其他操作
      alert('请输入有效的价格范围');
      return;
    };
    const minPrice = this.search_data.minPrice;
    const maxPrice = this.search_data.maxPrice;
    this.allTableData = this.filterTableData.filter((item) => {
      if(minPrice === null && maxPrice === null){
        return true;
      }else if (minPrice === null) {
        return item.price <= maxPrice;
      }else if (maxPrice === null) {
        return item.price >= minprice;
      } else{
        return item.price >= minPrice && item.price <= maxPrice;
      }
    });
    this.setPaginations();
  },



  handleCurrentChange(page) {
    // 当前页
    let sortnum = this.paginations.page_size * (page - 1);
    let table = this.allTableData.filter((item, index) => {
      return index >= sortnum;
    });
    // 设置默认分页数据
    this.tableData = table.filter((item, index) => {
      return index < this.paginations.page_size;
    });
  },
  handleSizeChange(page_size) {
    // 切换size
```

```javascript
      this.paginations.page_index = 1;
      this.paginations.page_size = page_size;
      this.tableData = this.allTableData.filter((item, index) => {
        return index < page_size;
      });
    },
    setPaginations() {
      // 总页数
      this.paginations.total = this.allTableData.length;
      this.paginations.page_index = 1;
      this.paginations.page_size = 20;    // 将page_size的值改为20
      // 设置默认分页数据
      this.tableData = this.allTableData.filter((item, index) => {
        return index < this.paginations.page_size;
      });
    },
      //其他中下拉选项的方法
    handleCommand(command) {
    this.$message('click on item ' + command);
    },
    handleAreaSelect(command) {
      this.areatext = command;
    },
    handlenum_of_transferSelect(command) {
    this.num_of_transferText = command;

    if (command === '') {
      // 不限转手次数，显示所有数据
      this.tableData = this.allTableData;
    } else if (command === '5次及以上') {
      this.tableData = this.allTableData.filter(item => {
        const num_of_transfer = parseInt(item.num_of_transfer.split('次')[0]);
        return num_of_transfer >= 5;
      });
    } else {
      this.tableData = this.allTableData.filter(item => {
        const num_of_transfer = parseInt(item.num_of_transfer.split('次')[0]);
        return num_of_transfer === parseInt(command);
      });
    }
    this.selectedNum_of_transfer = this.num_of_transferText;
},
handleAgeSelect(command) {
this.ageText = command;

if (command === "") {
  // 不限车龄，显示所有数据
  this.tableData = this.allTableData;
} else if (command === "五年及以上") {
  this.tableData = this.allTableData.filter((item) => {
    const licenseYear = parseInt(item.license_time.substr(0, 4));
```

```javascript
      const currentYear = new Date().getFullYear();
      return currentYear - licenseYear >= 5;
    });
  } else {
    const selectedAge = parseInt(command);
    this.tableData = this.allTableData.filter((item) => {
      const licenseYear = parseInt(item.license_time.substr(0, 4));
      const currentYear = new Date().getFullYear();
      return currentYear - licenseYear === selectedAge;
    });
  }
  this.selectedAge = this.ageText;
},
  handleTransmissionSelect(command) {
    this.transmissionText = command;

    if (command === "") {
      // 不限变速箱，显示所有数据
      this.tableData = this.allTableData;
    } else {
      this.tableData = this.allTableData.filter((item) => {
        return item.transmission.toLowerCase() === command.toLowerCase();
      });
    }
      // 将用户选择的变速箱存储在selectedTransmission变量中
  this.selectedTransmission = this.transmissionText;
  },

  handleMileageSelect(command) {
    this.mileageText = command;

    if (command === "") {
      // 不限里程，显示所有数据
      this.tableData = this.allTableData;
    } else {
      this.tableData = this.allTableData.filter((item) => {
        const mileage = this.parseMileage(item.mileage);
        switch (command) {
          case "1万公里以下":
            return mileage < 1;
          case "1-3万公里":
            return mileage >= 1 && mileage <= 3;
          case "3-5万公里":
            return mileage >= 3 && mileage <= 5;
          case "5-8万公里":
            return mileage >= 5 && mileage <= 8;
          case "8-10万公里":
            return mileage >= 8 && mileage <= 10;
          case "10万公里以上":
            return mileage > 10;
          default:
```

```javascript
        return true;
      }
    });
  }
  this.selectedMileage = this.mileageText;
},
handleEmissionsSelect(command) {
  this.emissionsText = command;

  if (command === "") {
    // 不限排量，显示所有数据
    this.tableData = this.allTableData;
  } else {
    const unit = "L"; // 假设排量单位为L
    const minValue = this.extractDisplacement(command.split("-")[0]);
    const maxValue = this.extractDisplacement(command.split("-")[1]);

    this.tableData = this.allTableData.filter((item) => {
      const displacement = this.extractDisplacement(item.displacement);
      return displacement >= minValue && displacement <= maxValue;
    });
  }
  this.selectedEmission = this.emissionsText;
},
handleEmissionsstandardsSelect(command) {
  this.emissionsstandardsText = command;

  if (command === "") {
    // 不限排放标准，显示所有数据
    this.tableData = this.allTableData;
  } else {
    this.tableData = this.allTableData.filter((item) => {
      return item.emission_standards === command;
    });
  }
  this.selectedEmissionsstandards = this.emissionsstandardsText;
},
handleColorSelect(command) {
  this.colorText = command;

  if (command === "") {
    // 不限颜色，显示所有数据
    this.tableData = this.allTableData;
  } else {
    this.tableData = this.allTableData.filter((item) => {
      return item.color === command;
    });
  }
  this.selectedColor = this.colorText;
},
handleDriveSelect(command) {
```

```javascript
      this.driveText = command;

      if (command === "") {
        // 不限驱动类型，显示所有数据
        this.tableData = this.allTableData;
      } else {
        this.tableData = this.allTableData.filter((item) => {
          return item.drive_model === command;
        });
      }
      this.selectedDrive = this.driveText;
    },
    handleFuelSelect(command) {
      this.fuelText = command;

      if (command === "") {
        // 不限燃料类型，显示所有数据
        this.tableData = this.allTableData;
      } else {
        this.tableData = this.allTableData.filter((item) => {
          return item.fuel_type === command;
        });
      }
      this.selectedFuel = this.fuelText;
    },

    clearSelectedBrand() {
        this.selectedBrand = '';
        // 其他逻辑...
      },
      clearSelectedModel() {
        this.selectedModel = '';
        // 其他逻辑...
      },
      clearSelectedPrice() {
        this.selectedPrice = '';
        // 其他逻辑...
      },

      clearSelectedTransmission() {
        this.selectedTransmission ='';
      },

      clearSelectedNum_of_transfer() {
        this.selectedNum_of_transfer ='';
      },

      clearSelectedAge() {
        this.selectedAge ='';
      },
      clearSelectedMileage() {
```

```javascript
      this.selectedMileage ='';
    },
    clearSelectedEmission() {
      this.selectedEmission ='';
    },
    clearselectedEmissionsstandards() {
      this.selectedEmissionsstandards ='';
    },

    clearselectedColor() {
      this.selectedColor ='';
    },
    clearselectedDrive() {
      this.selectedDrive ='';
    },
    clearselectedFuel() {
      this.selectedFuel ='';
    },
  },
};
```

</script>

```css
<style scoped>
  .pagination {
    text-align: right;
    margin-top: 10px;
  }
  .search {
  position: fixed;
  width: 100%;
  height: 100%;
  background-size: 850px,200px;
  background-repeat: no-repeat;
  background-attachment:fixed;
  background-position:90% 65%, 25% 48%;
  background-color: rgba(245, 246, 252, 1);
  }
  .container {
  display: flex;
  justify-content: flex-end;
  }
  .navigation .el-menu-item {
  font-family: "Microsoft YaHei"; /* 设置文字字体 */
  font-size: 16px; /* 设置文字大小 */
  font-weight: bold; /* 设置文字加粗 */
  text-decoration: none; /* 取消文字下划线 */
```

```css
}
.navigation .el-menu-item a {
font-family: "Microsoft YaHei"; /* 设置文字字体 */
font-size: 16px; /* 设置文字大小 */
font-weight: bold; /* 设置文字加粗 */
text-decoration: none; /* 取消文字下划线 */
}

.header1 {
height: 3rem; /* 设置间隔高度，可根据需要调整数值 */
}
.header2 {
height: 3rem; /* 设置间隔高度，可根据需要调整数值 */
}

/*上半部分背景卡片的颜色 */
.top-card {
background-color:rgba(255, 255, 255, 1);
margin-left: 0px;
margin-right:0px;
}
.bottom-row {
margin-top: 1.5rem; /* 控制底部行与上半部分的间距 */
}

/**全国选项的颜色和样式 */
.location-input {
  width: 150px;
  font-size: 8px; /* 修改内部文字的大小为 14px */
  font-weight: bold; /* 将内部文字设为加粗 */
  margin-bottom:0px;
}

.el-icon-arrow-down {
    font-size: 12px;
}
.bold-text {
font-weight: bold;
font-size: 12px; /* 设置字体大小 */
color: #000000; /* 设置字体颜色 */
}

.bold-text-yixuanze {
font-size: 8px; /* 设置字体大小 */
color:rgb(37, 38, 43); /* 设置字体颜色 */
}

/**品牌部分的样式 */
```

```css
.brand-container {
margin-top: 20px; /* 调整第一个容器与上方元素的距离 */
display: flex;
align-items: center;
}

.brand-container4 {
display: flex;
align-items: center;
}

.brand-container0 {
margin-top: 20px; /* 调整品牌与上方元素的距离 */
display: flex;
align-items: center;
}
.brand-container0 .bold-text {
margin-right: 15px; /* 添加右边距，调整"品牌"标签与文本之间的距离 */
}

.brand-container1 {
margin-top: 23px; /* 调整车系与上方元素的距离 */
display: flex;
align-items: center;
}
.brand-container1 .bold-text {
margin-right: 15px; /* 添加右边距，调整"品牌"标签与文本之间的距离 */
}

.brand-container2 {
margin-top: -10px; /* 调整价格与上方元素的距离 */
display: flex;
align-items: center;
}
.brand-container2 .bold-text {
margin-right: 15px; /* 添加右边距，调整"品牌"标签与文本之间的距离 */
}

.brand-list {
display: flex;
align-items: center;
}
.brand-list .item {
margin-right: 15px; /* 设置词与词之间的空隙大小，调整数值以获得所需的间距 */
display: inline;
font-size: 12px; /* 设置字体大小 */
}
.brand-list3 {
display: flex;
align-items: center;
}
```

```css
.brand-list3 .item {
margin-right: 15px; /* 设置词与词之间的空隙大小，调整数值以获得所需的间距 */
display: inline;
font-size: 6px; /* 设置字体大小 */
}
.brand-container .bold-text {
margin-right: 15px; /* 添加右边距，调整"品牌"标签与文本之间的距离 */
}

.brand-container .bold-text-yixuanze {
margin-right: 5px; /* 添加右边距，调整"品牌"标签与文本之间的距离 */
}
.brand-more {
    margin-right: 8px; /* 设置词与词之间的空隙大小，调整数值以获得所需的间距 */
    display: inline;
    font-size: 12px; /* 设置字体大小 */
}

.search-container {
  margin-left: auto; /* 使用auto将搜索框和按钮推向最右侧 */
  display: flex;
  align-items: center;
}

/**价格样式--输入框 */
/**给选中的字体设置悬停和选中样式 */
.item:hover {
  color: rgba(53, 99, 233, 1);
}

.hasChecked {
  background-color: rgb(24, 126, 199);
  color: white;
  padding: 2px;
}
.input-row {
    overflow: hidden; /* 清除浮动 */
    margin-bottom: 0px;
}
.input-row el-input {
float: left;
margin-right: 10px; /* 可选，用于设置输入框间的水平间距 */
}
.custom-input {
    width: 60px;
    font-size: 6px;
    margin-top: 18px;
}

.custom-input2 {
    width: 250px;
```

```css
        height: 30px;
        font-size: 6px;
    }

    .button1 {
        margin-left: 10px;
        padding: 0.5%;
        font-size: 8px;
        background-color: rgba(53, 99, 233, 1);
        color: white;
    }

    .buttonPrice {
        margin-left: 10px;
        margin-top: 28px;
        padding: 7%;
        float: right;
        font-size: 8px;
        background-color: rgba(53, 99, 233, 1);
        color: white;
    }
/**已选择的样式 */
    .selected-option {
      font-size: 5px; /* 根据需求调整字体大小 */
      display: inline-block;
      padding: 1px;
      border: 1px solid #ccc;
      border-radius: 3px;
      margin-right: 5px;
      }

    .selected-option i {
      cursor: pointer;
    }


    /**下半部分卡片的样式 */
    .card-container {
      display: flex; /* 使用 flexbox 布局 */
      flex-wrap: wrap; /* 超出容器宽度时换行 */
      justify-content: space-between; /* 在容器内水平分布子元素 */
    }


    .card-item {
      width: calc(25% - 10px); /* 确保一行显示四个卡片，减去 margin 的宽度 */
      margin-bottom: 20px; /* 调整卡片之间的间距 */
    }


  .card-content {
```

```css
  height: 100%;
  display: flex;
  flex-direction: column;
  justify-content: space-between;
}

.card-content .title {
white-space: nowrap; /* 强制在一行内展示文本 */
overflow: hidden; /* 隐藏溢出的部分 */
text-overflow: ellipsis; /* 用省略号表示溢出的部分 */
}

.card-img {
  max-width: 100%; /* 图片最大宽度为卡片宽度 */
  max-height: 100%; /* 图片最大高度为卡片高度 */
  margin-bottom:10px;
}
.bottom-card {
background-color: rgba(255, 255, 255, 1);
}

.card-column {
padding-left: 2px; /* 调整卡片列的左内间距 */
padding-right: 2px;


}
.card-row {
    margin-left: 0px;
    margin-right:0px;
    margin-bottom: 15px;
}

.title {
font-size: 15px;
}

.car-info {
margin-top: 4px;
display: flex;
}

.price {
float: left;
font-size: 18px;
color:rgba(53, 99, 233, 1);
margin-top: 4px;
}

.extra-info {
font-size: 10px;
color: #999;
```

```css
  margin-top: 2px;
  display: flex;
  }
  .separator2 {
    margin: 0 5px;
    color: #999;
  }

  .buttondetails {
    width:70px;
  padding: 1%;
  float: right;
  font-size: 10px;
  background-color: rgba(53, 99, 233, 1);
  color: white;
  margin-top:-14px;
  }

  .price-container {
      margin-right: auto;
      margin-top: 10px;
  }
  .button-container {
      margin-left: auto;
  }
  .clearfix:before,
  .clearfix:after {
      display: table;
      content: "";
  }

  .clearfix:after {
      clear: both
  }
</style>
```