

```
1 <template>
2 <div style="width: 100%;">
3 <!-- Swiper 容器 -->
4 <div class="group_666" style="position: relative;min-height: 350px;">
5 <swiper class="swiper-container h-full" @swiper="onSwiperNews" v-bind="swiperOptionsNews
6 ">
7 <swiper-slide>
8 
9 </swiper-slide>
10 <swiper-slide>
11 
12 </swiper-slide>
13 </swiper>
14 <!-- <div style="position: absolute;bottom: 50px;left: 0;width: 100%;z-index: 100;">
15 <div class="flex-row justify-center align-center">
16 <div>
17 
18 </div>
19 <div style="margin: 0px 20px;">
20 
21 </div>
22 <div>
23 
24 </div>
25 </div>
26 </div> -->
27 </div>
28 <div class="group_2 flex-col">
29 <div class="box_2 flex-col">
30 <div class="group_3 flex-row justify-center" style="background: #ffffff;height: 400px;">
31 <!-- <div class="group_5 flex-col justify-center align-center rightBox">
32 <div class="flex-col section_1sa"></div>
33 </div> -->
34 <div class="group_4 flex-col leftBox" style="padding-top: 40px;">
35 <div class="text-group_1 flex-col" style="width: 100%;">
36 <span class="text_7" style="width: 100%;position: relative;text-align: center;">
37 洛阳灵睿网络技术有限公司
38 <!--  -->
40 </span>
41 <span class="text_8" style="width: 100%;color: #3D3D3D;text-align: center;">
42 智慧信息化与数字内容服务领航者
43 </span>
44 </div>
45 <span class="text_top_12" style="height: auto;font-size: 14px;margin-top: 40px;">
46 洛阳灵睿网络技术有限公司成立于 2006 年，深耕互联网行业，专注智慧信息化与数字内容
47 服务领域 19 年。是集技术研发、创意策划、全媒体运营于一体的高新技术企业。公司以“技
48 术赋能场景，内容驱动价值”为核心使命，专注于为大型公立医院、教育机构、企事业单位、
49 品牌营销策略提供一站式智慧化解决方案与品牌数字化升级服务，助力客户实现效率提升、
50 品牌增值与精准传播。
```

```
1 </span>
2 <span class="textss13">
3  专注互联网领域，发散新媒体创思，执着深耕口碑服务，扎实落地品牌内容，共创新信息化
4  与内容服务业态。
5 </span>
6 <div class="group_28 flex-row justify-between numberTopBox"
7  style="width: 592px;text-align: center;">
8  <div class="text-group_9 flex-col justify-between" style="width: auto;height: 80px;">
9  <div class="text_55" style="font-size: 42px;text-align: center;width: auto;">
10 <n-number-animation :duration="5000" ref="numberAnimationInstRef" :active="false"
11 :from="0" :to="1000"></n-number-animation>
12 <span style="font-size: 24px;">+</span>
13 </div>
14 <span class="">服务客户</span>
15 </div>
16 <div class="text-group_10 flex-col justify-between" style="width: auto;height: 80px;">
17 <div class="text_58"
18 style="font-size: 42px;text-align: center;width: auto;"><n-number-animation
19 :duration="5000" ref="numberAnimationInstRefKH" :active="false" :from="0"
20 :to="19"></n-number-animation>
21 <span style="font-size: 24px;">年</span>
22 </div>
23 <span class="">技术沉淀</span>
24 </div>
25 <div class="text-group_11 flex-col justify-between" style="width: auto;height: 80px;">
26 <div class="text_61"
27 style="font-size: 42px;text-align: center;width: auto;"><n-number-animation
28 :duration="5000" ref="numberAnimationInstRefJS" :active="false" :from="0"
29 :to="100"></n-number-animation>
30 <span style="font-size: 24px;">+</span>
31 </div>
32 <span class="">覆盖行业</span>
33 </div>
34 <div class="text-group_12 flex-col justify-between" style="width: auto;height: 80px;">
35 <div class="text_64"
36 style="font-size: 42px;text-align: center;width: auto;"><n-number-animation
37 :duration="5000" ref="numberAnimationInstRefHY" :active="false" :from="0"
38 :to="30"></n-number-animation>
39 <span style="font-size: 24px;">+</span>
40 </div>
41 <span class="">技术团队</span>
42 </div>
43 </div>
44 </div>
45 </div>
46 <div class="flex rongyu1s bottomBox">
47 <div class="flex" style="width: 14%;align-items: center;">
48 
49 <span class="spanst">荣誉标签： </span>
50 </div>
```

```
1 <div class="flex" style="width: 85%;overflow: hidden;">
2 <n-marquee>
3 <div class="spans" style="margin-right: 50px;display: inline-block;">中国电子商务协会工业电
4 商委员会河南分会单位
5 </div>
6 <div class="spans" style="display: inline-block;margin-right: 50px;">新华网、人民网、今日头
7 条新闻联络中心
8 </div>
9 <div class="spans" style="display: inline-block;margin-right: 50px;">
10 灵睿与郑州、洛阳等地的网信以及网监系统保持长期合作获</div>
11 </n-marquee>
12 </div>
13 </div>
14 </div>
15 </div>
16 <div class="group_2 flex-col">
17 <div class="box_2 flex-col">
18 <div class="group_3 flex-row justify-center align-center" style="background: #F8F8F8;">
19 <div class="group_4 flex-col leftBox">
20 <div class="text-group_1 flex-col justify-between" style="width: 100%;">
21 <span class="text_7" style="width: 242px;position: relative;">
22 睿创团队介绍
23 
25 </span>
26 </div>
27 <span class="text_12" style="height: auto;font-size: 14px;">
28 灵睿基因一直是一群年轻的青年人组成,我们致力打造专业的团队化运营模式,集合程序猿、
29 设计狗、运营喵的创作者联盟。年轻不是资本,敢重新定义年轻才是,在互联网领域,我们
30 致力做一支专业,积极,上进的年轻团队;80后主导,90后主力,00后紧随。以“为客户
31 创造价值 用互联网思维打造品牌”为使命。以“正直诚信、实干勇敢、协作共赢、创业精
32 神”为价值观。努力发展,破茧成蝶。
33 </span>
34 </div>
35 <div class="group_5 flex-col justify-center align-center rightBox">
36 <div class="flex-col section_1sb"></div>
37 </div>
38 </div>
39 </div>
40 </div>
41 <div class="box_7 flex-col">
42 <span class="text_24 topBox">著作权和荣誉证书</span>
43 <span class="text_15 topBox">Company&nbsp;environment</span>
44 <div class="dswper" style="margin-top: 30px;">
45 <swiper ref="swiper_exp" :slidesPerView="5" loop :loopedSlides="3" :autoplay="{ delay: 2000 }
46 "
47 centeredSlides :modules="modules" :watchSlidesProgress="true" @swiper="onSwiperExp">
48 <swiper-slide v-for="(item, index) in 21" :key="index" style="padding: 0 10px;">
49 
50 </swiper-slide>
```

```
1 </swiper>
2 <!-- <div class="dswper_left">
3 
4 </div>
5 <div class="dswper_right">
6 
7 </div> -->
8 <div class="rongyu_cla flex-row justify-center align-center absolute w-full" style="gap: 30px;">
9 <div v-for="( _ , index) in 3" :key="index" style="width: 25px; height: 5px;"
10 :style="{ backgroundColor: activeSlideIndex === index ? '#fc7428' : '#C1C1C1' }">
11 </div>
12 </div>
13 </div>
14 </div>
15 <div class="box_17 flex-col" style="overflow: hidden;">
16 <span class="text_53">部分伙伴</span>
17 <span class="text_54" style="margin-top: 30px;">Our&nbsp;Business&nbsp;</span>
18 <div class="group_29 grid grid-cols-6 gap-6 mr-auto">
19 <div class="bottom_imgs topSj"></div>
20 <!-- <div v-for="item in 18" class="">
21 
22 </div> -->
23 </div>
24 </div>
25 </div>
26 </template>
27 <script lang="ts" setup>
28 import { Swiper, SwiperSlide } from 'swiper/vue'
29 import { nextTick, ref } from 'vue'
30 import 'swiper/css'
31 import { Autoplay, Navigation, Pagination, Scrollbar, A11y, EffectCoverflow, EffectFade } from "
32 swiper/modules";
33 import ScrollReveal from 'scrollreveal';
34 import { NNumberAnimation, NMarquee } from 'naive-ui'
35 let modules = [Autoplay, A11y, EffectCoverflow, EffectFade];
36 import "swiper/css";
37 import "swiper/css/navigation";
38 import "swiper/css/pagination";
39 import 'swiper/css/effect-coverflow';
40 import 'swiper/css/mousewheel'
41 import 'swiper/css/grid'
42 import 'swiper/css/effect-fade'
43 import 'swiper/css/autoplay'
44 import * as swiperAni from '@assets/animate/animate.js'
45 import $api from '@service/webRequest'
46 import { useI18n } from 'vue-i18n'
47 import { useStore } from '~/store'
48 const { locale } = useI18n()
49 const cdnUrl = useCdn()
50 let newsSwiper: any = null
```

```
1  const onSwiperNews = (swiper: any) => {
2    newsSwiper = swiper
3  }
4  const swiperOptionsNews = {
5    autoplay: {
6      delay: 5000,
7      disableOnInteraction: false,
8    },
9    direction: 'horizontal',
10   slidesPerView: 1,
11   speed: 500,
12   effect: 'fade',
13   loop: true,
14   modules: [EffectFade, Autoplay, Navigation],
15   navigation: {
16     nextEl: '#swipen_prev',
17     prevEl: '#swipen_next',
18   },
19 }
20 const activeSlideIndex = ref(0);
21 let vesSwiper: any = null
22 const onSwiperExp = (swiper: any) => {
23   vesSwiper = swiper
24   swiper.on('slideChange', () => {
25     activeSlideIndex.value = swiper.realIndex % 3
26   })
27 }
28 const swiper_exp = ref();
29 onMounted(() => {
30   animate()
31 })
32 const numberAnimationInstRef = ref()
33 const numberAnimationInstRefKH = ref()
34 const numberAnimationInstRefJS = ref()
35 const numberAnimationInstRefHY = ref()
36 const animate = () => {
37   const sr = ScrollReveal();
38   sr.reveal('.leftBox', {
39     origin: "left",
40     distance: "1000px",
41     duration: 1300,
42     delay: 0,
43     opacity: 0,
44     scale: 0.9,
45     reset: true,
46     mobile: true,
47   })
48   sr.reveal('.rightBox', {
49     origin: "right",
50     distance: "1000px",
```

```
1 duration: 1300,
2 delay: 100,
3 opacity: 0,
4 scale: 0.9,
5 reset: true,
6 mobile: true,
7 })
8 sr.reveal('.topBox', {
9 origin: "top",
10 distance: "1000px",
11 duration: 1300,
12 delay: 100,
13 opacity: 0,
14 scale: 0.9,
15 reset: true,
16 mobile: true,
17 })
18 sr.reveal('.bottomBox', {
19 origin: "bottom",
20 distance: "1000px",
21 duration: 1300,
22 delay: 100,
23 opacity: 0,
24 scale: 0.9,
25 reset: true,
26 mobile: true,
27 })
28 sr.reveal('.numberTopBox', {
29 origin: "top",
30 distance: "1000px",
31 duration: 1300,
32 delay: 100,
33 opacity: 0,
34 scale: 0.9,
35 reset: true,
36 mobile: true,
37 beforeReveal: function (el: any) {
38 numberAnimationInstRef.value.play()
39 numberAnimationInstRefKH.value.play()
40 numberAnimationInstRefJS.value.play()
41 numberAnimationInstRefHY.value.play()
42 },
43 })
44 sr.reveal('.topSj', {
45 origin: "bottom",
46 distance: "1000px",
47 duration: 1300,
48 delay: 5,
49 opacity: 0,
50 scale: 0.9,
```

```
1  reset: true,
2  mobile: true,
3  useDelay: 'always',
4  interval: 50,
5  })
6  }
7  </script>
8  <style lang="scss" scoped>
9  @import "@/assets/animate/animate.min.css";
10 @import "@/assets/index.scss";
11 :deep(.dswper .swiper-slide-next) {
12 transform: translate3d(0px, 0px, -507px) rotateX(0deg) rotateY(0deg) scale(1) !important;
13 }
14 :deep(.dswper .swiper-slide-prev) {
15 transform: translate3d(0px, 0px, -507px) rotateX(0deg) rotateY(0deg) scale(1) !important;
16 }
17 .group_666 .swiper-slide img {
18 width: 100%;
19 height: 500px;
20 height: auto;
21 transition: 1s linear 2s;
22 transform: scale(1.1, 1.1);
23 }
24 .group_666 .swiper-slide-active img,
25 .swiper-slide-duplicate-active img {
26 transition: 6s linear;
27 transform: scale(1, 1);
28 }
29 .bottom_imgs {
30 background: var(--qall) no-repeat;
31 width: 1230px;
32 height: 272px;
33 background-size: 100% 100%;
34 margin: 0 auto;
35 }
36 .section_1sa {
37 height: 410px;
38 background: var(--t1a) 100% no-repeat;
39 width: 576px;
40 z-index: 10;
41 }
42 .section_1sb {
43 height: 410px;
44 background: var(--t1b) 100% no-repeat;
45 width: 576px;
46 z-index: 10;
47 }
48 .textss13 {
49 width: 100%;
50 font-weight: 600;
```

```
1  font-size: 14px;
2  color: #768597;
3  line-height: 22px;
4  font-family: Microsoft YaHei UI-Regular;
5  text-align: center;
6  }
7  .rongyu1s {
8  width: 63%;
9  height: 68px;
10 background: #F8F8F8;
11 border-radius: 0px 0px 0px 0px;
12 align-items: center;
13 margin: 0 auto;
14 margin-bottom: 70px;
15 .spanst {
16 width: 90px;
17 height: 32px;
18 font-family: Microsoft YaHei UI, Microsoft YaHei UI;
19 font-weight: bold;
20 font-size: 18px;
21 color: #3D3D3D;
22 line-height: 32px;
23 text-align: left;
24 font-style: normal;
25 text-transform: none;
26 margin-left: 20px;
27 }
28 .spans {
29 height: 21px;
30 font-family: Microsoft YaHei UI, Microsoft YaHei UI;
31 font-weight: 400;
32 font-size: 14px;
33 color: #3D3D3D;
34 line-height: 21px;
35 text-align: left;
36 font-style: normal;
37 text-transform: none;
38 }
39 }
40 .n-marquee {
41 width: 100%;
42 }
43 .box_7 {
44 background: var(--ruanzhu) 100% no-repeat;
45 }
46 .rongyu_cla {
47 margin-top: 50px;
48 }
49 .text_top_12 {
50 color: rgb(118, 133, 151);
```

```
1  font-size: 16px;
2  font-family: Microsoft YaHei UI-Regular;
3  font-weight: 400;
4  text-align: left;
5  line-height: 24px;
6  width: 63%;
7  text-align: center;
8  margin: 0 auto;
9  }
10 @media (max-width: 1440px) {
11  .rongyu1s {
12  width: 84%;
13  }
14  .box_7 {
15  height: 600px;
16  }
17  .rongyu_cla {
18  margin-top: 20px;
19  }
20  }
21 </style>
22 <template>
23 <div style="width: 100%;">
24 <!-- Swiper 容器 -->
25 <div class="group_666" style="position: relative;min-height: 400px;">
26 <swiper class="swiper-container h-full" @swiper="onSwiperNews" v-bind="swiperOptionsNews
27 ">
28 <swiper-slide>
29 
30 </swiper-slide>
31 <swiper-slide>
32 
33 </swiper-slide>
34 </swiper>
35 <!-- <div style="position: absolute;bottom: 50px;left: 0;width: 100%;z-index: 100;">
36 <div class="flex-row justify-center align-center">
37 <div>
38 
39 </div>
40 <div style="margin: 0px 20px;">
41 
42 </div>
43 <div>
44 
45 </div>
46 </div>
47 </div> -->
48 </div>
49 <div class="group_2 flex-col">
50 <div class="box_2 flex-col">
```

```
1 <div class="group_3 flex-row justify-center align-center" style="height: 505px;gap: 72px;">
2 <div class="group_5 flex-col justify-center align-center rightBox">
3 <div class="section_1s flex-col"></div>
4 </div>
5 <div class="group_4 flex-col leftBox">
6 <div class="text-group_1 flex-col justify-between">
7 <span class="text_7">洛阳灵睿网络技术有限公司</span>
8 </div>
9 <span class="text_12">
10 从一支年轻的创业团队，慢慢发展成为业务遍及全国的信息化，内容支持型互联网公司，我
11 们仍旧保持着始终创业的心态，每一天努力向前，如果你也愿意加入我们，请联系我们的
12 HR，期待与你的会面。
13 </span>
14 <div class="flex" style="margin-top: 40px;">
15 <span class="text_13s">
16 
17 联系电话：17603796680
18 </span>
19 <span class="text_13s">
20 
21 微信：17603796680
22 </span>
23 </div>
24 </div>
25 </div>
26 </div>
27 </div>
28 <div class="box_17s flex-col" style="overflow: hidden;">
29 <span class="text_53">热门岗位</span>
30 <span class="text_54" style="margin-top: 20px;">Company&nbsp;environment</span>
31 <div class="hbform">
32 <div class="flex boxgs">
33 <div v-for="(item, index) in typeList" @click="openType(item)" :class="typeId == item.id ? 'title
34 active' : 'title'">{{ item.name
35 }}</div>
36 </div>
37 <div class="flex justify-center align-center flex-wrap" style="margin-top: 30px;">
38 <div v-for="item in list" class="flex-col box_18s" style="margin-bottom: 20px;">
39 <div class="titless">
40 <div class="a1s">{{ item.name}}</div>
41 <div class="a2s">任职要求请联系 HR</div>
42 </div>
43 <div class="line"></div>
44 <div class="bottomss">
45 <div>灵睿网络</div>
46 <div>{{ item.release_time_text }}</div>
47 </div>
48 </div>
49 </div>
50 </div>
```

```
1 </div>
2 </div>
3 </template>
4 <script lang="ts" setup>
5 import { Swiper, SwiperSlide } from 'swiper/vue'
6 import { nextTick, ref } from 'vue'
7 import 'swiper/css'
8 import { Autoplay, Navigation, Pagination, Scrollbar, A11y, EffectCoverflow, EffectFade } from "
9 swiper/modules";
10 import ScrollReveal from 'scrollreveal';
11 import { NNumberAnimation } from 'naive-ui'
12 let modules = [Autoplay, A11y, EffectCoverflow, EffectFade];
13 import "swiper/css";
14 import "swiper/css/navigation";
15 import "swiper/css/pagination";
16 import 'swiper/css/effect-coverflow';
17 import 'swiper/css/mousewheel'
18 import 'swiper/css/grid'
19 import 'swiper/css/effect-fade'
20 import 'swiper/css/autoplay'
21 import * as swiperAni from '@assets/animate/animate.js'
22 import $api from '@service/webRequest'
23 import { useI18n } from 'vue-i18n'
24 import { useStore } from '~/store'
25 const { locale } = useI18n()
26 const cdnUrl = useCdn()
27 let newsSwiper: any = null
28 const onSwiperNews = (swiper: any) => {
29   newsSwiper = swiper
30 }
31 const swiperOptionsNews = {
32   autoplay: {
33     delay: 5000,
34     disableOnInteraction: false,
35   },
36   direction: 'horizontal',
37   slidesPerView: 1,
38   speed: 500,
39   effect: 'fade',
40   loop: true,
41   modules: [EffectFade, Autoplay, Navigation],
42   navigation: {
43     nextEl: '#swipen_prev',
44     prevEl: '#swipen_next',
45   },
46 }
47 onMounted(() => {
48   animate()
49   getType();
50 })
```

```
1  const openType = (item: any) => {
2    typeId.value = item.id;
3    getList();
4  }
5  const typeId = ref(0)
6  const typeList = ref([])
7  const getType = () => {
8    $api.post("/api/home.recruitment_information/cate", { page: 1, size: 100 })
9    .then((res: any) => {
10     console.log(res);
11     typeId.value = res.data.data.list[0].id;
12     typeList.value = res.data.data.list;
13     getList();
14   })
15   .catch((err) => {
16     console.dir(err)
17   })
18 }
19 const list = ref([])
20 const getList = () => {
21   $api.get("/api/home.recruitment_information/index?cate_id="+typeId.value)
22   .then((res: any) => {
23     console.log(res)
24     list.value = res.data.data.list
25   })
26   .catch((err) => {
27     console.dir(err)
28   })
29 }
30 const numberAnimationInstRef = ref()
31 const numberAnimationInstRefKH = ref()
32 const numberAnimationInstRefJS = ref()
33 const numberAnimationInstRefHY = ref()
34 const animate = () => {
35   const sr = ScrollReveal();
36   sr.reveal('.leftBox', {
37     origin: "left",
38     distance: "1000px",
39     duration: 1300,
40     delay: 100,
41     opacity: 0,
42     scale: 0.9,
43     reset: true,
44     mobile: true,
45   })
46   sr.reveal('.rightBox', {
47     origin: "right",
48     distance: "1000px",
49     duration: 1300,
50     delay: 100,
```

```
1  opacity: 0,
2  scale: 0.9,
3  reset: true,
4  mobile: true,
5  })
6  sr.reveal('.topBox', {
7  origin: "top",
8  distance: "1000px",
9  duration: 1300,
10 delay: 100,
11 opacity: 0,
12 scale: 0.9,
13 reset: true,
14 mobile: true,
15 })
16 sr.reveal('.bottomBox', {
17 origin: "bottom",
18 distance: "1000px",
19 duration: 1300,
20 delay: 100,
21 opacity: 0,
22 scale: 0.9,
23 reset: true,
24 mobile: true,
25 })
26 sr.reveal('.numberTopBox', {
27 origin: "top",
28 distance: "1000px",
29 duration: 1300,
30 delay: 100,
31 opacity: 0,
32 scale: 0.9,
33 reset: true,
34 mobile: true,
35 beforeReveal: function (el: any) {
36 numberAnimationInstRef.value.play()
37 numberAnimationInstRefKH.value.play()
38 numberAnimationInstRefJS.value.play()
39 numberAnimationInstRefHY.value.play()
40 },
41 })
42 sr.reveal('.topSj', {
43 origin: "bottom",
44 distance: "1000px",
45 duration: 1300,
46 delay: 5,
47 opacity: 0,
48 scale: 0.9,
49 reset: true,
50 mobile: true,
```

```
1  useDelay: 'always',
2  interval: 50,
3  })
4  }
5  </script>
6  <style lang="scss" scoped>
7  @import "@/assets/animate/animate.min.css";
8  @import "@/assets/index.scss";
9  :deep(.dswper .swiper-slide-next) {
10 transform: translate3d(0px, 0px, -507px) rotateX(0deg) rotateY(0deg) scale(1) !important;
11 }
12 :deep(.dswper .swiper-slide-prev) {
13 transform: translate3d(0px, 0px, -507px) rotateX(0deg) rotateY(0deg) scale(1) !important;
14 }
15 .group_666 .swiper-slide img {
16 width: 100%;
17 height: 500px;
18 height: auto;
19 transition: 1s linear 2s;
20 transform: scale(1.1, 1.1);
21 }
22 .group_666 .swiper-slide-active img,
23 .swiper-slide-duplicate-active img {
24 transition: 6s linear;
25 transform: scale(1, 1);
26 }
27 .section_1s {
28 height: 360px;
29 background: var(--allpeople) 100% no-repeat;
30 background-size: 100% 100%;
31 width: 585px;
32 z-index: 10;
33 }
34 .text_13s {
35 width: 261px;
36 height: 48px;
37 line-height: 44px;
38 border-radius: 4px 4px 4px 4px;
39 border: 1px solid #768597;
40 color: #768597;
41 font-size: 16px;
42 font-family: Microsoft YaHei UI-Regular;
43 font-weight: 400;
44 text-align: center;
45 white-space: nowrap;
46 margin-right: 22px;
47 display: flex;
48 justify-content: center;
49 align-items: center;
50 cursor: pointer;
```

```
1  .imgs {
2  width: 24px;
3  height: 24px;
4  margin-right: 10px;
5  }
6  }
7  .box_17s {
8  width: 100%;
9  background: #F8F8F8;
10 justify-content: flex-center;
11 padding-bottom: 40px;
12 }
13 .hbform {
14 width: 100%;
15 }
16 .boxgs {
17 width: 100%;
18 margin-top: 30px;
19 justify-content: center;
20 align-items: center;
21 .title {
22 width: 130px;
23 height: 48px;
24 background: #FFFFFF;
25 border-radius: 78px 78px 78px 78px;
26 margin: 0 20px;
27 line-height: 48px;
28 text-align: center;
29 font-weight: 400;
30 font-size: 18px;
31 color: #222222;
32 cursor: pointer;
33 &.active {
34 background: linear-gradient(360deg, #FC7428 0%, #FF9357 100%);
35 color: #FFFFFF;
36 }
37 }
38 }
39 .box_18s {
40 width: 387px;
41 height: 149px;
42 background: var(--lianxiyp) 100% no-repeat;
43 z-index: 10;
44 justify-content: center;
45 margin-right: 20px;
46 .line {
47 width: 347px;
48 height: 1px;
49 background: #D8D8D8;
50 margin: 0 auto;
```

```
1  }
2  .titless {
3    width: 100%;
4    padding-top: 20px;
5    padding-left: 20px;
6    .als {
7      font-family: Microsoft YaHei, Microsoft YaHei;
8      font-weight: 700;
9      font-size: 18px;
10     color: #3D3D3D;
11   }
12   .a2s {
13     margin: 15px 0;
14     font-family: Microsoft YaHei, Microsoft YaHei;
15     font-weight: 400;
16     font-size: 12px;
17     color: #768597;
18   }
19 }
20 .bottomss {
21   width: 100%;
22   display: flex;
23   justify-content: space-between;
24   align-items: center;
25   padding: 20px;
26   color: #768597;
27 }
28 &:hover {
29   cursor: pointer;
30   box-shadow: 0px 0px 20px 0px rgba(0, 0, 0, 0.1);
31 }
32 }
33 </style>
34 <template>
35 <div style="width: 100%;">
36 <!-- Swiper 容器 -->
37 <div class="group_666" style="position: relative;min-height: 450px;">
38 <swiper class="swiper-container h-full" @swiper="onSwiperNews" v-bind="swiperOptionsNews
39 ">
40 <swiper-slide>
41 
43 </swiper-slide>
44 <swiper-slide>
45 
47 </swiper-slide>
48 </swiper>
49 </div>
50 <div class="box_11 flex-col bottom_imgss">
```

```
1 <div class="text-wrapper_10" style="margin-top: 70px;">
2 <span class="text_41">公司荣誉</span>
3 </div>
4 <div class="text-wrapper_11" style="margin-top: 20px;">
5 <span class="text_42">Company&nbsp;Honors&nbsp;</span>
6 </div>
7 <div class="boxgs flex-row flex-wrap">
8 <div class="box_con flex-col" v-for="(item, index) in lists" :key="index">
9 <div class="align-center" style="justify-content: center;margin-top: 10px;">
10 
11 </div>
12 <div class="flex-col"
13 style="justify-content: center;margin-top: 10px;border-top: 1px solid #EEEEEE;">
14 <span class="text_tt">
15 {{ item.title }}
16 </span>
17 <span class="text_cc">
18 {{ item.value }}
19 </span>
20 </div>
21 </div>
22 </div>
23 </div>
24 </div>
25 </template>
26 <script lang="ts" setup>
27 import { Swiper, SwiperSlide } from 'swiper/vue'
28 import { nextTick, ref } from 'vue'
29 import 'swiper/css'
30 import { Autoplay, Navigation, Pagination, Scrollbar, A11y, EffectCoverflow, EffectFade } from "
31 swiper/modules";
32 import ScrollReveal from 'scrollreveal';
33 import { NNumberAnimation } from 'naive-ui'
34 let modules = [Autoplay, A11y, EffectCoverflow, EffectFade];
35 import "swiper/css";
36 import "swiper/css/navigation";
37 import "swiper/css/pagination";
38 import 'swiper/css/effect-coverflow';
39 import 'swiper/css/mousewheel'
40 import 'swiper/css/grid'
41 import 'swiper/css/effect-fade'
42 import 'swiper/css/autoplay'
43 import * as swiperAni from '@assets/animate/animate.js'
44 import $api from '@service/webRequest'
45 import { useI18n } from 'vue-i18n'
46 import { useStore } from '~/store'
47 const { locale } = useI18n()
48 const cdnUrl = useCdn()
49 let newsSwiper: any = null
50 const onSwiperNews = (swiper: any) => {
```

```
1  newsSwiper = swiper
2  }
3  const swiperOptionsNews = {
4  autoplay: {
5  delay: 5000,
6  disableOnInteraction: false,
7  },
8  direction: 'horizontal',
9  slidesPerView: 1,
10 speed: 500,
11 effect: 'fade',
12 loop: true,
13 modules: [EffectFade, Autoplay, Navigation],
14 navigation: {
15 nextEl: '#swipen_prev',
16 prevEl: '#swipen_next',
17 },
18 }
19 let vesSwiper: any = null
20 const onSwiperExp = (swiper: any) => {
21 vesSwiper = swiper
22 }
23 const swiper_exp = ref();
24 const lists = ref([
25 { title: '扶贫助困情系家乡', value: '-洛阳市郑州商会-' },
26 { title: '互联网技术服务合作单位', value: '-洛阳百艺建筑装饰-' },
27 { title: '平安洛阳微信平台技术服务单位', value: '-洛阳市公安局-' },
28 { title: '健康洛阳微信平台技术合作单位', value: '-河南科技大学第一附属医院-' },
29 { title: '爱心捐助奉献奖', value: '-洛阳市郑州商会-' },
30 { title: '精彩洛阳微信平台技术服务单位', value: '-洛阳市宣传部网信办-' },
31 { title: '青年说短视频征集大赛', value: '-洛阳市公安局-' },
32 { title: '青年说短视频征集大赛', value: '-河南科技大学第一附属医院-' }
33 ]);
34 onMounted() => {
35 nextTick() => {
36 animate()
37 })
38 })
39 const numberAnimationInstRef = ref()
40 const numberAnimationInstRefKH = ref()
41 const numberAnimationInstRefJS = ref()
42 const numberAnimationInstRefHY = ref()
43 const animate = () => {
44 const sr = ScrollReveal();
45 sr.reveal('.leftBox', {
46 origin: "left",
47 distance: "1000px",
48 duration: 1300,
49 delay: 100,
50 opacity: 0,
```

```
1  scale: 0.9,
2  reset: true,
3  mobile: true,
4  })
5  sr.reveal('.rightBox', {
6  origin: "right",
7  distance: "1000px",
8  duration: 1300,
9  delay: 100,
10 opacity: 0,
11 scale: 0.9,
12 reset: true,
13 mobile: true,
14 })
15 sr.reveal('.topBox', {
16 origin: "top",
17 distance: "1000px",
18 duration: 1300,
19 delay: 100,
20 opacity: 0,
21 scale: 0.9,
22 reset: true,
23 mobile: true,
24 })
25 sr.reveal('.bottomBox', {
26 origin: "bottom",
27 distance: "1000px",
28 duration: 1300,
29 delay: 100,
30 opacity: 0,
31 scale: 0.9,
32 reset: true,
33 mobile: true,
34 })
35 sr.reveal('.numberTopBox', {
36 origin: "top",
37 distance: "1000px",
38 duration: 1300,
39 delay: 100,
40 opacity: 0,
41 scale: 0.9,
42 reset: true,
43 mobile: true,
44 beforeReveal: function (el: any) {
45 numberAnimationInstRef.value.play()
46 numberAnimationInstRefKH.value.play()
47 numberAnimationInstRefJS.value.play()
48 numberAnimationInstRefHY.value.play()
49 },
50 })
```

```
1  sr.reveal('.topSj', {
2  origin: "bottom",
3  distance: "1000px",
4  duration: 1300,
5  delay: 5,
6  opacity: 0,
7  scale: 0.9,
8  reset: true,
9  mobile: true,
10 useDelay: 'always',
11 interval: 50,
12 })
13 }
14 </script>
15 <style lang="scss" scoped>
16 @import "@/assets/animate/animate.min.css";
17 @import "@/assets/index.scss";
18 .group_666 .swiper-slide img {
19 width: 100%;
20 height: 500px;
21 height: auto;
22 transition: 1s linear 2s;
23 transform: scale(1.1, 1.1);
24 }
25 .group_666 .swiper-slide-active img,
26 .swiper-slide-duplicate-active img {
27 transition: 6s linear;
28 transform: scale(1, 1);
29 }
30 .bottom_imgss {
31 background: var(--bg-s);
32 height: 837px;
33 background-size: 100% 100%;
34 }
35 .boxgs {
36 width: 1300px;
37 margin: 0 auto;
38 margin-top: 30px;
39 .box_con {
40 margin: 20px;
41 background-color: #ffffff;
42 width: 277px;
43 height: 276px;
44 font-family: Microsoft YaHei UI-Regular;
45 border-radius: 4px;
46 .text_tt {
47 display: inline-block;
48 width: 100%;
49 margin-top: 15px;
50 overflow-wrap: break-word;
```

```
1  color: #3D3D3D;
2  font-size: 18px;
3  font-weight: 500;
4  text-align: center;
5  white-space: nowrap;
6  }
7  .text_cc {
8  display: inline-block;
9  width: 100%;
10 height: 30px;
11 overflow-wrap: break-word;
12 color: #999999;
13 font-size: 14px;
14 font-weight: 400;
15 text-align: center;
16 line-height: 30px;
17 }
18 }
19 }
20 </style>
21 <template>
22 <div style="width: 100%;">
23 <!-- Swiper 容器 -->
24 <div class="group_666" style="position: relative;height: 100vh;">
25 <swiper class="swiper-container h-full" @swiper="onSwiperNews" v-bind="swiperOptionsNews
26 ">
27 <swiper-slide>
28 <video class="video_1" autoplay="autoplay" loop="loop" muted="muted" playsinline="playsinline"
29 e"
30 :src="\${cdnUrl}/img/1212.mp4" style="width: 100%;height:100%;object-fit: cover;"></video>
31 </swiper-slide>
32 <swiper-slide>
33 
34 </swiper-slide>
35 <swiper-slide>
36 
37 </swiper-slide>
38 </swiper>
39 <div style="position: absolute;bottom: 50px;left: 0;width: 100%;z-index: 100;">
40 <div class="flex-row justify-center align-center">
41 <div id="swipen_prev" style="cursor: pointer;">
42 
43 </div>
44 <div style="margin: 0px 20px;">
45 
46 </div>
47 <div id="swipen_next" style="cursor: pointer;">
48 
49 </div>
50 </div>
```

```
1 </div>
2 </div>
3 <div class="group_2 flex-col">
4 <div class="box_2 flex-col">
5 <div class="group_3 flex-row justify-center align-center">
6 <div class="group_4 flex-col leftBox">
7 <div class="text-group_1 flex-col justify-between">
8 <span class="text_7">灵睿的互联网之道</span>
9 <span class="text_8">
10 The&nbsp;Way&nbsp;of&nbsp;the&nbsp;Internet
11 </span>
12 </div>
13 <div style="display: flex;justify-content: start;margin-top: 50px;">
14 <div @click="openType(1)" :class="typeId == 1 ? 'text_9' : 'text_11'">公司简介</div>
15 <div @click="openType(2)" :class="typeId == 2 ? 'text_9' : 'text_11'">运营方法论</div>
16 <div @click="openType(3)" :class="typeId == 3 ? 'text_9' : 'text_11'">互联网营销</div>
17 </div>
18 <div class="box_3 flex-col" style="justify-content: start;">
19 <div v-if="typeId == 1" class="box_4 flex-col"></div>
20 <div v-if="typeId == 2" class="box_4 flex-col" style="margin-left: 135px;"></div>
21 <div v-if="typeId == 3" class="box_4 flex-col" style="margin-left: 287px;"></div>
22 </div>
23 <div class="text_12" v-if="typeId == 1">
24 洛阳灵睿网络技术有限公司
25 <br>
26 ——智慧信息化与数字内容服务领航者
27 <br>
28 洛阳灵睿网络技术有限公司成立于 2006 年，深耕互联网行业，专注智慧信息化与数字内容
29 服务领域 19 年。是集技术研发、创意策划、全媒体运营于一体的高新技术企业。公司以“技
30 术赋能场景，内容驱动价值”为核心使命，专注于为大型公立医院、教育机构、企事业单位、
31 品牌营销策略提供一站式智慧化解决方案与品牌数字化升级服务，助力客户实现效率提升、
32 品牌增值与精准传播。
33 <br>
34 专注互联网领域，发散新媒体创思，执着深耕口碑服务，扎实落地品牌内容，共创新信息化
35 与内容服务业态。
36 </div>
37 <div class="text_12" v-if="typeId == 2">
38 在灵睿，我们的运营方法论是基于深入的<strong>市场洞察、用户需求分析</strong>和科学的
39 <strong>数据驱动</strong>，通过系统化的策略为品牌赋能。我们从运营的<strong>江湖
40 地位</strong>出发，帮助品牌在竞争激烈的市场中确立<strong>独特的定位和竞争力；
41 </strong>通过精确的运营基本逻辑，确保每一项策略与品牌核心价值相匹配，推动<strong>
42 可持续发展。</strong>
43 我们将运营分为三大核心手段：<strong>内容运营、活动运营和用户运营</strong>。内容运
44 营注重通过高质量、有共鸣的内容与受众建立深度联系；活动运营则通过创意和精准策划，
45 提升品牌曝光和用户参与度；用户运营关注从触达到转化的全过程，通过个性化服务增强用
46 户粘性与忠诚度。
47 灵睿深知，运营不仅依赖直觉与经验，更依<strong>赖数据支持。</strong>我们通过精细化
48 的<strong>数据统计与分析</strong>，实时监测运营效果并做出策略调整，
49 确保每个决策都有<strong>数据背书</strong>。通过这一整套运营体系，我们不仅能为客户
50 提供行之有效的解决方案，更能帮助他们在数字化浪潮中领先一步，实现<strong>持续增长
```

```
1  与价值创造。</strong>
2  </div>
3  <div class="text_12" v-if="typeId == 3">
4  在灵睿，我们通过深度的<strong>数据分析</strong>和精确的<strong>用户洞察</strong>，帮
5  助企业实现<strong>品牌的精准传播和长期增长。</strong>我们相信，成功的互联网营销不
6  仅仅是提高曝光率，更是通过<strong>内容创作、活动策划和用户互动</strong>等手段，打
7  造与受众的深层次连接，从而促进<strong>品牌的认知度和忠诚度。</strong>
8  我们帮助客户制定个性化的互联网营销策略，覆盖<strong>社交媒体、搜索引擎、视频内容
9  </strong>等多个渠道，确保每个接触点都能精准传达品牌价值。在实际操作中，我们依托
10 <strong>技术优势</strong>，结合行业<strong>最佳实践</strong>，为客户提供<strong>定制
11  化的营销方案</strong>，推动<strong>线上线下的无缝连接。</strong>
12  灵睿的互联网营销不仅是单一的推广行为，更是一个以用户为中心的全方位战略，帮助企业
13  在快速变化的市场中保持<strong>竞争力</strong>，实现<strong>持续增长。</strong>
14 </div>
15 <div class="text-wrapper_2 flex-col">
16 <nuxt-link to="/aboutUs/" class="text_13">
17 <span>查看全部</span>
18 </nuxt-link>
19 </div>
20 </div>
21 <div @click="videoShow = true"
22 class="group_5 flex-col justify-center align-center rightBox cursor-pointer">
23 <div class="section_1 flex-col">
24 <div class="image-wrapper_2 flex-col">
25 
26 </div>
27 </div>
28 <div class="content_box"></div>
29 </div>
30 </div>
31 </div>
32 </div>
33 <div class="group_2 flex-col">
34 <div class="box_5 flex-col">
35 <span class="text_14 topBox">我们的业务</span>
36 <span class="text_15 topBox">
37 Our&nbsp;BusinessWay
38 </span>
39 <div class="flex-row justify-center align-center" style="margin-top: 30px;">
40 <div class="leftBox section_2s flex-col" style="margin-right: 30px;">
41 <div class="group_6 flex-col">
42 <div class="box_6 flex-row align-center" style="width: 100%;">
43 <div class="text-group_2 flex-col justify-between">
44 <span class="text_17" style="color: #0256FF;">
45 专业技术服务
46 </span>
47 <span class="text_16" style="color: #0256FF;">
48 专业提供互联网技术支持，助力企业高效发展
49 </span>
50 </div>
```

```
1 <div class="image-wrapper_3s flex-col align-center">
2 
3 </div>
4 </div>
5 </div>
6 <div class="group_10s flex-col"></div>
7 <div class="group_7 flex-row justify-between">
8 <div class="image-wrapper_5 flex-col">
9 
11 </div>
12 <div class="text-group_3 flex-col justify-between">
13 <span class="text_18">AI 应用与落地</span>
14 <span class="text_19">
15 从模型选型、数据处理、算法优化到应用集成与部署的全栈服务
16 </span>
17 </div>
18 </div>
19 <div class="group_10 flex-col"></div>
20 <div class="group_7 flex-row justify-between">
21 <div class="image-wrapper_5 flex-col">
22 
24 </div>
25 <div class="text-group_3 flex-col justify-between">
26 <span class="text_18">定制软件开发</span>
27 <span class="text_19">
28 为您打造用户体验出色、性能稳定可靠的 Web 应用、移动 App 及小程序
29 </span>
30 </div>
31 </div>
32 <div class="group_10 flex-col"></div>
33 <div class="group_7 flex-row justify-between">
34 <div class="image-wrapper_5 flex-col">
35 
37 </div>
38 <div class="text-group_3 flex-col justify-between">
39 <span class="text_18">IT 运维服务</span>
40 <span class="text_19">
41 专业守护数据，防御威胁，护航企业数字安全
42 </span>
43 </div>
44 </div>
45 <div class="group_10 flex-col"></div>
46 <div class="group_7 flex-row justify-between">
47 <div class="image-wrapper_5 flex-col">
48 
50 </div>
```

```
1 <div class="text-group_3 flex-col justify-between">
2 <span class="text_18">网络与信息安全</span>
3 <span class="text_19">
4 专业提供网络信息安全防护、实时监测与风险处置服务
5 </span>
6 </div>
7 </div>
8 <div class="group_10 flex-col"></div>
9 <div class="group_7 flex-row justify-between">
10 <div class="image-wrapper_5 flex-col">
11 
13 </div>
14 <div class="text-group_3 flex-col justify-between">
15 <span class="text_18">智慧化解决方案</span>
16 <span class="text_19">
17 利用物联网、大数据、人工智能等前沿技术，提供智慧化解决方案
18 </span>
19 </div>
20 </div>
21 </div>
22 <div class="rightBox section_2sa flex-col">
23 <div class="group_6s flex-col">
24 <div class="box_6 flex-row align-center" style="width: 100%;">
25 <div class="text-group_2 flex-col justify-between">
26 <span class="text_17">
27 内容创作与运营
28 </span>
29 <span class="text_16" style="color: #FC7428;">
30 构建全域影响力，赋能品牌长效增长
31 </span>
32 </div>
33 <div class="image-wrapper_3s flex-col align-center">
34 
35 </div>
36 </div>
37 </div>
38 <div class="group_10s flex-col"></div>
39 <div class="group_7 flex-row justify-between">
40 <div class="image-wrapper_5 flex-col">
41 
43 </div>
44 <div class="text-group_3_3 flex-col justify-between">
45 <span class="text_18">全媒体推广解决方案</span>
46 <span class="text_19">
47 覆盖各大主流平台定制【流量矩阵+内容策略+效果追踪】一体化方案
48 </span>
49 </div>
50 </div>
```

```
1 <div class="group_10 flex-col"></div>
2 <div class="group_7 flex-row justify-between">
3 <div class="image-wrapper_5 flex-col">
4 
5 </div>
6 <div class="text-group_3_3 flex-col justify-between">
7 <span class="text_18">品牌账号代运营</span>
8 <span class="text_19">
9 全链路赋能品牌增长，内容+策略+数据 • 打造品效合一的社交资产
10 </span>
11 </div>
12 </div>
13 <div class="group_10 flex-col"></div>
14 <div class="group_7 flex-row justify-between">
15 <div class="image-wrapper_5 flex-col">
16 
18 </div>
19 <div class="text-group_3_3 flex-col justify-between">
20 <span class="text_18">线上线下活动策划</span>
21 <span class="text_19">
22 专业守护数据，防御威胁，护航企业数字安全
23 </span>
24 </div>
25 </div>
26 <div class="group_10 flex-col"></div>
27 <div class="group_7 flex-row justify-between">
28 <div class="image-wrapper_5 flex-col">
29 
30 </div>
31 <div class="text-group_3_3 flex-col justify-between">
32 <span class="text_18">互联网新闻发布</span>
33 <span class="text_19">
34 整合全网权威渠道，依托海量优质媒体资源，定制新闻发布策略
35 </span>
36 </div>
37 </div>
38 <div class="group_10 flex-col"></div>
39 <div class="group_7 flex-row justify-between">
40 <div class="image-wrapper_5 flex-col">
41 
42 </div>
43 <div class="text-group_3_3 flex-col justify-between">
44 <span class="text_18">IP 孵化与达人合作</span>
45 <span class="text_19">
46 深度孵化+精准匹配，内容资产×流量裂变 • 打造 IP 与达人的共生生态
47 </span>
48 </div>
49 </div>
50 </div>
```

```
1 </div>
2 </div>
3 </div>
4 <div style="background-color: #ffffff;">
5 <div class="box_7 flex-col">
6 <span class="text_24 topBox">行业案例</span>
7 <span class="text_15 topBox">Industry&nbsp;Cases</span>
8 <div class="dswper">
9 <swiper ref="swiper_exp" :slidesPerView="2" loop :autoplay="false" :modules="modules"
10 :watchSlidesProgress="true" @swiper="onSwiperExp">
11 <swiper-slide v-for="(item, index) in 14" :key="index">
12 
13 </swiper-slide>
14 </swiper>
15 <div class="flex-row justify-center align-center absolute w-full bottom-[-50px]" style="gap: 30px;
16 ">
17 <div v-for="(, index) in 3" :key="index" style="width: 20px; height: 5px;"
18 :style="{ backgroundColor: activeSlideIndex === index ? '#fc7428' : '#C1C1C1' }">
19 </div>
20 </div>
21 </div>
22 </div>
23 </div>
24 <div class="box_11 flex-col">
25 <div class="text-wrapper_10">
26 <span class="text_41">新闻动态</span>
27 </div>
28 <div class="text-wrapper_11">
29 <span class="text_42">news information</span>
30 </div>
31 <div class="box_12 flex-row justify-center" v-if="newsList.length > 0">
32 
35 <div class="news_my_home box_13 flex-col topBox" @click="openUrl(newsList[1].id)">
36 <div class="text-group_6">
37 <div class="one-line-ellipsis news_title_home">
38 {{ newsList[1].title }}
39 </div>
40 <div class="two-line-ellipsis"
41 style="color: #768597;font-size: 14px;font-weight: 400;margin-top: 20px;">
42 {{ newsList[1].subtitle }}
43 </div>
44 </div>
45 <div class="group_22 flex-row">
46 <div class="section_6 flex-col"></div>
47 <span class="text_44">{{ newsList[1].cate.name }}</span>
48 <span class="text_45">{{ retTime(newsList[1].release_time_text) }}</span>
49 </div>
50 </div>
```

```
1 
3 </div>
4 <div class="box_14 flex-row justify-center" v-if="newsList.length > 0">
5 <div class="news_my_home group_23 flex-col leftBoxTop" @click="openUrl(newsList[0].id)"
6 style="cursor: pointer;">
7 <div class="text-group_7">
8 <div class="news_title_home one-line-ellipsis">
9 {{ newsList[0].title }}
10 </div>
11 <div class="two-line-ellipsis"
12 style="color: #768597;font-size: 14px;font-weight: 400;margin-top: 20px;">
13 {{ newsList[0].subtitle }}
14 </div>
15 </div>
16 <div class="group_24 flex-row">
17 <div class="group_25 flex-col"></div>
18 <span class="text_47">{{ newsList[0].cate.name }}</span>
19 <span class="text_48">{{ retTime(newsList[0].release_time_text) }}</span>
20 </div>
21 </div>
22 
25 <div class="news_my_home group_26 flex-col rightBox" @click="openUrl(newsList[2].id)"
26 style="cursor: pointer;">
27 <div class="text-group_8">
28 <div class="text-group_7" style="margin: 0;">
29 <div class="news_title_home one-line-ellipsis">
30 {{ newsList[2].title }}
31 </div>
32 <div class="two-line-ellipsis"
33 style="color: #768597;font-size: 14px;font-weight: 400;margin-top: 20px;">
34 {{ newsList[2].subtitle }}
35 </div>
36 </div>
37 </div>
38 <div class="box_15 flex-row">
39 <div class="group_27 flex-col"></div>
40 <span class="text_50">{{ newsList[2].cate.name }}</span>
41 <span class="text_51">{{ retTime(newsList[2].release_time_text) }}</span>
42 </div>
43 </div>
44 </div>
45 <div class="box_16 flex-row">
46 <div class="text-wrapper_12 flex-col">
47 <nuxt-link to="/societyDutyNew/" class="text_52">
48 <span>查看更多</span>
49 </nuxt-link>
50 </div>
```

```
1 </div>
2 </div>
3 <div class="box_17 flex-col" style="overflow: hidden;">
4 <span class="text_53">合作伙伴</span>
5 <span class="text_54">Our&nbsp;Business&nbsp;</span>
6 <div class="group_29 grid grid-cols-6 gap-6 mr-auto">
7 <div class="bottom_imgs topSj"></div>
8 </div>
9 </div>
10 <div class="box_18 flex-col" style="overflow: hidden;">
11 <span class="text_53" style="color: #fff;">联系我们</span>
12 <span class="text_54" style="color: #fff;">cooperative&nbsp;partner&nbsp;</span>
13 <div class="numberTopBox" style="margin:0 auto;">
14 <form class="contactForm mt-6 w-full" @submit.prevent="submitForm">
15 <div class="inputform">
16 <input v-model="formData.name" class="contactInput mt-[15px] md:mt-[30px] w-full md:w-[70
17 0px]"
18 type="text" placeholder="输入您的姓名" />
19 <input v-model="formData.mobile"
20 class="contactInput mt-[15px] md:ml-[30px] md:mt-[30px] w-full md:w-[700px]" type="text"
21 placeholder="请输入您的联系电话" />
22 <input v-model="formData.emil"
23 class="contactInput mt-[15px] md:ml-[30px] md:mt-[30px] w-full md:w-[700px]" type="text"
24 placeholder="请输入您的邮箱" />
25 </div>
26 <div class="relative mt-4 md:mt-8 w-full md:w-[1200px]">
27 <textarea v-model="formData.message" id="myTextarea" placeholder="为了更好地帮助您，请
28 尽量提供详细的信息" rows="5"
29 maxLength="100" class="w-full md:w-[1200px]"></textarea>
30 <button type="submit" class="contactBut">
31 {{ '提交' }}
32 </button>
33 </div>
34 </form>
35 </div>
36 </div>
37 <div v-if="videoShow" class="pop" style="display: block;">
38 <div class="abs popCont pop_video_cont">
39 <a @click="videoShow = false" class="abs pop_video_close closePop"></a>
40 <video x5-playsinline="" playsinline="" webkit-playsinline="" preload=""
41 x5-video-orientation="portrait" controls="" autoplay :src="{{cdnUrl}}/img/222.mp4`"
42 draggable="true"></video>
43 </div>
44 </div>
45 </div>
46 </template>
47 <script lang="ts" setup>
48 import { Swiper, SwiperSlide } from 'swiper/vue'
49 import { nextTick, ref } from 'vue'
50 import 'swiper/css'
```

```
1 import { Autoplay, Navigation, Pagination, Scrollbar, A11y, EffectCoverflow, EffectFade } from "
2 swiper/modules";
3 import ScrollReveal from 'scrollreveal';
4 import { NNumberAnimation } from 'naive-ui'
5 let modules = [Autoplay, A11y, EffectCoverflow, EffectFade];
6 import "swiper/css";
7 import "swiper/css/navigation";
8 import "swiper/css/pagination";
9 import 'swiper/css/effect-coverflow';
10 import 'swiper/css/mousewheel'
11 import 'swiper/css/grid'
12 import 'swiper/css/effect-fade'
13 import 'swiper/css/autoplay'
14 import * as swiperAni from '@assets/animate/animate.js'
15 import $api from '@service/webRequest'
16 import { useI18n } from 'vue-i18n'
17 import { useStore } from '~/store'
18 const cdnUrl = useCdn()
19 const { locale } = useI18n()
20 const typeId = ref(1);
21 const openType = (type) => {
22   console.log(type);
23   typeId.value = type;
24 }
25 const videoShow = ref(false)
26 let newsSwiper: any = null
27 const onSwiperNews = (swiper: any) => {
28   newsSwiper = swiper
29 }
30 const swiperOptionsNews = {
31   autoplay: {
32     delay: 11000,
33     disableOnInteraction: false,
34   },
35   direction: 'horizontal',
36   slidesPerView: 1,
37   speed: 500,
38   effect: 'fade',
39   loop: true,
40   modules: [EffectFade, Autoplay, Navigation],
41   navigation: {
42     nextEl: '#swipen_prev',
43     prevEl: '#swipen_next',
44   },
45 }
46 let vesSwiper: any = null
47 const onSwiperExp = (swiper: any) => {
48   vesSwiper = swiper
49   swiper.on('slideChange', () => {
50     activeSlideIndex.value = swiper.realIndex % 3
```

```
1 <?php
2 namespace app\adminapi\controller\home;
3 use app\common\controller\AdminApi;
4 use app\adminapi\model\home\Images as ImagesModel;
5 /**
6  * 官网首页轮播图
7  */
8 class Images extends AdminApi
9 {
10     protected $model = null;
11     /**
12      * 初始化操作
13      * @access protected
14      */
15     public function _initialize()
16     {
17         $this->model = new ImagesModel;
18         parent::_initialize();
19     }
20     /**
21      * @ApiTitle( 轮播图列表)
22      * @ApiSummary(轮播图列表)
23      * @ApiMethod(GET)
24      * @ApiParams(name = "keywords", type = "string",required=false,description = "搜索关键字
25 ")
26      * @ApiParams(name = "page", type = "string",required=true,description = "页数")
27      * @ApiParams(name = "limit", type = "string",required=true,description = "条数")
28      * @ApiParams(name = "type", type = "string",required=true,description = "跳转类型:in=内部
29 跳转,out=外部跳转")
30      * @ApiParams(name = "showtype", type = "string",required=false,description = "展示类
31 型:image=图片,video=视频")
32      * @ApiReturn({
33      *
34      *})
35      */
36     public function index()
37     {
38         $user_id = 0;
39         $user = $this->auth->getUser();//登录用户
40         if($user)$user_id = $user['id'];
41         $page = $this->request->get('page/d', 0); //页数
42         $limit = $this->request->get('limit/d', 0); //条数
43         $params = [];
44         $params["keywords"] = $this->request->get('keywords/s', ''); //搜索关键字
45         $params["showtype"] = $this->request->get('showtype/s', ''); //搜索关键字
46         $params["type"] = $this->request->get('type/s', ''); //筛选学员和教练单
47         try{
48             //当前申请状态
49             $res = $this->model::allList($page, $limit,$params);
50             // if($user_id=='670153'){
```

```
1 //      file_put_contents("ceshi66.txt",(new AppointmentOrder()->getLastSql());
2 //      }
3     } catch (\Exception $e){
4         $this->error($e->getMessage());
5     }
6     $this->success('查询成功', $res);
7 }
8 /**
9  * 查看轮播图详情
10  *
11  * @ApiMethod (GET)
12  * @ApiParams (name="id", type="string", required=true, description="ID")
13  */
14 public function detail()
15 {
16     $admin_id = $this->auth->id;
17     $id = $this->request->get('id/d');
18     try{
19         $menulist = $this->model->detail($id,$show_field=[],$except_field=[]);
20     } catch (\Exception $e) {
21         $this->error($e->getMessage());
22     }
23     $this->success('查询成功', $menulist);
24 }
25 /**
26  * 删除轮播图
27  *
28  * @ApiMethod (POST)
29  * @ApiParams (name="ids", type="string", required=true, description="要删除的 ids")
30  */
31 public function del()
32 {
33     //      $admin_id = $this->auth->id;
34     $ids = $this->request->post('ids/s');
35     try{
36         $menulist = $this->model->del($ids,true);
37     } catch (\Exception $e) {
38         $this->error($e->getMessage());
39     }
40     $this->success('删除成功', $menulist);
41 }
42 /**
43  * @ApiTitle(添加轮播图)
44  * @ApiSummary(添加轮播图)
45  * @ApiMethod(POST)
46  * @ApiParams(name = "title", type = "string",required=true,description = "轮播图标题")
47  * @ApiParams(name = "image", type = "string",required=true,description = "展示图")
48  * @ApiParams(name = "type", type = "int",required=true,description = "跳转类型:in=内部跳
49 转,out=外部跳转")
50  * @ApiParams(name = "showtype", type = "string",required=true,description = "展示类
```

```

1  型:image=图片,video=视频")
2      * @ApiParams(name = "url", type = "string",required=true,description = "跳转链接")
3      * @ApiParams(name = "weigh", type = "string",required=false,description = "权重")
4      * @ApiResponse({
5          *
6      *})
7      */
8      public function add(){
9          //      $user_id = 73;
10         //      $user = $this->auth->getUser();//登录用户
11         //      if($user)$user_id = $user['id'];
12         $params = [];
13         //      $params["user_id"] = $user_id; //老师 id
14         $params["image"] = $this->request->post('image/s, '); //课程标签
15         $params["title"] = $this->request->post('title/s, '); //课程标签
16         $params["type"] = $this->request->post('type/s, '); //课程标签
17         $params["showtype"] = $this->request->post('showtype/s, '); //老师 id
18         $params["url"] = $this->request->post('url/s, ');
19         $params["weigh"] = $this->request->post('weigh/s, '); //老师 id
20         try{
21             $res = $this->model->add($params,true);
22         } catch (\Throwable $e){
23             $this->error($e->getMessage());
24         }
25         $this->success('添加成功', $res);
26     }
27     /**
28     * @ApiTitle(编辑轮播图)
29     * @ApiSummary(编辑轮播图)
30     * @ApiMethod(POST)
31     * @ApiRoute (/adminapi/home.images/edit/ids/{ids})
32     * @ApiParams (name="ids", type="string", required=true, description="需要编辑的 id")
33     * @ApiParams(name = "title", type = "string",required=true,description = "轮播图标题")
34     * @ApiParams(name = "image", type = "string",required=true,description = "展示图")
35     * @ApiParams(name = "type", type = "int",required=true,description = "跳转类型:in=内部跳
36 转,out=外部跳转")
37     * @ApiParams(name = "showtype", type = "string",required=true,description = "展示类
38 型:image=图片,video=视频")
39     * @ApiParams(name = "url", type = "string",required=true,description = "跳转链接")
40     * @ApiParams(name = "weigh", type = "string",required=false,description = "权重")
41     * @ApiResponse({
42         *
43     *})
44     */
45     public function edit($ids = null){
46         //      $user_id = 73;
47         //      $user = $this->auth->getUser();//登录用户
48         //      if($user)$user_id = $user['id'];
49         $params = [];
50         //      $params["user_id"] = $user_id; //老师 id

```

```
1      $params["image"] = $this->request->post('image/s', ''); //课程标签
2      $params["title"] = $this->request->post('title/s', ''); //课程标签
3      $params["type"] = $this->request->post('type/s', ''); //课程标签
4      $params["showtype"] = $this->request->post('showtype/s', ''); //老师 id
5      $params["url"] = $this->request->post('url/s', '');
6      $params["weigh"] = $this->request->post('weigh/s', ''); //老师 id
7      try{
8          $res = $this->model->edit($ids,$params,true);
9      } catch (\Throwable $e){
10         $this->error($e->getMessage());
11     }
12     $this->success('编辑成功', $res);
13 }
14 }<?php
15 namespace app\adminapi\controller\home;
16 use app\common\controller\AdminApi;
17 use app\adminapi\model\home\LeaveWord as LeaveWordModel;
18 /**
19  * 网站留言管理
20  */
21 class LeaveWord extends AdminApi
22 {
23     /**
24      * @var \app\adminapi\model\home\LeaveWord
25      */
26     protected $model = null;
27     public function _initialize()
28     {
29         parent::_initialize();
30         $this->model = new LeaveWordModel;
31     }
32     /**
33      * @ApiTitle( 网站留言列表)
34      * @ApiSummary(网站留言列表)
35      * @ApiMethod(GET)
36      * @ApiParams(name = "keywords", type = "string",required=false,description = "搜索关键字
37 ")
38      * @ApiParams(name = "page", type = "string",required=true,description = "页数")
39      * @ApiParams(name = "limit", type = "string",required=true,description = "条数")
40      * @ApiParams(name = "status", type = "string",required=true,description = "状态:1=未处
41 理,2=已处理")
42      * @ApiParams(name = "mobile", type = "string",required=false,description = "联系电话")
43      * @ApiReturn({
44      *
45      *})
46      */
47     public function index()
48     {
49         $user_id = 0;
50         $user = $this->auth->getUser();//登录用户
```

```
1      if($user)$user_id = $user['id'];
2      $page = $this->request->get('page/d', 0); //页数
3      $limit = $this->request->get('limit/d', 0); //条数
4      $params = [];
5      $params["keywords"] = $this->request->get('keywords/s', ""); //搜索关键字
6      $params["status"] = $this->request->get('status/s', ""); //搜索关键字
7      $params["mobile"] = $this->request->get('mobile/s', ""); //搜索关键字
8      // $type = $this->request->get('type/s', ""); //筛选学员和教练单
9      try{
10         //当前申请状态
11         $res = $this->model::allList($page, $limit,$params);
12         // if($user_id=='670153'){
13         //     file_put_contents("ceshi66.txt",(new AppointmentOrder()->getLastSql());
14         // }
15     } catch (\Exception $e){
16         $this->error($e->getMessage());
17     }
18     $this->success('查询成功', $res);
19 }
20 /**
21  * 查看网站留言详情
22  *
23  * @ApiMethod (GET)
24  * @ApiParams (name="id", type="string", required=true, description="ID")
25  */
26 public function detail()
27 {
28     $admin_id = $this->auth->id;
29     $id = $this->request->get('id/d');
30     try{
31         $menulist = $this->model->detail($id,$show_field=[],$except_field=[]);
32     } catch (\Exception $e) {
33         $this->error($e->getMessage());
34     }
35     $this->success('查询成功', $menulist);
36 }
37 /**
38  * 删除网站留言
39  *
40  * @ApiMethod (POST)
41  * @ApiParams (name="ids", type="string", required=true, description="要删除的 ids")
42  */
43 public function del()
44 {
45     // $admin_id = $this->auth->id;
46     $ids = $this->request->post('ids/s');
47     try{
48         $menulist = $this->model->del($ids,true);
49     } catch (\Exception $e) {
50         $this->error($e->getMessage());
```

```
1     }
2     $this->success('删除成功', $menulist);
3 }
4 /**
5  * @ApiTitle(新增网站留言)
6  * @ApiSummary(新增网站留言)
7  * @ApiMethod(POST)
8  * @ApiParams(name = "name", type = "string",required=false,description = "提交者姓名")
9  * @ApiParams(name = "mobile", type = "string",required=true,description = "联系电话")
10 * @ApiParams(name = "emil", type = "string",required=true,description = "邮箱地址")
11 * @ApiParams(name = "message", type = "string",required=false,description = "留言内容")
12 * @ApiReturn({
13 *
14 *})
15 */
16 public function add()
17 {
18     $user_id = 0;
19     $user = $this->auth->getUser();//登录用户
20     if($user)$user_id = $user['id'];
21     $params=[];
22     $params["name"] = $this->request->post('name/s', ""); //页数
23     $params["mobile"] = $this->request->post('mobile/s', ""); //条数
24     $params["emil"] = $this->request->post('emil/s', ""); //搜索关键字
25     $params["message"] = $this->request->post('message/s', ""); //搜索关键字
26     try{
27         //当前申请状态
28         $res = $this->model->add($params,true);
29         // var_dump($this->model->getLastSql());
30     } catch (\Exception $e){
31         $this->error($e->getMessage());
32     }
33     $this->success('添加成功', $res);
34 }
35 /**
36 * @ApiTitle(编辑网站留言)
37 * @ApiSummary(编辑网站留言)
38 * @ApiMethod(POST)
39 * @ApiRoute (/adminapi/home.leave_word/edit/ids/{ids})
40 * @ApiParams (name="ids", type="string", required=true, description="需要编辑的 id")
41 * @ApiParams(name = "name", type = "string",required=false,description = "提交者姓名")
42 * @ApiParams(name = "mobile", type = "string",required=true,description = "联系电话")
43 * @ApiParams(name = "emil", type = "string",required=true,description = "邮箱地址")
44 * @ApiParams(name = "message", type = "string",required=false,description = "留言内容")
45 * @ApiReturn({
46 *
47 *})
48 */
49 public function edit($ids = null){
50 //     $user_id = 73;
```

```
1 // $user = $this->auth->getUser();//登录用户
2 // if($user)$user_id = $user['id'];
3 $params = [];
4 // $params["user_id"] = $user_id; //老师 id
5 $params["name"] = $this->request->post('name/s', ""); //页数
6 $params["mobile"] = $this->request->post('mobile/s', ""); //条数
7 $params["emil"] = $this->request->post('emil/s', ""); //搜索关键字
8 $params["message"] = $this->request->post('message/s', ""); //搜索关键字
9 try{
10     $res = $this->model->edit($ids,$params,true);
11 } catch (\Throwable $e){
12     $this->error($e->getMessage());
13 }
14 $this->success('编辑成功', $res);
15 }
16 }<?php
17 namespace app\adminapi\controller\home;
18 use app\common\controller\AdminApi;
19 use app\adminapi\model\home\News as NewsModel;
20 /**
21  * 新闻管理
22  */
23 class News extends AdminApi
24 {
25     protected $model = null;
26     /**
27      * 初始化操作
28      * @access protected
29      */
30     public function _initialize()
31     {
32         $this->model = new NewsModel;
33         parent::_initialize();
34     }
35     /**
36      * @ApiTitle( 新闻列表)
37      * @ApiSummary(新闻列表)
38      * @ApiMethod(GET)
39      * @ApiParams(name = "keywords", type = "string",required=false,description = "搜索关键字
40 ")
41      * @ApiParams(name = "page", type = "string",required=true,description = "页数")
42      * @ApiParams(name = "limit", type = "string",required=true,description = "条数")
43      * @ApiParams(name = "recommend", type = "string",required=false,description = "置顶:0=
44 否,1=是")
45      * @ApiParams(name = "cate_id", type = "int",required=false,description = "新闻类目 id")
46      * @ApiReturn({
47      *
48      *})
49      */
50     public function index()
```

```
1  {
2      $user_id = 0;
3      $user = $this->auth->getUser();//登录用户
4      if($user)$user_id = $user['id'];
5      $page = $this->request->get('page/d', 0); //页数
6      $limit = $this->request->get('limit/d', 0); //条数
7      $keywords = $this->request->get('keywords/s', ''); //搜索关键字
8      $recommend = $this->request->get('recommend/s', ''); //搜索关键字
9      // $params = [];
10     $cate_id = $this->request->get('cate_id/s', ''); //搜索关键字
11     // $type = $this->request->get('type/s', ''); //筛选学员和教练单
12     try{
13         //当前申请状态
14         $res = $this->model->allList($page, $limit,$keywords,$recommend,$cate_id,[]);
15     //     var_dump($this->model->getLastSql());
16     } catch (\Exception $e){
17         $this->error($e->getMessage());
18     }
19     $this->success('查询成功', $res);
20 }
21 /**
22  * 查看新闻详情
23  *
24  * @ApiMethod (GET)
25  * @ApiParams (name="id", type="string", required=true, description="ID")
26  */
27 public function detail()
28 {
29     $admin_id = $this->auth->id;
30     $id = $this->request->get('id/d');
31     try{
32         $menulist = $this->model->detail($id,$show_field=[],$except_field=[]);
33     } catch (\Exception $e) {
34         $this->error($e->getMessage());
35     }
36     $this->success('查询成功', $menulist);
37 }
38 /**
39  * 删除新闻
40  *
41  * @ApiMethod (POST)
42  * @ApiParams (name="ids", type="string", required=true, description="要删除的 ids")
43  */
44 public function del()
45 {
46     // $admin_id = $this->auth->id;
47     $ids = $this->request->post('ids/s');
48     try{
49         $menulist = $this->model->del($ids,true);
50     } catch (\Exception $e) {
```

```

1      $this->error($e->getMessage());
2    }
3    $this->success('删除成功', $menulist);
4  }
5  /**
6   * @ApiTitle(添加新闻)
7   * @ApiSummary(添加新闻)
8   * @ApiMethod(POST)
9   * @ApiParams(name = "cate_id", type = "string",required=true,description = "新闻类目")
10  * @ApiParams(name = "title", type = "int",required=true,description = "新闻标题")
11  * @ApiParams(name = "subtitle", type = "int",required=true,description = "新闻副标题")
12  * @ApiParams(name = "image", type = "int",required=true,description = "封面")
13  * @ApiParams(name = "content", type = "int",required=true,description = "内容")
14  * @ApiParams(name = "recommend", type = "string",required=true,description = "置顶:0=
15 否,1=是")
16  * @ApiParams(name = "release_time", type = "string",required=true,description = "发布时间:
17 例子 2025-05-09 17:30:46")
18  * @ApiParams(name = "weigh", type = "string",required=false,description = "权重")
19  * @ApiResponse({
20  *
21  *})
22  */
23  public function add(){
24  //    $user_id = 73;
25  //    $user = $this->auth->getUser();//登录用户
26  //    if($user)$user_id = $user['id'];
27    $params = [];
28  //    $params["user_id"] = $user_id; //老师 id
29    $params["cate_id"] = $this->request->post('cate_id/d', ""); //课程标签
30    $params["title"] = $this->request->post('title/s', ""); //课程标签
31    $params["subtitle"] = $this->request->post('subtitle/s', ""); //课程标签
32    $params["image"] = $this->request->post('image/s', ""); //老师 id
33    $params["content"] = $this->request->post('content/s', ""); //老师 id
34    $params["recommend"] = $this->request->post('recommend/s', ""); //老师 id
35    $params["release_time"] = $this->request->post('release_time/s', ""); //老师 id
36    $params["weigh"] = $this->request->post('weigh/d', ""); //老师 id
37    try{
38      $res = $this->model->add($params,true);
39    } catch (\Throwable $e){
40      $this->error($e->getMessage());
41    }
42    $this->success('添加成功', $res);
43  }
44  /**
45   * @ApiTitle(编辑新闻)
46   * @ApiSummary(编辑新闻)
47   * @ApiMethod(POST)
48   * @ApiRoute (/adminapi/home.news/edit/ids/{ids})
49   * @ApiParams (name="ids", type="string", required=true, description="需要编辑的 id")
50   * @ApiParams(name = "cate_id", type = "string",required=true,description = "新闻类目")

```

```

1      * @ApiParams(name = "title", type = "int",required=true,description = "新闻标题")
2      * @ApiParams(name = "subtitle", type = "int",required=true,description = "新闻副标题")
3      * @ApiParams(name = "image", type = "int",required=true,description = "封面")
4      * @ApiParams(name = "content", type = "int",required=true,description = "内容")
5      * @ApiParams(name = "recommend", type = "string",required=true,description = "置顶:0=
6 否,1=是")
7      * @ApiParams(name = "release_time", type = "string",required=true,description = "发布时间:
8 例子 2025-05-09 17:30:46")
9      * @ApiParams(name = "weigh", type = "string",required=false,description = "权重"))
10     * @ApiResponse({
11     *
12     *})
13     */
14     public function edit($ids = null){
15         // $user_id = 73;
16         // $user = $this->auth->getUser();//登录用户
17         // if($user)$user_id = $user['id'];
18         $params = [];
19         $params["cate_id"] = $this->request->post('cate_id/d', ""); //课程标签
20         $params["title"] = $this->request->post('title/s', ""); //课程标签
21         $params["subtitle"] = $this->request->post('subtitle/s', ""); //课程标签
22         $params["image"] = $this->request->post('image/s', ""); //老师 id
23         $params["content"] = $this->request->post('content/s', ""); //老师 id
24         $params["recommend"] = $this->request->post('recommend/s', ""); //老师 id
25         $params["release_time"] = $this->request->post('release_time/s', ""); //老师 id
26         $params["weigh"] = $this->request->post('weigh/d', ""); //老师 id
27         try{
28             $res = $this->model->edit($ids,$params,true);
29         } catch (\Throwable $e){
30             $this->error($e->getMessage());
31         }
32         $this->success('编辑成功', $res);
33     }
34 }<?php
35 namespace app\adminapi\controller\home;
36 use app\common\controller\AdminApi;
37 use app\adminapi\model\home\NewsCate as NewsCateModel;
38 /**
39  * 新闻分类管理
40  */
41 class NewsCate extends AdminApi
42 {
43     protected $model = null;
44     /**
45      * 初始化操作
46      * @access protected
47      */
48     public function _initialize()
49     {
50         $this->model = new NewsCateModel;

```

```

1     parent::_initialize();
2 }
3 /**
4  * @ApiTitle( 新闻分类列表)
5  * @ApiSummary(新闻分类列表)
6  * @ApiMethod(GET)
7  * @ApiParams(name = "keywords", type = "string",required=false,description = "搜索关键字
8  ")
9  * @ApiParams(name = "page", type = "string",required=true,description = "页数")
10 * @ApiParams(name = "limit", type = "string",required=true,description = "条数")
11 * @ApiParams(name = "status", type = "string",required=true,description = "状态:1=上架,2=
12 下架")
13 * @ApiParams(name = "hot", type = "string",required=false,description = "状态搜索条件: 0=
14 非热门分类,1=热门分类")
15 * @ApiReturn({
16 *
17 *})
18 */
19 public function index()
20 {
21     $user_id = 0;
22     $user = $this->auth->getUser();//登录用户
23     if($user)$user_id = $user['id'];
24     $page = $this->request->get('page/d', 0); //页数
25     $limit = $this->request->get('limit/d', 0); //条数
26     $params = [];
27     $params["keywords"] = $this->request->get('keywords/s', ""); //搜索关键字
28     $params["status"] = $this->request->get('status/s', ""); //搜索关键字
29     $params["hot"] = $this->request->get('hot/s', ""); //搜索关键字
30 //     $type = $this->request->get('type/s', ""); //筛选学员和教练单
31     try{
32         //当前申请状态
33         $res = $this->model::allList($page, $limit,$params);
34 //         if($user_id=='670153'){
35 //             file_put_contents("ceshi66.txt",(new AppointmentOrder()->getLastSql());
36 //         }
37     } catch (\Exception $e){
38         $this->error($e->getMessage());
39     }
40     $this->success('查询成功', $res);
41 }
42 /**
43  * 查看新闻分类详情
44  *
45  * @ApiMethod (GET)
46  * @ApiParams (name="id", type="string", required=true, description="ID")
47  */
48 public function detail()
49 {
50     $admin_id = $this->auth->id;

```

```
1      $id = $this->request->get('id/d');
2      try{
3          $menulist = $this->model->detail($id,$show_field=[],$except_field=[]);
4      } catch (\Exception $e) {
5          $this->error($e->getMessage());
6      }
7      $this->success('查询成功', $menulist);
8  }
9  /**
10   * 删除新闻分类
11   *
12   * @ApiMethod (POST)
13   * @ApiParams (name="ids", type="string", required=true, description="要删除的 ids")
14   */
15   public function del()
16   {
17       //      $admin_id = $this->auth->id;
18       $ids = $this->request->post('ids/s');
19       try{
20           $menulist = $this->model->del($ids,true);
21       } catch (\Exception $e) {
22           $this->error($e->getMessage());
23       }
24       $this->success('删除成功', $menulist);
25   }
26   /**
27   * @ApiTitle(添加分类)
28   * @ApiSummary(添加分类)
29   * @ApiMethod(POST)
30   * @ApiParams(name = "image", type = "string",required=true,description = "icon")
31   * @ApiParams(name = "name", type = "int",required=true,description = "类目名")
32   * @ApiParams(name = "status", type = "string",required=true,description = "状态:1=上架,2=
33   下架")
34   * @ApiParams(name = "hot", type = "string",required=true,description = "热门:0=否,1=是")
35   * @ApiParams(name = "weigh", type = "string",required=false,description = "权重")
36   * @ApiResponse({
37   *
38   *})
39   */
40   public function add(){
41       //      $user_id = 73;
42       //      $user = $this->auth->getUser();//登录用户
43       //      if($user)$user_id = $user['id'];
44       $params = [];
45       //      $params["user_id"] = $user_id; //老师 id
46       $params["image"] = $this->request->post('image/s, '); //课程标签
47       $params["name"] = $this->request->post('name/s, '); //课程标签
48       $params["status"] = $this->request->post('status/s, '); //课程标签
49       $params["hot"] = $this->request->post('hot/s, '); //老师 id
50       $params["weigh"] = $this->request->post('weigh/s, '); //老师 id
```

```

1      try{
2          $res = $this->model->add($params,true);
3      } catch (\Throwable $e){
4          $this->error($e->getMessage());
5      }
6      $this->success('添加成功', $res);
7  }
8  /**
9   * @ApiTitle(编辑分类)
10  * @ApiSummary(编辑分类)
11  * @ApiMethod(POST)
12  * @ApiRoute (/adminapi/home.news_cate/edit/ids/{ids})
13  * @ApiParams (name="ids", type="string", required=true, description="需要编辑的 id")
14  * @ApiParams(name = "image", type = "string",required=true,description = "icon")
15  * @ApiParams(name = "name", type = "int",required=true,description = "类目名")
16  * @ApiParams(name = "status", type = "string",required=true,description = "状态:1=上架,2=
17 下架")
18  * @ApiParams(name = "hot", type = "string",required=true,description = "热门:0=否,1=是")
19  * @ApiParams(name = "weigh", type = "string",required=false,description = "权重")
20  * @ApiReturn({
21  *
22  *})
23  */
24  public function edit($ids = null){
25      // $user_id = 73;
26      // $user = $this->auth->getUser();//登录用户
27      // if($user)$user_id = $user['id'];
28      $params = [];
29      // $params["user_id"] = $user_id; //老师 id
30      $params["image"] = $this->request->post('image/s', ""); //课程标签
31      $params["name"] = $this->request->post('name/s', ""); //课程标签
32      $params["status"] = $this->request->post('status/s', ""); //课程标签
33      $params["hot"] = $this->request->post('hot/s', ""); //老师 id
34      $params["weigh"] = $this->request->post('weigh/s', ""); //老师 id
35      try{
36          $res = $this->model->edit($ids,$params,true);
37      } catch (\Throwable $e){
38          $this->error($e->getMessage());
39      }
40      $this->success('编辑成功', $res);
41  }
42 }<?php
43 namespace app\adminapi\controller\home;
44 use app\common\controller\AdminApi;
45 use app\adminapi\model\home\RecruitmentInformation as RecruitmentInformationModel;
46 use app\adminapi\model\home\RecruitmentInformationCate;
47 /**
48  * 招聘信息管理
49  */
50 class RecruitmentInformation extends AdminApi

```

```
1  {
2      /**
3       * @var \app\adminapi\model\home\RecruitmentInformation
4       */
5      protected $model = null;
6      public function _initialize()
7      {
8          parent::_initialize();
9          $this->model = new RecruitmentInformationModel;
10     }
11     /**
12     * @ApiTitle( 招聘信息列表)
13     * @ApiSummary(招聘信息列表)
14     * @ApiMethod(GET)
15     * @ApiParams(name = "keywords", type = "string",required=false,description = "搜索关键字
16     ")
17     * @ApiParams(name = "page", type = "string",required=true,description = "页数")
18     * @ApiParams(name = "limit", type = "string",required=true,description = "条数")
19     * @ApiParams(name = "recommend", type = "string",required=false,description = "置顶:0=
20     否,1=是")
21     * @ApiParams(name = "cate_id", type = "int",required=false,description = "职位分类 id")
22     * @ApiResponse({
23     *
24     *})
25     */
26     public function index()
27     {
28         $user_id = 0;
29         $user = $this->auth->getUser();//登录用户
30         if($user)$user_id = $user['id'];
31         $page = $this->request->get('page/d', 0); //页数
32         $limit = $this->request->get('limit/d', 0); //条数
33         $keywords = $this->request->get('keywords/s', ""); //搜索关键字
34         $recommend = $this->request->get('recommend/s', ""); //搜索关键字
35         // $params=[];
36         $cate_id = $this->request->get('cate_id/s', ""); //搜索关键字
37         // $type = $this->request->get('type/s', ""); //筛选学员和教练单
38         try{
39             //当前申请状态
40             $res = $this->model->allList($page, $limit,$keywords,$recommend,$cate_id,[]);
41             // var_dump($this->model->getLastSql());
42         } catch (\Exception $e){
43             $this->error($e->getMessage());
44         }
45         $this->success('查询成功', $res);
46     }
47     /**
48     * 查看招聘信息详情
49     *
50     * @ApiMethod (GET)
```

```
1      * @ApiParams (name="id", type="string", required=true, description="ID")
2      */
3      public function detail()
4      {
5          $admin_id = $this->auth->id;
6          $id = $this->request->get('id/d');
7          try{
8              $menulist = $this->model->detail($id,$show_field=[],$except_field=[]);
9          } catch (\Exception $e) {
10              $this->error($e->getMessage());
11          }
12          $this->success('查询成功', $menulist);
13      }
14      /**
15       * 删除招聘信息
16       *
17       * @ApiMethod (POST)
18       * @ApiParams (name="ids", type="string", required=true, description="要删除的 ids")
19       */
20      public function del()
21      {
22          // $admin_id = $this->auth->id;
23          $ids = $this->request->post('ids/s');
24          try{
25              $menulist = $this->model->del($ids,true);
26          } catch (\Exception $e) {
27              $this->error($e->getMessage());
28          }
29          $this->success('删除成功', $menulist);
30      }
31      /**
32       * @ApiTitle(添加招聘信息)
33       * @ApiSummary(添加招聘信息)
34       * @ApiMethod(POST)
35       * @ApiParams(name = "name", type = "int",required=true,description = "职位名称")
36       * @ApiParams(name = "recommend", type = "string",required=false,description = "置顶:0=
37       否,1=是")
38       * @ApiParams(name = "cate_id", type = "int",required=false,description = "职位分类 id")
39       * @ApiParams(name = "weigh", type = "string",required=false,description = "权重")
40       * @ApiReturn({
41       *
42       *})
43       */
44      public function add(){
45          // $user_id = 73;
46          // $user = $this->auth->getUser();//登录用户
47          // if($user)$user_id = $user['id'];
48          $params = [];
49          // $params["user_id"] = $user_id; //老师 id
50          $params["name"] = $this->request->post('name/s, '); //课程标签
```

```
1      $params["recommend"] = $this->request->post('recommend/s', ''); //课程标签
2      $params["cate_id"] = $this->request->post('cate_id/s', ''); //老师 id
3      $params["weigh"] = $this->request->post('weigh/s', ''); //老师 id
4      try{
5          $res = $this->model->add($params,true);
6      } catch (\Throwable $e){
7          $this->error($e->getMessage());
8      }
9      $this->success('添加成功', $res);
10 }
11 /**
12  * @ApiTitle(编辑招聘信息)
13  * @ApiSummary(编辑招聘信息)
14  * @ApiMethod(POST)
15  * @ApiRoute (/adminapi/home.recruitment_information/edit/ids/{ids})
16  * @ApiParams (name="ids", type="string", required=true, description="需要编辑的 id")
17  * @ApiParams(name = "name", type = "int",required=true,description = "职位名称")
18  * @ApiParams(name = "recommend", type = "string",required=false,description = "置顶:0=
19 否,1=是")
20  * @ApiParams(name = "cate_id", type = "int",required=false,description = "职位分类 id")
21  * @ApiParams(name = "weigh", type = "string",required=false,description = "权重")
22  * @ApiReturn({
23  *
24  *})
25  */
26 public function edit($ids = null){
27     // $user_id = 73;
28     // $user = $this->auth->getUser();//登录用户
29     // if($user)$user_id = $user['id'];
30     $params = [];
31     // $params["user_id"] = $user_id; //老师 id
32     $params["name"] = $this->request->post('name/s', ''); //课程标签
33     $params["recommend"] = $this->request->post('recommend/s', ''); //课程标签
34     $params["cate_id"] = $this->request->post('cate_id/s', ''); //老师 id
35     $params["weigh"] = $this->request->post('weigh/s', ''); //老师 id
36     try{
37         $res = $this->model->edit($ids,$params,true);
38     } catch (\Throwable $e){
39         $this->error($e->getMessage());
40     }
41     $this->success('编辑成功', $res);
42 }
43 }<?php
44 namespace app\adminapi\controller\home;
45 use app\common\controller\AdminApi;
46 use app\adminapi\model\home\RecruitmentInformationCate as RecruitmentInformationCateMode
47 l;
48 /**
49  * 招聘分类管理
50  */
```

```
1 class RecruitmentInformationCate extends AdminApi
2 {
3     protected $model = null;
4     /**
5      * 初始化操作
6      * @access protected
7      */
8     public function _initialize()
9     {
10         $this->model = new RecruitmentInformationCateModel;
11         parent::_initialize();
12     }
13     /**
14      * @ApiTitle( 招聘分类列表)
15      * @ApiSummary(招聘分类列表)
16      * @ApiMethod(GET)
17      * @ApiParams(name = "keywords", type = "string",required=false,description = "搜索关键字
18 ")
19      * @ApiParams(name = "page", type = "string",required=true,description = "页数")
20      * @ApiParams(name = "limit", type = "string",required=true,description = "条数")
21      * @ApiParams(name = "status", type = "string",required=true,description = "状态:1=上架,2=
22 下架")
23      * @ApiReturn({
24      *
25      *})
26      */
27     public function index()
28     {
29         $user_id = 0;
30         $user = $this->auth->getUser();//登录用户
31         if($user)$user_id = $user['id'];
32         $page = $this->request->get('page/d', 0); //页数
33         $limit = $this->request->get('limit/d', 0); //条数
34         $params = [];
35         $params["keywords"] = $this->request->get('keywords/s', ""); //搜索关键字
36         $params["status"] = $this->request->get('status/s', ""); //搜索关键字
37         // $type = $this->request->get('type/s', ""); //筛选学员和教练单
38         try{
39             //当前申请状态
40             $res = $this->model::allList($page, $limit,$params);
41             // if($user_id=='670153'){
42             //     file_put_contents("ceshi66.txt",(new AppointmentOrder()->getLastSql());
43             // }
44         }catch (\Exception $e){
45             $this->error($e->getMessage());
46         }
47         $this->success('查询成功', $res);
48     }
49     /**
50      * 查看招聘分类详情
```

```
1      *
2      * @ApiMethod (GET)
3      * @ApiParams (name="id", type="string", required=true, description="ID")
4      */
5      public function detail()
6      {
7          $admin_id = $this->auth->id;
8          $id = $this->request->get('id/d');
9          try{
10             $menulist = $this->model->detail($id,$show_field=[],$except_field=[]);
11         } catch (\Exception $e) {
12             $this->error($e->getMessage());
13         }
14         $this->success('查询成功', $menulist);
15     }
16     /**
17     * 删除招聘分类
18     *
19     * @ApiMethod (POST)
20     * @ApiParams (name="ids", type="string", required=true, description="要删除的 ids")
21     */
22     public function del()
23     {
24         // $admin_id = $this->auth->id;
25         $ids = $this->request->post('ids/s');
26         try{
27             $menulist = $this->model->del($ids,true);
28         } catch (\Exception $e) {
29             $this->error($e->getMessage());
30         }
31         $this->success('删除成功', $menulist);
32     }
33     /**
34     * @ApiTitle(添加分类)
35     * @ApiSummary(添加分类)
36     * @ApiMethod(POST)
37     * @ApiParams(name = "name", type = "int",required=true,description = "类目名")
38     * @ApiParams(name = "status", type = "string",required=true,description = "状态:1=上架,2=
39     下架")
40     * @ApiParams(name = "weigh", type = "string",required=false,description = "权重")
41     * @ApiResponse( {
42     *
43     * })
44     */
45     public function add(){
46         // $user_id = 73;
47         // $user = $this->auth->getUser();//登录用户
48         // if($user)$user_id = $user['id'];
49         $params = [];
50         // $params["user_id"] = $user_id; //老师 id
```

```
1      $params["name"] = $this->request->post('name/s', ''); //课程标签
2      $params["status"] = $this->request->post('status/s', ''); //课程标签
3      $params["weigh"] = $this->request->post('weigh/s', ''); //老师 id
4      try{
5          $res = $this->model->add($params,true);
6      } catch (\Throwable $e){
7          $this->error($e->getMessage());
8      }
9      $this->success('添加成功', $res);
10 }
11 /**
12  * @ApiTitle(编辑分类)
13  * @ApiSummary(编辑分类)
14  * @ApiMethod(POST)
15  * @ApiRoute (/adminapi/home.recruitment_information_cate/edit/ids/{ids})
16  * @ApiParams (name="ids", type="string", required=true, description="需要编辑的 id")
17  * @ApiParams(name = "name", type = "int",required=true,description = "类目名")
18  * @ApiParams(name = "status", type = "string",required=true,description = "状态:1=上架,2=
19 下架")
20  * @ApiParams(name = "weigh", type = "string",required=false,description = "权重")
21  * @ApiReturn({
22  *
23  *})
24  */
25 public function edit($ids = null){
26     // $user_id = 73;
27     // $user = $this->auth->getUser();//登录用户
28     // if($user)$user_id = $user['id'];
29     $params = [];
30     // $params["user_id"] = $user_id; //老师 id
31     $params["name"] = $this->request->post('name/s', ''); //课程标签
32     $params["status"] = $this->request->post('status/s', ''); //课程标签
33     $params["weigh"] = $this->request->post('weigh/s', ''); //老师 id
34     try{
35         $res = $this->model->edit($ids,$params,true);
36     } catch (\Throwable $e){
37         $this->error($e->getMessage());
38     }
39     $this->success('编辑成功', $res);
40 }
41 }
42 <?php
43 namespace app\adminapi\model\home;
44 use app\common\model\BaseModel;
45 use think\Model;
46 use traits\model\SoftDelete;
47 class Images extends BaseModel
48 {
49     use SoftDelete;
50     // 表名
```

```
1   protected $name = 'home_images';
2   // 自动写入时间戳字段
3   protected $autoWriteTimestamp = 'integer';
4   // 定义时间戳字段名
5   protected $createTime = 'createtime';
6   protected $updateTime = false;
7   protected $deleteTime = 'deletetime';
8   // 追加属性
9   protected $append = [
10      'type_text',
11      'showtype_text'
12  ];
13  protected static function init()
14  {
15      self::afterInsert(function ($row) {
16          if (!$row['weigh']) {
17              $pk = $row->getPk();
18              $row->getQuery()->where($pk, $row[$pk])->update(['weigh' => $row[$pk]]);
19          }
20      });
21  }
22  public function getTypeList()
23  {
24      return ['in' => __('Type in'), 'out' => __('Type out')];
25  }
26  public function getShowtypeList()
27  {
28      return ['image' => __('Showtype image'), 'video' => __('Showtype video')];
29  }
30  public function getTypeTextAttr($value, $data)
31  {
32      $value = $value ?: ($data['type'] ?? "");
33      $list = $this->getTypeList();
34      return $list[$value] ?? "";
35  }
36  public function getShowtypeTextAttr($value, $data)
37  {
38      $value = $value ?: ($data['showtype'] ?? "");
39      $list = $this->getShowtypeList();
40      return $list[$value] ?? "";
41  }
42  public function getImageAttr($value, $data)
43  {
44      if (!empty($value)) return cdnurl($value, true);
45  }
46  public static function allList($page, $limit, $params=[]){
47      $sort = "weigh desc,id desc";
48      // if(!empty($params['status'])){
49      //     $params['status'] = '1';
50      // }
```

```

1      return (new self)->getBaseList($params, $page, $limit,$sort);
2  }
3  /**得到基础条件
4   * @param $status
5   * @param null $model
6   * @param string $alisa
7   */
8  public static function getBaseWhere($whereData = [], $model = null, $alisa = "", $with = false)
9  {
10     if (!$model) {
11         $model = new static;
12         if ($alisa && !$with) $model = $model->alias($alisa);
13     }
14     if ($alisa) $alisa = $alisa . '.';
15     $tableFields = (new static)->getTableFields();
16     foreach ($tableFields as $fields)
17     {
18         if(in_array($fields, ['type','showtype']))continue;
19         // if (isset($whereData[$fields]) && $whereData[$fields]) $model = $model->where("{ $alisa
20 sa} { $fields} ", '=', $whereData[$fields]);
21         if (isset($whereData[$fields]) && $whereData[$fields]){
22             if(is_array($whereData[$fields])){
23                 $model = $model->where("{ $alisa} { $fields} ", $whereData[$fields][0], $whereData[$
24 fields][1]);
25             }else{
26                 $model = $model->where("{ $alisa} { $fields} ", '=', $whereData[$fields]);
27             }
28         }
29     }
30     if (isset($whereData['type']) && $whereData['type']) $model = $model->where("{ $alisa} typ
31 e", 'in', $whereData['type']);
32     if (isset($whereData['showtype']) && $whereData['showtype']) $model = $model->where("{
33 $alisa} showtype", 'in', $whereData['showtype']);
34     if (isset($whereData['keywords']) && $whereData['keywords']) $model = $model->where("{ $
35 alisa} id|{ $alisa} url", 'Like', "%{$whereData['keywords']}%");
36     return $model;
37 }
38 /** 通用详情(后台 api 版本)
39 * @param $params
40 * @param $trans
41 * @return $this
42 * @throws \Exception
43 */
44 public function detail($id,$show_field=[],$except_field=[]){
45     $row = $this->get($id);
46     if (!$row) {
47         throw new \Exception(__('No Results were found'));
48     }
49     if($show_field){
50         $row->visible($show_field);

```

```
1     }
2     if($except_field){
3         $row->hidden($except_field);
4     }
5     return $row;
6 }
7 /** 通用删除(后台 api 版本)
8  * @param $params
9  * @param $trans
10  * @return $this
11  * @throws \Exception
12  */
13 public function del($ids = null,$trans=false){
14     if (empty($ids)) {
15         throw new \Exception(__('Parameter %s can not be empty', 'ids'));
16     }
17     //判断逻辑
18     $pk = $this->getPk();
19     $adminIds = $this->getDataLimitAdminIds();
20     if (is_array($adminIds)) {
21         $this->where($this->dataLimitField, 'in', $adminIds);
22     }
23     $list = $this->where($pk, 'in', $ids)->select();
24     $count = 0;
25     if($trans){
26         self::beginTrans();
27     }
28     $res = true;
29     try{
30         foreach ($list as $item) {
31             $count += $item->delete();
32         }
33         if($trans){
34             self::commitTrans();
35         }
36     }catch (\Exception $e){
37         if($trans){
38             self::rollbackTrans();
39         }
40         throw new \Exception($e->getMessage().$e->getFile().$e->getLine());
41     }
42     return $count;
43 }
44 /** 通用新增(后台 api 版本)
45  * @param $params
46  * @param $trans
47  * @return $this
48  * @throws \Exception
49  */
50 public function add($params,$trans=false){
```

```
1      if (empty($params)) {
2          throw new \Exception(__('Parameter %s can not be empty', ''));
3      }
4      if ($this->dataLimit && $this->dataLimitFieldAutoFill) {
5          $params[$this->dataLimitField] = $this->auth->id;
6      }
7      //判断逻辑
8      if($trans){
9          self::beginTrans();
10     }
11     $res = true;
12     try{
13         //是否采用模型验证
14         if ($this->modelValidate) {
15             $name = str_replace("\\model\\", "\\validate\\", get_class($this));
16             $validate = is_bool($this->modelValidate) ? ($this->modelSceneValidate ? $name . '.add
17 ' : $name) : $this->modelValidate;
18             $this->validateFailException()->validate($validate);
19         }
20         $result = $this->allowField(true)->save($params);
21         if($trans){
22             self::commitTrans();
23         }
24     } catch (\Exception $e){
25         if($trans){
26             self::rollbackTrans();
27         }
28         throw new \Exception($e->getMessage().$e->getFile().$e->getLine());
29     }
30     return $this;
31 }
32 /** 通用编辑(后台 api 版本)
33  * @param $params
34  * @param $trans
35  * @return $this
36  * @throws \Exception
37  */
38 public function edit($id,$params,$trans=false){
39     $row = $this->get($id);
40     if (!$row) {
41         throw new \Exception(__('No Results were found'));
42     }
43     $adminIds = $this->getDataLimitAdminIds();
44     if (is_array($adminIds) && !in_array($row[$this->dataLimitField], $adminIds)) {
45         throw new \Exception(__('You have no permission'));
46     }
47     if (empty($params)) {
48         throw new \Exception(__('Parameter %s can not be empty', ''));
49     }
50     //判断逻辑
```

```
1      if($trans){
2          self::beginTrans();
3      }
4      $res = true;
5      try{
6          //是否采用模型验证
7          if ($this->modelValidate) {
8              $name = str_replace("\\model\\", "\\validate\\", get_class($this));
9              $validate = is_bool($this->modelValidate) ? ($this->modelSceneValidate ? $name . '.edit
10 ': $name) : $this->modelValidate;
11              $row->validateFailException()->validate($validate);
12          }
13          $result = $row->allowField(true)->save($params);
14          if($trans){
15              self::commitTrans();
16          }
17      } catch (\Exception $e){
18          if($trans){
19              self::rollbackTrans();
20          }
21          throw new \Exception($e->getMessage().$e->getFile().$e->getLine());
22      }
23      return $row;
24  }
25 }
26 <?php
27 namespace app\adminapi\model\home;
28 use app\common\model\BaseModel;
29 use think\Model;
30 class LeaveWord extends BaseModel
31 {
32     // 表名
33     protected $name = 'leave_word';
34     // 自动写入时间戳字段
35     protected $autoWriteTimestamp = 'integer';
36     // 定义时间戳字段名
37     protected $createTime = 'createtime';
38     protected $updateTime = 'updatetime';
39     protected $deleteTime = false;
40     // 追加属性
41     protected $append = [
42         'status_text'
43     ];
44     public function getStatusList()
45     {
46         return ['1' => __('Status 1'), '2' => __('Status 2')];
47     }
48     public function getStatusTextAttr($value, $data)
49     {
50         $value = $value ?: ($data['status'] ?? "");
```

```

1      $list = $this->getStatusList();
2      return $list[$value] ?? "";
3  }
4  public static function allList($page, $limit,$params=[]){
5      $sort = "id desc";
6      //      if(!empty($params['status'])){
7      //          $params['status'] = '1';
8      //      }
9      return (new self)->getBaseList($params, $page, $limit,$sort);
10 }
11 /**得到基础条件
12  * @param $status
13  * @param null $model
14  * @param string $alisa
15  */
16 public static function getBaseWhere($whereData = [], $model = null, $alisa = "", $with = false)
17 {
18     if (!$model) {
19         $model = new static;
20         if ($alisa&&!$with) $model = $model->alias($alisa);
21     }
22     if ($alisa) $alisa = $alisa . '.';
23     $tableFields = (new static)->getTableFields();
24     foreach ($tableFields as $fields)
25     {
26         if(in_array($fields, ['status']))continue;
27         //      if (isset($whereData[$fields]) && $whereData[$fields]) $model = $model->where("{ $ali
28 sa} { $fields} ", '=', $whereData[$fields]);
29         if (isset($whereData[$fields]) && $whereData[$fields]){
30             if(is_array($whereData[$fields])){
31                 $model = $model->where("{ $alisa} { $fields} ", $whereData[$fields][0], $whereData[
32 fields][1]);
33             }else{
34                 $model = $model->where("{ $alisa} { $fields} ", '=', $whereData[$fields]);
35             }
36         }
37     }
38     if (isset($whereData['status']) && $whereData['status']) $model = $model->where("{ $alisa} s
39 tatus", 'in', $whereData['status']);
40     if (isset($whereData['keywords'])&&$whereData['keywords']) $model = $model->where("{ $
41 alisa} name|{ $alisa} mobile|{ $alisa} message|{ $alisa} email", 'Like', "%{$whereData['keywords']}%"
42 ");
43     return $model;
44 }
45 /** 通用详情(后台 api 版本)
46  * @param $params
47  * @param $trans
48  * @return $this
49  * @throws \Exception
50  */

```

```
1 public function detail($id,$show_field=[],$except_field=[]){
2     $row = $this->get($id);
3     if (!$row) {
4         throw new \Exception(__('No Results were found'));
5     }
6     if($show_field){
7         $row->visible($show_field);
8     }
9     if($except_field){
10        $row->hidden($except_field);
11    }
12    return $row;
13 }
14 /** 通用删除(后台 api 版本)
15  * @param $params
16  * @param $trans
17  * @return $this
18  * @throws \Exception
19  */
20 public function del($ids = null,$trans=false){
21     if (empty($ids)) {
22         throw new \Exception(__('Parameter %s can not be empty', 'ids'));
23     }
24     //判断逻辑
25     $pk = $this->getPk();
26     $adminIds = $this->getDataLimitAdminIds();
27     if (is_array($adminIds)) {
28         $this->where($this->dataLimitField, 'in', $adminIds);
29     }
30     $list = $this->where($pk, 'in', $ids)->select();
31     $count = 0;
32     if($trans){
33         self::beginTrans();
34     }
35     $res = true;
36     try{
37         foreach ($list as $item) {
38             $count += $item->delete();
39         }
40         if($trans){
41             self::commitTrans();
42         }
43     } catch (\Exception $e){
44         if($trans){
45             self::rollbackTrans();
46         }
47         throw new \Exception($e->getMessage().$e->getFile().$e->getLine());
48     }
49     return $count;
50 }
```

```
1  /** 通用新增(后台 api 版本)
2  * @param $params
3  * @param $trans
4  * @return $this
5  * @throws \Exception
6  */
7  public function add($params,$trans=false){
8      if (empty($params)) {
9          throw new \Exception(__('Parameter %s can not be empty', ));
10     }
11     // if ($this->dataLimit && $this->dataLimitFieldAutoFill) {
12     //     $params[$this->dataLimitField] = $this->auth->id;
13     // }
14     //判断逻辑
15     if($trans){
16         self::beginTrans();
17     }
18     $res = true;
19     try{
20         $params["status"] = '1';
21         //是否采用模型验证
22         if ($this->modelValidate) {
23             $name = str_replace("\\model\\", "\\validate\\", get_class($this));
24             $validate = is_bool($this->modelValidate) ? ($this->modelSceneValidate ? $name . '.add
25 ': $name) : $this->modelValidate;
26             $this->validateFailException()->validate($validate);
27         }
28         $result = $this->allowField(true)->save($params);
29         if($trans){
30             self::commitTrans();
31         }
32     } catch (\Exception $e){
33         if($trans){
34             self::rollbackTrans();
35         }
36         throw new \Exception($e->getMessage().$e->getFile().$e->getLine());
37     }
38     return $this;
39 }
40 /** 通用编辑(后台 api 版本)
41 * @param $params
42 * @param $trans
43 * @return $this
44 * @throws \Exception
45 */
46 public function edit($id,$params,$trans=false){
47     $row = $this->get($id);
48     if (!$row) {
49         throw new \Exception(__('No Results were found'));
50     }
```

```
1      $adminIds = $this->getDataLimitAdminIds();
2      if (is_array($adminIds) && !in_array($row[$this->dataLimitField], $adminIds)) {
3          throw new \Exception(__('You have no permission'));
4      }
5      if (empty($params)) {
6          throw new \Exception(__('Parameter %s can not be empty', ''));
7      }
8      //判断逻辑
9      if($trans){
10         self::beginTrans();
11     }
12     $res = true;
13     try{
14         //是否采用模型验证
15         if ($this->modelValidate) {
16             $name = str_replace("\\model\\", "\\validate\\", get_class($this));
17             $validate = is_bool($this->modelValidate) ? ($this->modelSceneValidate ? $name . '.edit
18 ': $name) : $this->modelValidate;
19             $row->validateFailException()->validate($validate);
20         }
21         $result = $row->allowField(true)->save($params);
22         if($trans){
23             self::commitTrans();
24         }
25     } catch (\Exception $e){
26         if($trans){
27             self::rollbackTrans();
28         }
29         throw new \Exception($e->getMessage().$e->getFile().$e->getLine());
30     }
31     return $row;
32 }
33 }
34 <?php
35 namespace app\adminapi\model\home;
36 use app\common\model\BaseModel;
37 use think\Model;
38 use traits\model\SoftDelete;
39 class News extends BaseModel
40 {
41     use SoftDelete;
42     // 表名
43     protected $name = 'news';
44     // 自动写入时间戳字段
45     protected $autoWriteTimestamp = 'integer';
46     // 定义时间戳字段名
47     protected $createTime = 'createtime';
48     protected $updateTime = 'updatetime';
49     protected $deleteTime = 'deletetime';
50     // 追加属性
```

```
1   protected $append = [
2       'recommend_text',
3       'release_time_text'
4   ];
5   protected static function init()
6   {
7       self::afterInsert(function ($row) {
8           if (!$row['weigh']) {
9               $pk = $row->getPk();
10              $row->getQuery()->where($pk, $row[$pk])->update(['weigh' => $row[$pk]]);
11          }
12      });
13  }
14  public function getRecommendList()
15  {
16      return ['0' => __('Recommend 0'), '1' => __('Recommend 1')];
17  }
18  public function getRecommendTextAttr($value, $data)
19  {
20      $value = $value ?: ($data['recommend'] ?? '');
21      $list = $this->getRecommendList();
22      return $list[$value] ?? '';
23  }
24  public function getReleaseTimeTextAttr($value, $data)
25  {
26      $value = $value ?: ($data['release_time'] ?? '');
27      return is_numeric($value) ? date("Y-m-d H:i:s", $value) : $value;
28  }
29  protected function setReleaseTimeAttr($value)
30  {
31      return $value === '' ? null : ($value && !is_numeric($value) ? strtotime($value) : $value);
32  }
33  public function getImageAttr($value, $data)
34  {
35      if (!empty($value)) return cdnurl($value, true);
36  }
37  public function cate()
38  {
39      return $this->belongsTo(NewsCate::class, 'cate_id', 'id', [], 'LEFT')->setEagerlyType(0);
40  }
41  /**得到基础条件
42   * @param $status
43   * @param null $model
44   * @param string $alisa
45   */
46  public static function getBaseWhere($whereData = [], $model = null, $alisa = '', $with = false)
47  {
48      if (!$model) {
49          $model = new static;
50          if ($alisa && !$with) $model = $model->alias($alisa);
```

```

1      }
2      if ($alisa) $alisa = $alisa . '.';
3      $tableFields = (new static)->getTableFields();
4      foreach ($tableFields as $fields)
5      {
6          if(in_array($fields, ['cate_id','recommend']))continue;
7          //      if (isset($whereData[$fields]) && $whereData[$fields]) $model = $model->where("{ $alisa
8          sa} {$fields}", '=', $whereData[$fields]);
9          if (isset($whereData[$fields]) && $whereData[$fields]){
10             if(is_array($whereData[$fields])){
11                 $model = $model->where("{ $alisa} {$fields}", $whereData[$fields][0], $whereData[$
12                 fields][1]);
13             }else{
14                 $model = $model->where("{ $alisa} {$fields}", '=', $whereData[$fields]);
15             }
16         }
17     }
18     if (isset($whereData['keywords'])&&$whereData['keywords']){
19         $model = $model->where("{ $alisa} title{ $alisa} subtitle", 'LIKE', "%{$whereData['keywor
20         ds']}%" );
21     }
22     if (isset($whereData['recommend']) && $whereData['recommend']!="" && $whereData['re
23     commend']!=null){
24         $model = $model->where("{ $alisa} recommend", '=', $whereData['recommend'] );
25     }
26     if (isset($whereData['cate_id'])&&$whereData['cate_id']){
27         $model = $model->where("{ $alisa} cate_id", 'in', $whereData['cate_id'] );
28     }
29     if (isset($whereData['time'])&&$whereData['time']){
30         $model = $model->time(["{ $alisa} release_time",$whereData['time']);
31     }
32     //      if (isset($whereData['has_evaluate'])&&$whereData['has_evaluate']){
33     //          //1 查已评价 2 查未评价
34     //          if($whereData['has_evaluate'] == 1){
35     //              //1 查已评价
36     //              $model = $model->where("{ $alisa} classes_evaluate_id", '<>', 0);
37     //          }else{
38     //              //2 查未评价
39     //              $model = $model->whereExists(function ($query) use ($alisa) {
40     //                  $order_table_name = (new \app\common\model\school\classes\hourorder\Order())->
41     //                  getQuery()->getTable();
42     //                  $query->table($order_table_name)->where($order_table_name . '.classes_order_id='
43     //                  . $alisa . 'id')->where('status', '=', '3');
44     //                  }->where("{ $alisa} classes_evaluate_id", '=', 0);
45     //              }
46     //          }
47     //      }
48     return $model;
49 }
50 public static function allList($page, $limit,$keywords,$recommend="", $cate_id=[], $params=[])

```