

Router link under the hood renders anchor tag. Jevabe onnanno component page e router link set korte hoy,

```
<router-link to="/teams">Teams</router-link>
```

----- Router link Active class -----

```
a: hover,  
a: active,  
a.router-link-active {  
  color: #f1a80a;  
  border-color: #f1a80a;  
  background-color: #1a037e;  
}
```

**router-link-active** hocche vue js er router link er default class, uporer css ta add korle routerlink active dekhay page visit korle. Eta hocche parent theke child router porjonto jay.

**router-link-exact-active** hocche only jei route ase accurate parent e thake.

----- watch properties for routing -----

```
created() {  
  this.loadmembers(this.$route);  
},  
  
watch: {  
  $route(newRoute) {  
    this.loadmembers(newRoute);  
  },  
},
```

Ekhon methods er moddhe jevabe route paremeter ke pass kore,

```
loadmembers(route) {  
  const teamId = route.params.id;  
},
```

### ----- redirect routing -----

From main.js, `/:pathMatch(.*)` bolte bojhano hocche jekono kichu likhle direct oi path kei fetch korbe.

```
{ path: '/:pathMatch(.*)', redirect: '/teams' },
```

### ----- nested route -----

From main.js,

```
{
  path: '/teams',
  component: TeamsList,
  children: [
    {
      path: ':id',
      component: TeamMembers,
      props: true,
    },
  ],
},
```

Ekhon jei parent component er nested route tar mathay ekta `<router-view>` diye rakhte hobe, for an example **TeamList** is the parent component.

```
<template>
  <ul>
    <router-view :key="Math.random()"></router-view>
    <teams-item
      v-for="team in teams"
      :key="team.id"
      :name="team.name"
      :member-count="team.members.length"
      :id="team.id"
    ></teams-item>
  </ul>
</template>
```

----- naming route and fetching with parameter -----

From main.js,

```
{
  path: '/teams/:id',
  component: TeamMembers,
  props: true,
  name: 'teams',
},
```

Now from TeamsItem.vue,

```
<router-link :to="{ name: 'teams', params: { id: 't1' } }"
  >View Members</router-link
>
```

----- vue js query search -----

Query jevabe add kore,

```
<router-link
  :to="{
    name: 'teams',
    params: { id: 't1' },
    query: { search: 'example' },
  }"
  >View Members</router-link
>
```

Query view korar niom,

```
const search = this.$route.query.search
```

//----- route with computed property -----

```
computed: {
  teamMembersLink() {
    return {
      name: 'teams',
      params: { id: this.id },
      query: { sort: 'asc' },
    };
  },
},
```

Then using it via router-view,

```
<router-link :to="teamMembersLink">View Members</router-link>
```

Scroll behavior vue,

```
scrollBehavior(to, from, savedPosition) {  
  if (savedPosition) {  
    // if savedPosition is available, return to that position  
    return savedPosition;  
  } else {  
    // scroll to the top of the page  
    return { top: 0 };  
  }  
}
```

//----- Before entering to route example //-----

```
{  
  path: '/teams',  
  component: TeamsList,  
  
  beforeEnter: (to, from, next) => {  
    // Perform some action before entering the route  
    console.log('Entering this route');  
    next();  
  },  
},
```

//----- Before router Leave example //-----

```
beforeRouteLeave(to, from, next) {  
  console.log('UsersList Cmp beforeRouteLeave') ;  
  console.log(to, from);  
  
  if(this.changes){  
    next();  
  }  
  else {  
    const userWantsToLeave = confirm('Are you sure? You got unsaved changes!');  
    next (userWantsToLeave) ;  
  }  
}
```