

Basic Routing Setup

src/App.vue

src/main.js

```
<template>
  <nav>
    <RouterLink to="/">Home</RouterLink>
    <RouterLink to="/product-list">Product List</RouterLink>
  </nav>
  <RouterView />
</template>
```

use RouterLink and RouterView

```
import router from './router'

const app = createApp(App)
app.use(router)
```

add router to the app

src/router/index.js

```
const router = createRouter({
  ...
  routes: [
    {
      path: '/',
      name: 'home',
      component: HomeView
    },
    {
      path: '/product-list',
      name: 'product-list',
      component: ProductListView
    }
  ]
})
```

set up the routes

src/views/HomeView.vue

```
<template>
  <h1>Home Page</h1>
  ...
</template>
```

src/views/ProductListView.vue

```
<template>
  <h1>Product List Page</h1>
  ...
</template>
```

Dynamic Routing

src/views/ProductListView.vue

src/views/ProductView.vue

```
<template>
  <h1>Product List Page</h1>
  <ul>
    <li>
      <RouterLink to="/product/1">Computer</RouterLink>
    </li>
    <li>
      <RouterLink to="/product/2">Laptop</RouterLink>
    </li>
    ...
  </ul>
</template>
```

```
<script>
export default {
  data() {
    return {
      product: {}
    }
  },
  created() {
    const id = parseInt(this.$route.params.id)
    this.product = products.find(id)
  }
}
</script>
```

get the param with Options API

src/router/index.js

```
const router = createRouter({
  ...
  routes: [
    ...
    {
      path: '/product/:id',
      name: 'product',
      component: ProductView
    }
  ]
})
```

set up the param with a colon prefix

```
<script setup>
import { useRoute } from 'vue-router'

const route = useRoute()
const id = parseInt(route.params.id)
const product = products.find(id)
</script>
```

```
<template>
  <main>
    <h1>{{ product.name }}</h1>
  </main>
</template>
```

get the param with Composition API



Nested Routing

src/router/index.js

```
const router = createRouter({
  ...
  routes: [
    {
      path: '/',
      ...
    },
    {
      path: '/contact',
      ...
    },
    {
      path: '/product/:id',
      name: 'product',
      component: ProductView,
      children: [
        {
          path: 'reviews',
          component: ProductReviews
        },
        {
          path: 'variations',
          component: ProductVariations
        }
      ]
    }
  ]
})
```

product/1

product/1/reviews

product/1/variations

set up the **nested routes** with the **children** option

src/views/ProductView.vue

```
<template>
  <h1>{{ product.name }}</h1>
  ...
  <RouterView></RouterView>
</template>
```

put another **RouterView** here

src/components/ProductReviews.vue

```
<template>
  <h2>Reviews</h2>
  ...
</template>
```

src/components/ProductVariations.vue

```
<template>
  <h2>Variations</h2>
  ...
</template>
```

Go to a page using script

```
<script setup>
import { useRouter } from 'vue-router'

const router = useRouter()

const goToProductList = () => {
  router.push('/product-list')
}
</script>
```

with **Composition API**

```
<script>
export default {
  methods: {
    goToProductList() {
      this.$router.push('/product-list')
    }
  }
}
</script>
```

with **Options API**

Redirect to somewhere else

```
const router = createRouter({
  ...
  routes: [
    {
      path: '/',
      name: 'home',
      redirect: to => ({ path: '/product-list' })
    },
    {
      path: '/product-list',
      name: 'product-list',
      component: ProductListView
    }
  ]
})
```

The **Home** page route will be redirected to the **Product List** page.

