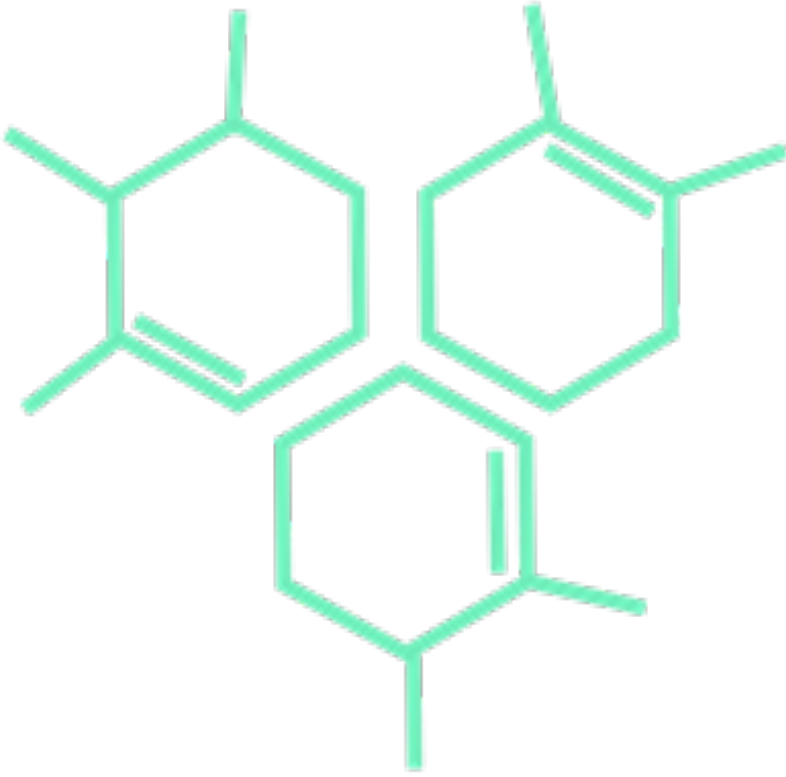
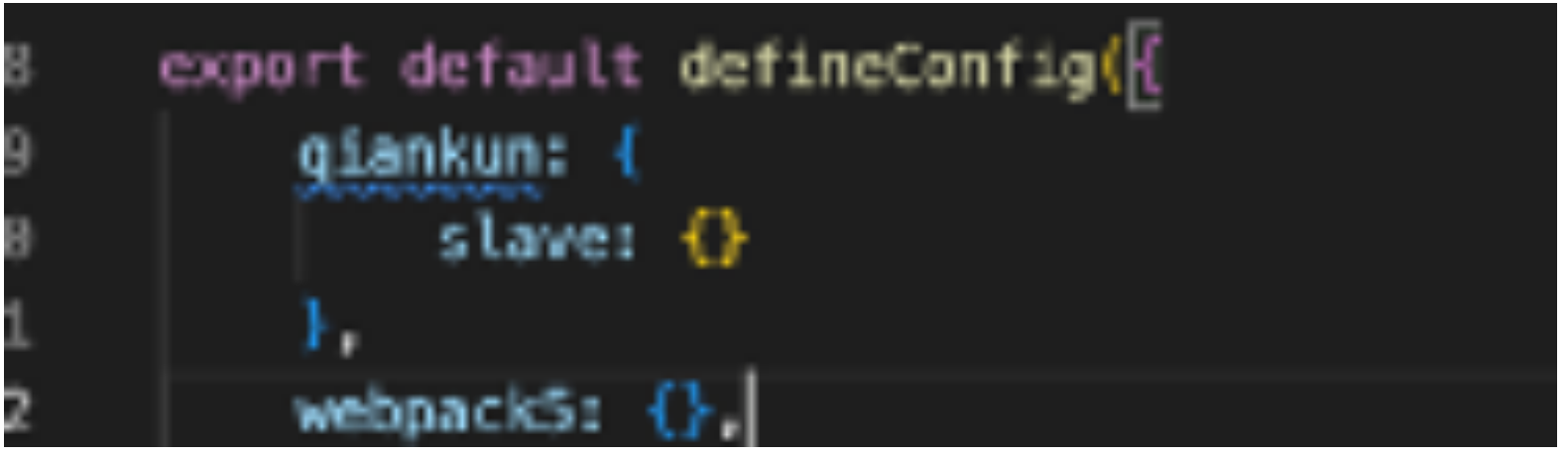


Step.1

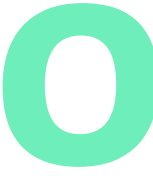
Step.2

Step.3

```
    <Route path={ROUTES.                                    }
element={<MicroApp />}></Route>
```


export default defineConfig({
qiankun: {
slave: {}
},
webpack5: {},

e



Horizons Tech Session

```
import { loadMicroApp } from 'qiankun';
export default class MicroApp extends React.Component {
  containerRef: any = React.createRef();
  microApp: any = null;
    componentDidMount() {
    this.microApp = loadMicroApp(
      {
        ...appConfig,
        props: {
          accessToken: tokenStorage.getAccessToken(),
          refreshToken: tokenStorage.getRefreshToken(),
        },
        container: this.containerRef.current,
      },
      /*other configs*/
    );
    this.microApp.mountPromise.then(() => {
      this.setState({
        loading: false,
      });
      if (this.microApp.getStatus() !== MICRO_STATUS.MOUNTED) {
        throw new CatchAbleError('micro load failed');
      }
    });
  }
}
```

Implementation of qiankun

Implementation of qiankun

Joinv2 add qiankun@2.8.4 dependency
Define MicroApp Component

Step.1

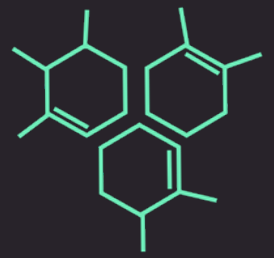
Joinv2 add micro front-end route paths
Eg:<Route path={ROUTES.EXPENSESAPPROVAL}
element={<MicroApp />}></Route>

Step.2

Joinv1 add umi qiankun config, closing chunks config, adding name filed in package.json

```
export default defineConfig({
    qiankun: {
        slave: {}
    },
    webpack5: {},
```

Step.3



	Module Federation	Qiankun
Availability	Y	Y
Modifications	import { useDispatch, useSelector } from 'umi' Where umi store management is used, it needs to be replaced	Add logical processing according to the operating environment Add route prefix
Time/days	3	2
Risks	Amount of alteration	Uncertain stability