

```
import { createSlice, createAsyncThunk } from "@reduxjs/toolkit";
import BaoCaoService from "services/sale/BaoCaoService";
```

```
export const getBaoCaoTonKho = createAsyncThunk(
  "BaoCao/getBaoCaoTonKho",
  async (data, { rejectWithValue }) => {
    try {
      const response = await BaoCaoService.getBaoCaoTonKho(data);
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);
```

```
export const getBangKeNhap = createAsyncThunk(
  "BaoCao/getBangKeNhap",
  async (data, { rejectWithValue }) => {
    try {
      const response = await BaoCaoService.getBangKeNhap(data);
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);
```

```
export const getBangKeXuat = createAsyncThunk(
  "BaoCao/getBangKeXuat",
  async (data, { rejectWithValue }) => {
    try {
      const response = await BaoCaoService.getBangKeXuat(data);
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);
```

```
export const getBCDoanhThu = createAsyncThunk(
  "BaoCao/getBCDoanhThu",
  async (data, { rejectWithValue }) => {
    try {
      const response = await BaoCaoService.getBCDoanhThu(data);
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);
```

```
const initialState = {
  loading: false,
  BaoCaotonList: [],
  BangKeNhapList: [],
  BangKeXuatList: [],
  BaoCaoDoanhThuList: [],
};
```

```
export const BaoCaoSlice = createSlice({
  name: "BaoCao",
  initialState,
  reducers: {
    showLoading: (state) => {
      state.loading = true;
    },
    setAllDmkkList: (state, action) => {
      state.allDmKkList = action.payload;
    },
  },
  extraReducers: (builder) => {
```

```
builder
  .addCase(getBaoCaoTonKho.fulfilled, (state, action) => {
    state.loading = false;
    state.BaoCaotonList = action.payload;
  })
  .addCase(getBangKeNhap.fulfilled, (state, action) => {
    state.loading = false;
    state.BangKeNhapList = action.payload;
  })
  .addCase(getBangKeXuat.fulfilled, (state, action) => {
    state.loading = false;
    state.BangKeXuatList = action.payload;
  })
  .addCase(getBCDoanhThu.fulfilled, (state, action) => {
    state.loading = false;
    state.BaoCaoDoanhThuList = action.payload;
  });
},
});
```

```
export const { showLoading } = BaoCaoSlice.actions;
```

```
export default BaoCaoSlice.reducer;
```