

```

import React, { useEffect, useRef, useState } from "react";

import { useDispatch, useSelector } from "react-redux";

import { SystemButtonSource } from "constants";
//import moment from "moment";
import MeditechTablePage, { FilterComponent } from "components/table-layout";
import { Tabs } from "antd";

import {
  ExportBaoCao,
  GetBaoCaoCongNo,
  GetBaoCaoDoanhThu,
} from "store/slices/khamSuckHoeDoanSlice";
import DMBaoCaoCongNo from "../DSBaoCaoCongNo";
import DSBaoCaoDoanhThu from "../DSBaoCaoDoanhThu";
//import DSBaoCaoDoanhThu from "../DSBaoCaoDoanhThu";

const initSearch = {
  startDate: null, //moment(moment().now).month(-6)
  endDate: null, //moment(moment().now)
  searchText: null,
};

const BaoCaoCongNo = ({ queryId, openModalDetail }) => {
  const { branchId } = useSelector((state) => state.auth);
  const dispatch = useDispatch();
  const { loading, BaoCaoCongNoList, BaoCaoDoanhThuList } = useSelector(
    (state) => state.ksk
  );
  const [tabIndex, setTabIndex] = useState(1);
  // eslint-disable-next-line react-hooks/exhaustive-deps
  const reloadData = () => {
    searchFormRef.current = {
      ...searchFormRef.current,
      branchId: branchId,
    };
    dispatch(GetBaoCaoCongNo(searchFormRef.current));
    dispatch(GetBaoCaoDoanhThu(searchFormRef.current));
  };

  useEffect(() => {
    reloadData();
    // eslint-disable-next-line react-hooks/exhaustive-deps
  }, [dispatch, branchId]);

  const searchFormRef = useRef({ ...initSearch });
  const [searchFormData, setSearchFormData] = useState(searchFormRef.current);
  const searchHandle = (dataSearch) => {
    searchFormRef.current = { ...searchFormRef.current, ...dataSearch };
    setSearchFormData({ ...searchFormRef.current });
    reloadData();
  };

  const onRenderSearchView = () => {
    return (
      <FilterComponent
        datasource={searchFormData}
        onSearchImmediate={searchHandle}
        // renderFilterMoreView={renderFilterMoreView}
        renderInputSearch
        renderDatePicker
      >
      {
        /* <Select
          key="congTyId"
          data-field="congTyId"
          style={{ width: "100%", minWidth: 200 }}
          placeholder="Công ty"
          options={CongTyList}
          allowClear
          mode="multiple"
        > */
      }
    );
  };

```

```

    </FilterComponent>
  );
};

const onChangeTab = (e) => {
  setTabIndex(e);
};

const tabList = [
  {
    key: 1,
    label: "Báo cáo doanh thu",
    children: (
      <DSBaoCaoDoanhThu
        BaoCaoDoanhThuList={BaoCaoDoanhThuList}
        loading={loading}
      />
    ),
  },
  {
    key: 2,
    label: "Báo cáo công nợ",
    children: (
      <DMBaoCaoCongNo BaoCaoCongNoList={BaoCaoCongNoList} loading={loading} />
    ),
  },
];

const getRightButton = (isLocked) => {
  const arr = [];
  if (isLocked) {
    arr.push({
      keyName: SystemButtonSource.downloadBtn,
      title: "Xuất Excel",
      showMore: false,
      // disable: !allowAddNew,
    });
  }
  return arr;
};

const onRightTopBtnHandler = (key) => {
  switch (key) {
    case SystemButtonSource.downloadBtn:
      searchFormRef.current = { ...searchFormRef.current, loi: tabIndex };
      setSearchFormData({ ...searchFormRef.current });
      dispatch(
        ExportBaoCao({
          ...searchFormRef.current,
          onSuccess: (data) => {
            const link = document.createElement("a");
            const linkUrl = data.data.replace("http:", "https:");
            link.href = linkUrl;
            link.click();
          },
        })
      );
      break;
    default:
      break;
  }
};

return (
  <>
    <MeditechTablePage
      tableTitle="BÁO CÁO CÔNG NỢ"
      btnHeaderRightSource={getRightButton(true)}
      btnHeaderRightSourceHandle={onRightTopBtnHandler}
      onRenderSearchView={onRenderSearchView}
    >
      { /* <DMBaoCaoCongNo BaoCaoCongNoList={BaoCaoCongNoList} loading={loading} /> */ }
      <Tabs
        items={tabList}
        destroyInactiveTabPane

```

```
        onChange={(e) => onChangeTab(e)}
      />
    </MeditechTablePage>
  </>
);
};
export default BaoCaoCongNo;
```