Pagination

1. Import pagination under List component.

import Pagination from "@mui/material/Pagination";

2) Import stack

import Stack from "@mui/material/Stack";

3) add stack and pagination after Grid in list component.

 <div style={{ position: "relative", top: "30px" ,display:"flex",justifyContent:"center" }}>

        <Stack spacing={2} xs={12}>

          <Pagination

            style={{fontWeight:"200px"}}

            count={data.length / 10}

            page={page}

            onChange={handleChange}

            siblingCount={4}

            color="secondary"

          />

        </Stack>

      </div>

4) Add three const under List.

a) next count: how much do you want to show on the page put here example(10)

b)Previous Count: it should be zero.

c) Page: page is dominated for page count and from which page do you want show your page number so from page one we want to show the page one (1)

const [nextCount, setNextCount] = useState(10);

  const [prevCount, setPrevCount] = useState(0);

  const [page, setPage] = useState(1);

5) Add slice method under map.

<Grid container spacing={2}>

        {data.length > 0 && data.slice(prevCount, nextCount).map((item) =>

         <TodoItem item={item} />)}

6) Add onchange handle. refer screenshot of point Three for moredetails.

            onChange={handleChange}

7) Write onchange function and its condition.

const handleChange = (event, value) => {

    setNextCount(10\* value); //10 is for how many item do you want to show on the page.// (page count is Value) example 10\*2 =20 on second page it will show 11 to 20 item//

    setPrevCount(10\* value - 10);

    setPage(value);//Page count//

  };

Find all codes below under TODO list tab,

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

import { TodoItem } from "./TodoItem";

import { Grid } from "@mui/material";

import axios from "axios";

import Pagination from "@mui/material/Pagination";

import Stack from "@mui/material/Stack";

export const TodoList = () => {

  const [data, setData] = useState([]);

  const [nextCount, setNextCount] = useState(10);

  const [prevCount, setPrevCount] = useState(0);

  const [page, setPage] = useState(1);

  const getData = async () => {

    const result = await axios.get(

      "https://jsonplaceholder.typicode.com/todos"

    );

    setData(result.data);

  };

  useEffect(() => {

    getData();

  }, []);

  const handleChange = (event, value) => {

    setNextCount(10\* value);

    setPrevCount(10\* value - 10);

    setPage(value);

  };

  return (

    <div>

      <Grid container spacing={2}>

        {data.length > 0 && data.slice(prevCount, nextCount).map((item) =>

         <TodoItem item={item} />)}

      </Grid>

      <div style={{ position: "relative", top: "30px" ,display:"flex",justifyContent:"center" }}>

        <Stack spacing={2} xs={12}>

          <Pagination

            style={{fontWeight:"200px"}}

            count={data.length / 10}

            page={page}

            onChange={handleChange}

            siblingCount={4}

            color="secondary"

          />

        </Stack>

      </div>

    </div>

  );

};