

Implementing Column Reordering in FrameDataGrid

Overview

This guide walks through implementing column reordering (shuffle) in the FrameDataGrid component of a Next.js app using TanStack React Table, Tailwind CSS, and ShadCN UI. It also includes optional features like resetting to default order and persisting user preferences in localStorage.

- Add Column Order State

```
const [columnOrder, setColumnOrder] = React.useState<string[]>(
  columns.map((col) => col.id!)
);
```

- Enable Column Ordering in useReactTable

```
const table = useReactTable({
  data,
  columns,
  state: { columnFilters, sorting, columnOrder },
  onColumnFiltersChange: setColumnFilters,
  onSortingChange: setSorting,
  onColumnOrderChange: setColumnOrder,
  getCoreRowModel: getCoreRowModel(),
  getPaginationRowModel: getPaginationRowModel(),
  manualPagination: true,
  manualFiltering: true,
  manualSorting: true,
  enableColumnOrdering: true,
});
```

- Update Table Header to Allow Drag-Drop

```
<th
  draggable={!frozenColumnIds.includes(columnId)}

  onStartDrag={e => e.dataTransfer.setData("text/plain", columnId)}

  onDragOver={e => !frozenColumnIds.includes(columnId) && e.preventDefault()}

  onDrop={e => {
    const sourceId = e.dataTransfer.getData("text/plain");

    if (!frozenColumnIds.includes(sourceId) && sourceId !== columnId) {

      const newOrder = [...columnOrder];

      newOrder.splice(newOrder.indexOf(sourceId), 1);

      newOrder.splice(newOrder.indexOf(columnId), 0, sourceId);

      setColumnOrder(newOrder);

    }
  }}
>...</th>
```

- Add Reset Order Button

```
<Button onClick={() => setColumnOrder(columns.map((col) => col.id!))}>

  Reset Order

</Button>
```

- Optional: Persist to localStorage

```
const initialOrder = () => {

  const saved = localStorage.getItem("frameDataGridColumnOrder");

  const defaultOrder = columns.map((col) => col.id!);

  if (saved) {

    const parsed = JSON.parse(saved);

    if (Array.isArray(parsed) && parsed.every((id) => defaultOrder.includes(id))) {
```

```

    return parsed;
  }
}

return defaultOrder;
};

const [columnOrder, setColumnOrder] = useState<string[]>(initialOrder);

useEffect(() => {

  localStorage.setItem("frameDataGridColumnOrder", JSON.stringify(columnOrder));

}, [columnOrder]);

```

- Reset with Storage Clear

```

<Button onClick={() => {

  localStorage.removeItem("frameDataGridColumnOrder");

  setColumnOrder(columns.map((col) => col.id!));

}}>

  Reset Order

</Button>

```

Summary

- Column reorder via drag-and-drop for non-frozen columns
- Frozen columns remain locked in place
- 'Reset Order' restores original sequence
- Optional localStorage persistence of column order
- Compatible with existing sort/filter/freeze logic