

# How to get React Table Row Data Onclick

Asked 2 years, 10 months ago   Modified 1 year, 1 month ago   Viewed 33k times



8



2



Hi I am trying to set up my react app such that when you click on a button in a row item in my react-table the data in that row is passed onto another component. At the moment I am simply trying to console.log the correct data but am unsure of how to pass react-table row data based on click. How can I do this? Thanks

My Dummy data stored in state along with the button (Show Detailed View) which I want to trigger the data passing onclick:

```
columns: [
  {
    Header: "First Name",
    accessor: "fname"
  },
  {
    Header: "Last Name",
    accessor: "lname"
  },
  {
    Header: "Employee Group",
    accessor: "egroup"
  },
  {
    Header: "Date of Birth",
    accessor: "dob"
  },
  {
    Header: "",
    id: "id",
    Cell: ({ row }) => (
      <button onClick={e => this.handleShow()}>
        Detailed View
      </button>
    )
  },
],
posts: [
  {
    fname: "gerald",
    lname: "nakhle",
    egroup: "faisbuk",
    dob: "8/10/1995"
  }
]
```

My Call to render the table:

```
<ReactTable columns={this.state.columns} data={this.state.posts}></ReactTable>
```

My onclick handler function but im not sure how I can access the table row data I'm after

```
handleShow(e) {
  console.log(e);
}
```

reactjs onclick react-props react-table

Share Improve this question

edited May 11, 2019 at 1:43

asked May 11, 2019 at 1:09

Follow



Tamjid

1,995

3

11

29

Sorted by:

Highest score (default)



## 4 Answers



You will need to add an onClick handler for the row

4



```
const onClick = (state, rowInfo, column, instance) => {
  return {
    onClick: e => {
      console.log('A Td Element was clicked!')
      console.log('it produced this event:', e)
      console.log('It was in this column:', column)
      console.log('It was in this row:', rowInfo)
      console.log('It was in this table instance:', instance)
    }
  }
}
```

```
<ReactTable columns={this.state.columns} data={this.state.posts} getTrProps=
{onClick}></ReactTable>
```

check this post for more info [react-table component onClick event for a column](#)

Share Improve this answer Follow

answered May 11, 2019 at 4:11



Khalid

227

1

5



For React-table v7:

2

Pass a callback prop `onRowClicked` to the table component,



In your table component call the callback:



```
...row.getRowProps({
  onClick: e => props.onRowClicked && props.onRowClicked(row, e),
})
```

In react-table v7, all the spread operator on HTML element that starts with `get...Props` are [props getter](#) for example:

row.getRowProps(), cell.getCellProps(), column.getHeaderProps(),  
getTableBodyProps(), getTableProps() etc.. You can pass any more attributes to extend  
it. for example:

```
...cell.getCellProps({
  style: {color: 'red'},
  onClick: ()=> {}
})
```

Share Improve this answer Follow

answered Feb 1, 2021 at 10:02



Harish Kulkarni

914 7 15

- 1 this is very helpful as I was adding onClick in the td tag directly and not passing onClick as a prop to row.getRowProps function which was causing unwanted behaviour in the table. – Nabeel Hussain Shah May 1, 2021 at 20:35

In your Table definition :

0

```
export function TableCustom({handleShow}) {
  const columns = React.useMemo(() => [{
    Header: 'Action',
    accessor: 'action',
    Cell: props => <button className="btn1" onClick={() =>
handleShow(props)}>Details</button>,
  },]

  return <ReactTable>;
});
```

And in your parent component : view the data of the clicked row :

```
const handleShow = (cell) => {
  console.log(cell?.row?.original);
}
```

Share Improve this answer Follow

answered Feb 12, 2021 at 13:48



pushStack

1,517 1 10 14

Also make sure you are binding your function in your constructor, or use the following syntax:

-7

```
handleShow = (e) => {
  console.log(e);
}
```

Share Improve this answer Follow

answered Feb 26, 2020 at 19:56