React 5: Lists & Keys

IN608: Intermediate Application Development Concepts

Kaiako: Tom Clark & Grayson Orr

Last Session’s Content

- File structure
- Events
- Conditional rendering

Today’s Content

- Lists & keys

Lists & Keys

Lists & Keys

- Consider the following function component:

```
const NumberList = (props) => {
  const numbers = props.numbers
  const listItems = numbers.map((number) => <li>{number}</li>)
  return <ul>{listItems}</ul>
}
```

- What is happening?
  - Loop through nums using the `map()` function
  - Returns an `<li>` element for each item
  - Assign the resulting array of elements to `listItems`
  - Return a `<ul>` element containing `listItems`

Lists & Keys

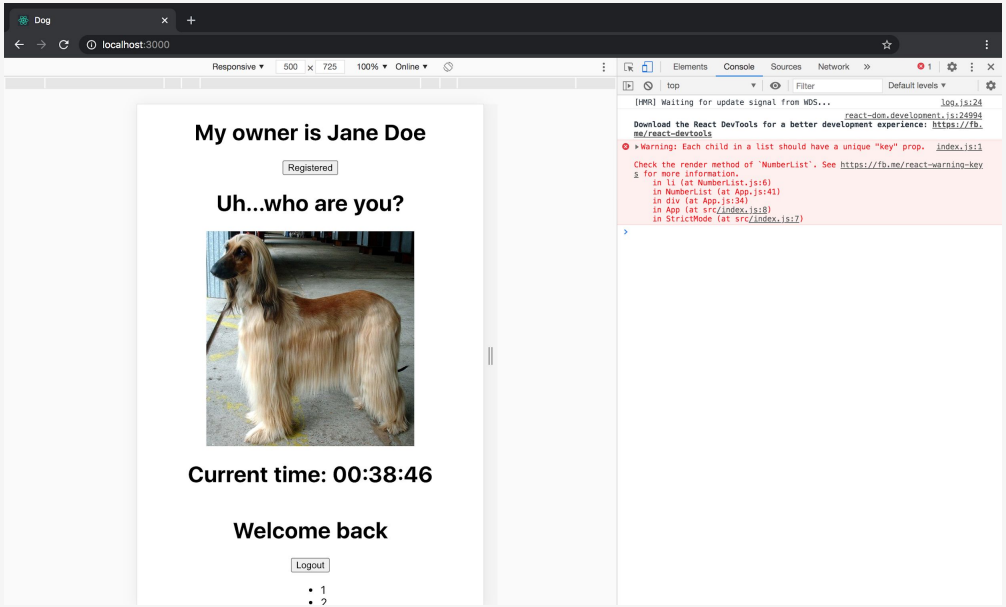
- In `App.js`

```
import React from 'react'
...
import NumberList from './NumberList'

const App = () => {
  ...
  const numbers = [1, 2, 3, 4, 5]
  return (
    <div className='main-container'>
      ...
      <NumberList numbers={numbers} />
    </div>
  )
}

export default App
```

- What happens when you run this code?
  - You will be given a warning that a key should be provided for the list of elements
- What is a key?
  - They help React identify which items have changed, are added or are removed
  - Given to the elements inside the array for stable identity

Lists & Keys

- Open Developer Tools in Chrome or Firefox
- How do we fix this?



Lists & Keys

- Use a string that uniquely identifies a list item among its siblings
- In most cases, you would use an ID from your data as key

```
const NumberList = (props) => {
  const numbers = props.numbers
  const listItems = numbers.map((number) => (
    <li key={number.toString()}>{number}</li>
  ))
  return <ul>{listItems}</ul>
}
```

- If you do not have a stable ID for a rendered item, you may use the item's index as a key

```
const NumberList = (props) => {
  const numbers = props.numbers
  const listItems = numbers.map((number, index) => (
    <li key={index}>{number}</li>
  ))
  return <ul>{listItems}</ul>
}
```

- An index is not recommended if an item's order changes
- Causes negative impact on performance & may cause issues with a component's state

Lists & Keys

- How do we extract components with keys?
  - Two separate function components
  - `ListItem` - the key is not specified in the `<li>`
  - `NumberList` - the key is specified in the array

```
// ListItem.js
const ListItem = (props) => <li>{props.value}</li>

// NumberList.js
const NumberList = (props) => {
  const numbers = props.numbers
  const listItems = numbers.map((number) => (
    <ListItem key={number.toString()} value={number} />
  ))
  return <ul>{listItems}</ul>
}
```

Lists & Keys

- Keys used within arrays should be unique among siblings
- Though, they do not need to be globally unique
- The same keys can be used to produce two different arrays

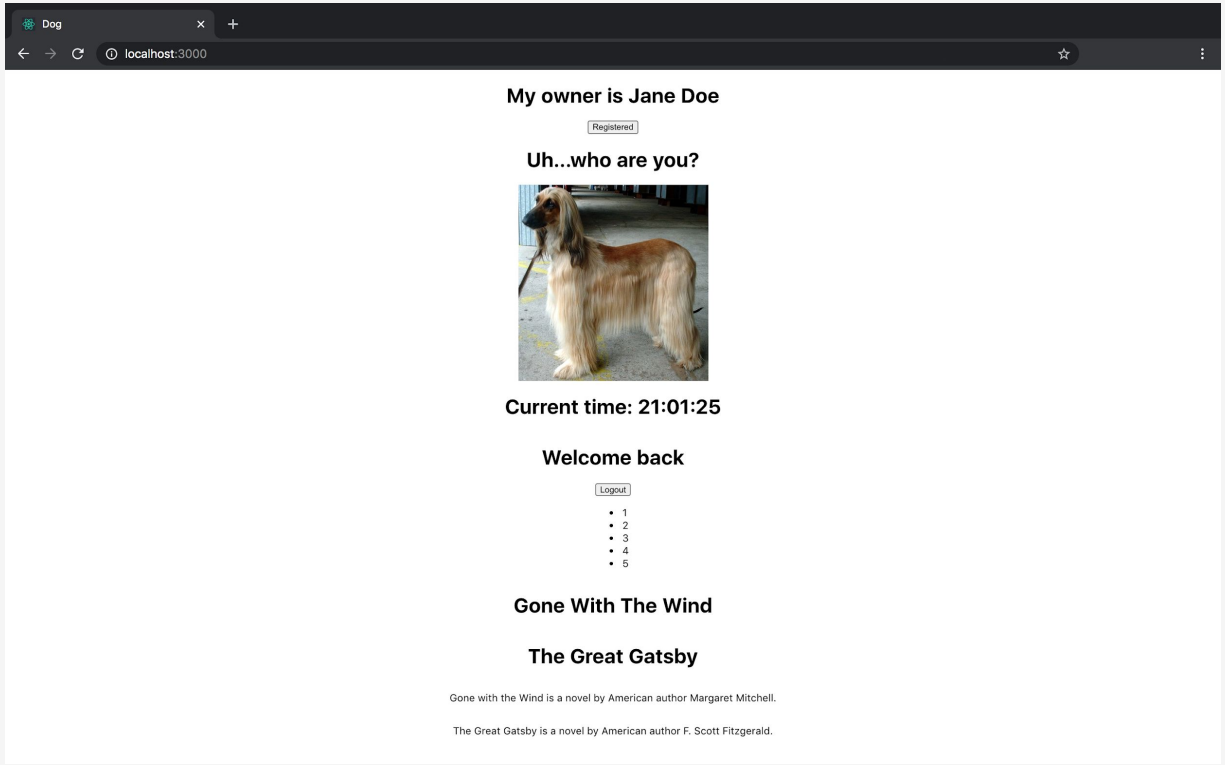
```
// Book.js
const Book = (props) => {
  const title = props.books.map((book) => (
    <div key={book.id}>
      <h1>{book.title}</h1>
    </div>
  ))

  const content = props.books.map((book) => (
    <div key={book.id}>
      <p>{book.content}</p>
    </div>
  ))

  return (
    <React.Fragment>
      {title}
      {content}
    </React.Fragment>
  )
}
```

Lists & Keys

- In `App.js`

```
import React from 'react'
...
import Book from './Book'

const App = () => {
  ...
  const books = [
    { id: 1, title: 'Gone With The Wind', content: 'Gone with the Wind is a novel by American author Margaret Mitchell.' },
    { id: 2, title: 'The Great Gatsby', content: 'The Great Gatsby is a novel by American author F. Scott Fitzgerald.' },
  ]
  return (
    <div className='main-container'>
      ...
      <Book books={books} />
    </div>
  )
}

export default App
```

Lists & Keys


Dog
localhost:3000
My owner is Jane Doe
Registered
Uh...who are you?
Current time: 21:01:25
Welcome back
Logout
1
2
3
4
5
Gone With The Wind
The Great Gatsby
Gone with the Wind is a novel by American author Margaret Mitchell.
The Great Gatsby is a novel by American author F. Scott Fitzgerald.

Programming Activity

- Checkout to master - `git checkout master`
- Create a new branch called 20-practical - `git checkout -b 20-practical`
- Open the file `20-practical.pdf` and work on the tasks described