01.12.2023

A Card Component That Has Multiple "Slots"
https://github.com/academind/react-typescript-course-resources/blob/main/examples/Card.tsx

A reusable `Card` component that does not just accept the `children` prop but also accepts an `actions` prop which receives JSX as value.

It may be used like this:

```
<Card
  title='My Card'
  actions={<button onClick={() => console.log('Button clicked!')}>Click Me!</button>}>
  <p>Some content</p>
</Card>
```

A Button Component That Accepts Text & An Icon
https://github.com/academind/react-typescript-course-resources/blob/main/examples/IconButton.tsx

A reusable `IconButton` component which will output text side-by-side next to a configurable icon (or actually any component). It accepts an `icon` prop which wants a component identifier as a value.

It may be used like this:

```
<IconButton
  icon={HeartIcon}
  onClick={() => console.log('Button clicked!')}>
  Like
</IconButton>
```

01.12.2023

A Generic List Component

A reusable `List` component which outputs an `<ul>` that wraps a list that may output any kind of value (strings, numbers, objects, ...). The component accepts the `items` as a prop as well as a `renderItem` prop that expects to get a function that controls how each item should be rendered.

It may be used like this:

```
<main>
  <section>
    <h2>Users</h2>
    <List
      items={users}
      renderItem={(user) => <li key={user.id}>{user.name}</li>}
    />
  </section>
  <section>
    <h2>Hobbies</h2>
    <List
      items={hobbies}
      renderItem={(hobby) => <li key={hobby}>{hobby}</li>}
    />
  </section>
</main>
```