

3 Zobrazování seznamů

Ukážeme si, jak v React komponentách zobrazovat seznamy.

[◀ Předchozí](#)
Podmíněné zobrazení

Následující [▶](#)
Události, stav

Naše cesta Reactem pokračuje k další z velmi důležitých dovedností. Téměř každá webová aplikace zobrazuje nějaký seznam, ať už to je nákupní košík, seznam divadelních představení, jídelní lístek nebo třeba seznam nepřečtených e-mailů. V této lekci si ukážeme, jak v Reactu zařídit zobrazení seznamu uloženého v nějakém JavaScriptovém poli.

Opakování metody map

Na zobrazování seznamů budeme v Reactu používat metodu `map`, kterou již znáte z kapitoly o pokročilém JavaScriptu. Jistě se nám vyplatí si tuto metodu připomenout a zopakovat.

Metoda `map` slouží k tomu, abychom z jednoho JavaScriptového pole vyrobili pole jiné pomocí nějaké transformační funkce. Takto například vyrobíme e-maily pro seznam uživatelů.

```
const names = ['petr', 'jana', 'marek', 'eva', 'lenka', 'ondra'];
const emails = names.map((name) => {
  return `${name}@mejlik.cz`;
});
```

Jistě si také vzpomenete, že jsme při zápisu takovýchto transformací používali zkrácený zápis arrow funkcí.

```
const names = ['petr', 'jana', 'marek', 'eva', 'lenka', 'ondra'];
const emails = names.map((name) => `${name}@mejlik.cz`);
```

Zkracování funkcí nám však pomůže i v psaní reactových komponent.

Zkrácený zápis delších funkcí

Při práci v Reactu budeme často mít komponentu, která nedělá nic jiného, než že vrací nějaké JSX. Takové JSX se nám většinou nevejde na jeden řádek. Přesto můžeme funkci s komponentu šikovně zkrátit.

Mejmě následující komponentu.

```
const Time = (props) => {  
  const { hours, minutes } = props;  
  return (  
    <div className="time">  
      <span className="time__hours">{hours}</span>  
      :  
      <span className="time__mins">{minutes}</span>  
    </div>  
  );  
}
```

Tuto komponentu bychom mohli převést na zkrácený zápis, ale vadí nám tam řádek s destrukurováním. Destrukturování se však v JavaScriptu dá přesunout přímo do parametru funkce.

```
const Time = ({ hours, minutes }) => {  
  return (  
    <div className="time">  
      <span className="time__hours">{hours}</span>  
      :  
      <span className="time__mins">{minutes}</span>  
    </div>  
  );  
}
```

Takovýto zápis je pro nás novinka, ale v praxi se běžně používá, takže je dobré jej dostat pod kůži. Nyní už komponenta nedělá nic jiného, než že vrací JSX element. Můžeme ji proto převést na zkrácený zápis. Sluší se ale v zápise ponechat kulaté závorky pro lepší čitelnost.

```
const Time = ({ hours, minutes }) => (  
  <div className="time">  
    <span className="time__hours">{hours}</span>  
    :  
    <span className="time__mins">{minutes}</span>  
  </div>  
);
```

Pokud je komponenta malá, klidně se bez závorek obejdeme.

```
const User = ({ login }) => <div className="user__login">{login}</div>;
```

Všimněte si, že pomocí takovéto komponenty bychom například mohli vyrobit pole JSX elementů z našeho pole uživatelů.

```
const names = ['petr', 'jana', 'marek', 'eva', 'lenka', 'ondra'];  
const users = names.map((name) => <User login={name} />);
```

Z tohoto bodu zbývá už jen malý krůček, abychom takovéto pole JSX komponent dokázali zobrazit na naší stránce. To si však necháme až na druhou část lekce.

Následující >
Cvičení: Zkracování a map

Kódím.cz

Verze 2.0.0-beta.7