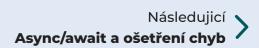


/ Webové aplikace / Digitální akademie Web / React / Opakování: LeviExpress, druhá část / Plánek sedadel

## 10 Opakování: LeviExpress, druhá část

Dokončíme základní verzi LeviExpressu a troufneme si i na rozšířenou funkčnost.





**2** Řady sedadel

Nyní zařídíme zobrazování sedadel v řadách. Plánek sedadel bude vypadat tak, že v HTML bude pro každou řadu sedadel jedna komponenta SeatRow a teprve uvnitř této komponenty budou jednotlivá sedadla – komponenty Seat .

## Vyberte sedadlo



1. V projektu vytvořte komponentu SeatRow, která představuje jednu řadu sedadel. Bude vracet div s třídou seat-row, který v dalších krocích naplníme sedadly tak, jak nám přijdou z API. Zatím do komponenty natvrdo vložte pár sedadel jen pro testovací účely. Komponentu SeatRow pak vložte do div u seats v komponentě SeatPicker.

2. Komponenta SeatRow bode očekávat prop s názvem row, ve které budou data pro jednu čadu sedadel. Pro testovací účely si vytvořte proměnnou testRow, která bude obsahovat takovýto objekt.

- 3. Předejte tuto proměnou komponentě SeatRow a uvnitř ní pomocí funkce map zobrazte jednotlivá sedadla. Jako key prop u jednotlivých sedadel můžete použít samotné číslo sedadla.
- 4. Nyní máme vše připraveno pro zobrazení správného plánku sedadel podle dat z API. Pracovat budeme v komponentě SeatPicker tam, kde máme testovací řadu sedadel. Když se podíváte do konzole na výpis objektu uloženého ve stavu journey komponenty Home, uvidíte, že máte velké štěstí. Ve vlastnosti seats je pole, které představuje přímo jednotlivé řady v autobusu.
- 5. Nyní je tedy potřeba údaje o sedadlech předat z komponenty Home do komponenty SeatPicker. Zároveň se bude později hodit i identifikátor spoje. Do komponenty SeatPicker tedy přidejte dvě props seats (vloží se do ní journey.seats) a journeyId (sem přijde journey.journeyId).
- 6. Ještě je potřeba upravit komponentu Home tak, aby komponenta SeatPicker byla vidět jedině v případě, že je stav journey jiný než null tedy stejně, jako se zobrazuje komponenta JourneyDetail. Ověřte v prohlížeči, že po vyhledání spoje se zobrazí podrobnosti cesty a také komponenta pro výběr sedadel zatím s vašimi testovacími sedadly.
- 7. Uvnitř komponenty SeatPicker projděte pole seats pomocí funkce map , a pro každý řádek pole vytvořte jednu komponentu SeatRow . I komponenty SeatRow potřebují prop key . Zde bohužel nemáme žádnou rozumnou datovou položku, kterou bychom jako klíč mohli použít. Vzpomeňme si však, že

funkce vložená do funkce map může mít dva parametry, druhý parametr je pořadové číslo (takzvaný index) aktuálního prvku. V tomto případě jej výjimečně můžeme použít jako key pro SeatRow.

- 8. Pokud jste všechno zařídili správně, měli byste po vyhledání cesty vidět sedadla rozmístěná stejně jako ve vzorovém designu stránky. V tuto chvíli už nám stačí pouze správně zobrazit zabraná sedadla. Zda je sedadlo zabrané udává vlastnost is0ccupied v datech z API. Stačí tedy komponentě Seat přidat prop is0ccupied a poslat do ní hodnotu obdrženou z API.
- 9. Uvnitř komponenty Seat zařiďte aby se na element svg přidala CSS třída seat--occupied ve chvíli, kdy je sedadlo zabrené.
- 10. Pokochejte se krásným plánkem sedadel a commitněte změny.

## Kódím.cz

Verze 2.0.0-beta.7