

MergeMap e Observables

Maria assistiu a aula e ficou empolgada ao saber que consegue ligar dois `Observables` do **RxJS** utilizando o operado `mergeMap()` que sempre receberá um callback, onde esse callback recebe como parâmetro a resposta do `Observable` anterior e **deve retornar obrigatoriamente** um novo `Observable` que dará continuidade ao próximo fluxo de execução assíncrona.

Então, para treinar seus novos conhecimentos, ela escreveu o código abaixo:

```
//agendamentos-service.ts

//código anterior omitido
export class AgendamentoService {

  //código anterior omitido

  agenda(agendamento) {
    //efetua o agendamento na API e retorna um observable
  }
}

//cadastro.ts

//código anterior omitido
export class AgendamentoPage {

  //construtor, injeção do Storage e do AgendamentoService omitidos

  cadastra(agendamento) {
    this._agendamentoService
      .agenda(agendamento)
      .mergeMap(() => this.salva(agendamento))
      .subscribe(
        () => alert('Agendamento realizado!'),
        () => alert('Falha no agendamento!')
      );
  }

  salva(agendamento) {
    let chave = //cria a chave...
    this._storage.set(chave, agendamento);
  }
}
```

Mas ela está insegura e deseja que você avalie o que ela fez! Portanto, marque a opção correta!

Selecione uma alternativa

A O código está correto!

B

Faltou retornar o resultado da operação do método `set` dentro de `salva()`. O código deveria ficar:

```
//cadastro.ts

//código anterior omitido
export class AgendamentoPage {

  //construtor, injeção do Storage e do AgendamentoService omitidos

  salva(agendamento) {
    let chave = //cria a chave...
    return this._storage.set(chave, agendamento);
  }
}
```

C

O código não está correto pois o operador `mergeMap()` deve receber um callback que **obrigatoriamente** tem que retornar um outro `Observable` e isso não ocorre uma vez que o método `salva()` invocado dentro do `mergeMap()` não retorna um `Observable` como precisamos.