

## Vuex چیست؟

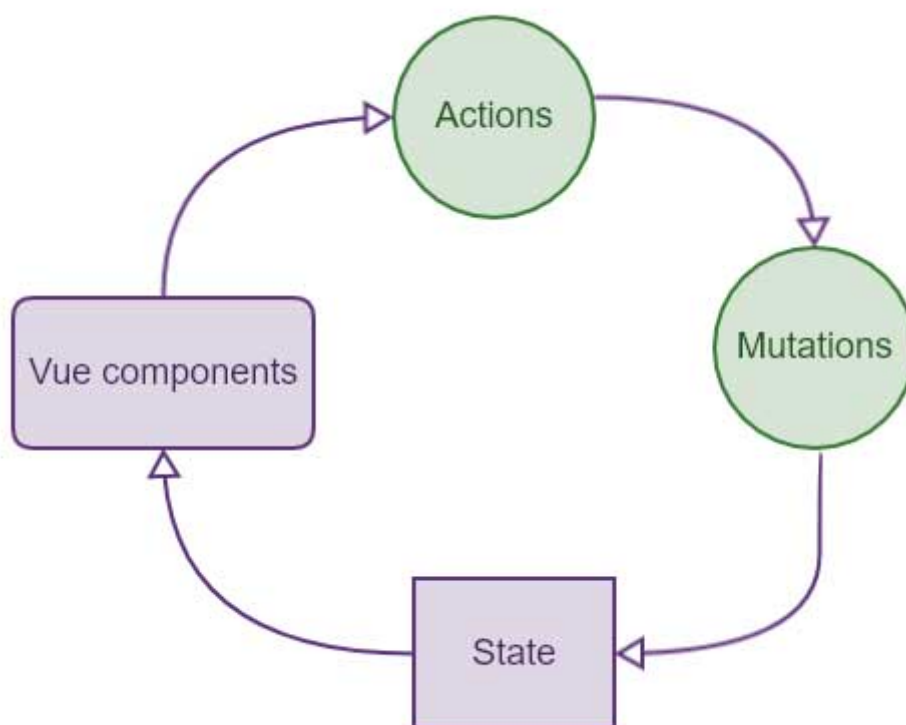
Vuex یک کتابخانه جاوا اسکریپت برای مدیریت وضعیت های برنامه است. این کتابخانه به عنوان یک منبع مرکزی برای داده های برنامه عمل می کند و می توانید در سراسر برنامه از داده های این منبع استفاده کنید.

ایده اولیه اینکار از کتابخانه های React Redux و Flux گرفته شده است. وقتی که برنامه سمت کلاینتی که مشغول توسعه آن هستیم در حال پیچیده تر شدن است، vuex در اینجا وارد عمل می شود

این کتابخانه از سه قسمت زیر تشکیل شده است:

- **State**: یک منبع مورد اعتماد است که برنامه تان از آن استفاده می کند.
- **View**: ویو یک نمایش از state است.
- **Action**: راه های ممکن که یک state بر اساس ورودی های کاربر تغییر می کند.

در زیر نمایش خیلی ساده از مفهوم جریان داده یک طرفه را مشاهده می کنید.



اکشن ها mutation را فراخوانی می کنند و mutation آبجکت را تغییر می دهد. اینکار، state را طبق اکشن اتفاق افتاده تغییر می دهد. به این ترتیب state برنامه تغییر کرده و طبق همان کامپوننت تغییرات را اطلاع داده و مطابق این تغییرات با آن رفتار می کند.

ابتدا کامپوننت vue یک اکشن را اجرا میکند، که اینکار توسط کلیک روی یک دکمه و در پاسخ به یک درخواست Http اتفاق می افتد.

به این ترتیب یک اکشن فراخوانی می شود، سپس مطابق با نوع اکشن تغییری اتفاق می افتد و وضعیت (state) را تغییر می دهد و کامپوننت های دیگر از این تغییر مطلع می شوند.

## State

اگر شما از آموزش vuex استفاده می کنید، تنها یک مخزن برای کامپوننت های vue.js برنامه تان خواهید داشت. این محل ذخیره سازی یک منبع منحصر به فرد از state های برنامه است. در این محل state های برنامه ذخیره می شوند. استفاده از vuex برای مدیریت state های برنامه vue.js کار آسانی است. تنها کاری که باید بکنید این است که state ها را توسط ارسال اکشن به آن تغییر دهید و اینکار را باید بطور غیرمستقیم انجام دهید چون ما می خواهیم state های بعدی را پیش بینی کنیم.

## Mutation

تنها راه برای تغییر state در vuex استفاده از mutation ها است. ما می توانیم بصورت مستقیم یک state را تغییر دهیم، اما در اینجا اینکار را انجام نمی دهیم، چون می خواهیم جزئیات هر مرحله ای که در برنامه اتفاق می افتد را به شما نشان دهیم. ما باید state بعدی را پیش بینی کنیم. به منظور اشکال زدائی از برنامه (دیباگ کردن)، state را بطور مستقیم تغییر نمی دهیم، بلکه فقط از طریق mutation ها اقدام به تغییر آنها می کنیم.

## اکشن ها

اکشن ها همانند mutation ها هستند، با این تفاوت که:

1. بجای تغییر یک state، اکشن ها mutation را اجرا می کنند.
2. اکشن ها در صورت تمایل می توانند شامل عملیات غیرهمزمان باشند

## چه موقع باید از Vuex استفاده کرد؟

اگر برنامه تعداد زیادی state داشته باشد و شما نتوانید همه آنها را مدیریت کنید، بهتر است از Redux و در این مقاله vuex استفاده کنید. اما با این حال موارد زیر را مطالعه بفرمایید:

1. اگر برنامه دائماً در حال بزرگتر شدن است، vuex یک روش کارآمد برای مدیریت کردن داده های هر عملیات است و اینکار را با vuex store یا مخزن vuex انجام می دهد.
2. برنامه های کوچک نیاز به vuex ندارند.
3. اگر state یی دارید که می خواهید با کامپوننت های دیگر به اشتراک بگذارید، آنگاه به vuex نیاز پیدا خواهید کرد. اشتراک گذاری state ها عملی رایج در هنگام استفاده از این قبیل کتابخانه ها است.

پس اگر یک state خاص توسط چندین کامپوننت در حال استفاده شدن است، بهتر است از آموزش Vuex استفاده کنید.