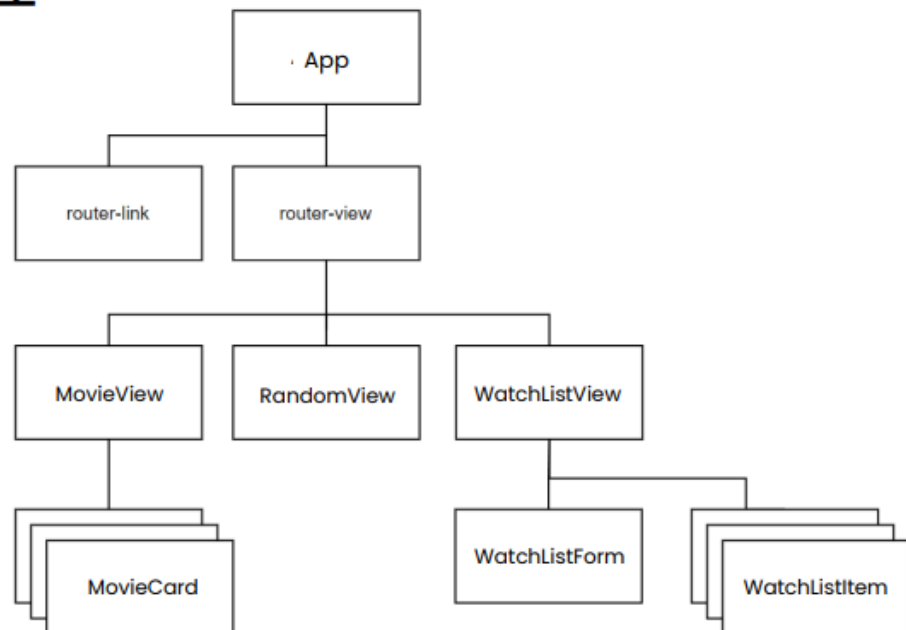


README

#Vuex

- 컴포넌트 트리 구조가 깊어질 때 전역에서 데이터를 편하게 활용하기 위해 사용
- component (dispatch : action 호출) ⇒ action (commit : mutation 호출) ⇒ mutation (only state 수정) ⇒ state

컴포넌트 구조



#index

```
index.js
import axios from 'axios'
import Vue from 'vue'
import Vuex from 'vuex'

Vue.use(Vuex)

export default new Vuex.Store({
  state: {
    // Key: '164f8d9d4790c9d3f59a0cbda5d95c69',
    url: 'https://api.themoviedb.org/3/movie/top_rated?api_key=<<key>>&language=ko-KR',
    movies: null,
    wishlist: [],
  },
  getters: {
  },
  mutations: {
```

```

    GETDATA(state,movies) {
      state.movies=movies
    },
    CREATE_WISH_ITEM(state,wishItem) {
      const wish = {
        title: wishItem,
        isSelected: false,
      }
      state.wishlist.push(wish)
    },
    COMPLETE(state,selectedItem) {
      state.wishlist.forEach((wish) => {
        if (wish === selectedItem) {
          wish.isSelected = !wish.isSelected
        }
      })
    }
  },
  actions: {
    getData(context) {
      axios.get(context.state.url)
        .then((response) => {
          const movies = response.data.results
          context.commit('GETDATA',movies)
        })
        .catch((error) => {
          console.log(error)
        })
    }
  },
  modules: {
  }
})

```

- API키를 받아 온 뒤 데이터 형식 확인
⇒ axios 이용해 비동기

#MovieView

Tweedles Movie

[MovieView](#) | [RandomView](#) | [WatchListView](#)



20 Century Girl

보라의 둘도 없는 친구 연두는 심장 수술을 위해 미국에 가면서 자신이 좋아하는 남자 백현진에 대한 모든 정보를 수집해 달라고 보라에게 부탁한다. 보라는 백현진의 가장 친한 친구 풍운호와 먼저 친해 지기로 한다. 하지만 보라의 서투른 계획은 예상치 못한 방향으로 전개된다. 새로운 세기가 오기 1년 전인 1999년, 17세가 된 보라는 첫 사랑의 열병에 빠진다.



The Godfather

시실리에서 이민온 뒤, 정치권까지 영향력을 미치는 거물로 자리잡은 돈 콜레오네는 갖가지 고인을 호소하는 사람들의 문제를 해결해주며 대부라 불리운다. 한편 솔로소라는 인물은 콜레오네가와 라이벌인 닷티리아 패밀리와 손잡고 새로운 마약 사업을 제안한다. 돈 콜레오네가 마약 사업에 참여하지 않기로 하자, 돈 콜레오네를 저격해 그는 중상을 입고 사경을 헤매게 된다. 그 뒤, 돈 콜레오네의 아들 소니는 조직력을 총 동원해 다른 패밀리들과 파를 부르는 전쟁을 시작하는



The Shawshank Redemption

죽방밥을 온 행 간부 앤디 듀프레인은 아내와 그녀의 정부를 살해했다는 누명을 쓴다. 주변의 증언과 살해 현장의 그릴듯한 증거들로 그는 중신형을 선고받고 악질범들만 수용한다는 지옥같은 고도소 소생기로 향한다. 인간 말종 쓰레기들만 모인 그곳에서 그는 이루 말할 수 없는 억압과 짐승보다 못한 취급을 당한다. 그러던 어느 날, 간수의 세금을 면제받게 해 준 덕분에 그는 일약 고도소의 비공식 회계사로 일하게 된다. 그 와중에 고도소 소장



罗小黑战记

술속의 집을 잃고 홀로 떠돌던 검은 고양이 요정 '소흑'은 도시 뒷골목에서 미스터리한 능력의 요정 '중식'을 만나 위기를 모면한다. '중식'의 무리와 버려진 섬에서 행복 한 시간을 보내는 '소흑'. 그러던 중 최강 능력의 집행자 '무한'이 '중식'을 찾아 섬에 오자 '중식' 일행은 달아나고, '소흑'만 남게 된다. 홀로 남은 '소흑'을 요정들의 회관으로 데려가려는 '무한'과 '무한'을 무서운 인간이라 여겨 도망치려는 '소흑'. 둘은 여정 속에서 점점 마음을 열게 되고, '무한'은 '소흑'에게 특



The Godfather Part II

아버지의 장례식 도중에 말아들 파올로가 총에 맞아 죽고, 비토(로버트 드 니로)는 겨우 도망쳐 미국으로 건너온다. 대부로 성장한 후 비토는 다시 지지를 찾아 복수를 한다. 새롭게 등장한 젊은 대부 마이클(알 파치노)은 본거지를 라스 베가스으로 옮기고 가족의 사업을 가능한 합법적인 것으로 바꾸려고 애 쓴다. 그런 과정에서 자신을 제거 하려는 음모를 알게되고 그는 냉혹하고 신숙하게 반대파들을 제거, 조직을 더욱 확대해 나간다. 이를 위해 마이클은 배신한 형마저 죽이



Schindler's List

2차 세계대전 당시 독일군이 점령한 폴란드 시류에 맞춰 자신의 성공을 추구하는 기회주의자 싘들러는 유대인이 경영하는 그릇 공장을 인수한다. 그는 공장을 인수하기 위해 나찌 당원이 되고 독일군에게 뇌물을 바치는 등 갖은 방법을 동원한다. 그러나 냉혹한 기회주의자였던 싘들러는 유대인 회계사인 스타인과 친분을 맺으면서 냉혹한 유대인 학살에 대한 양심의 소리를 듣기 시작한다. 마침내 그는 강제 수용소로 끌려가 죽음을 맞게될 유대인들을 구해내기로 결심하고, 독일

```
# MovieView.vue
```

```
<template>
  <div class="row d-flex justify-content-center">
    <MovieCard
      v-for="movie in getMovieData "
      :key=movie.id
      :movie=movie
      id="card"
    />
    <!-- {{getMovieData}} -->
    <h5 id="h5">@created by tweedles</h5>
  </div>
</template>
```

```
<script>
import MovieCard from '@/components/MovieCard'
export default {
  name: 'MovieView',
  components: {
    MovieCard,
  },
  methods: {
    getData() {
      this.$store.dispatch('getData')
    }
  },
  created() {
    this.getData()
  },
  computed: {
    getMovieData() {
      return this.$store.state.movies
    }
  }
}
</script>
```

```
<style>
#card {
  padding: 5px;
}
```

```

}

#h5 {
  color: lightgray;
}
</style>

```

```

# MovieCard.vue

<template>
  <div class="card" style="width: 18rem; margin-left: 20px; margin-bottom: 20px;">
    
    <div class="card-body">
      <h5 class="card-title">{{ movie.original_title }}</h5>
      <hr>
      <p class="card-text">{{ movie.overview }}</p>
      <!-- <a href="#" class="btn btn-primary">Go somewhere</a -->
    </div>
  </div>
</template>

<script>
export default {
  name: 'MovieCard',
  props: {
    movie: Object,
  },
  computed: {
    posterPath() {
      return `https://image.tmdb.org/t/p/original/${this.movie.poster_path}`
    }
  }
}
</script>

<style>
</style>

```

#RandomView

Tweedles Movie

[MovieView](#) | [RandomView](#) | [WatchListView](#)



```
RandomView.vue
<template>
  <div class="container">
    <div class="d-flex justify-content-center">
      <!-- {{ randomMovie }} -->
      <div class="card shadow-sm p-3 mb-5 bg-body rounded" style="width: 30rem;">
        
        <div class="card-body">
          <h5 class="card-title">{{ randomMovie.original_title }}</h5>
          <hr>
          <p class="card-text">{{ randomMovie.overview }}</p>
          <!-- <a href="#" class="btn btn-primary">Go somewhere</a> -->
        </div>
      </div>
    </div>
  </div>
</template>

<script>
export default {
  name: 'RandomView',
  data() {
    return {
      randomMovie : null,
    }
  },
  methods: {
    getData() {
      this.$store.dispatch('getData')
    }
  },
  created() {
    this.getData()
    const idx = Math.floor(Math.random() * this.$store.state.movies.length)
    return this.randomMovie = this.$store.state.movies[idx]
  },
  computed: {
```

```

    posterPath() {
      return `https://image.tmdb.org/t/p/original/${this.randomMovie.poster_path}`
    }
  }
}
</script>

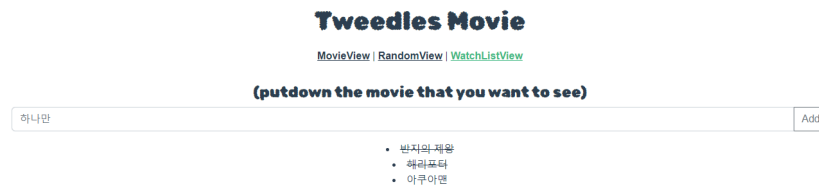
<style>

</style>

```

- 화면에 처음부터 띄워놓기 위해 created() 함수를 이용
⇒ this.\$store.state.movies의 기본값이 null이므로 data를 받아오기 위해 getData() 함수를 실행
- random으로 element를 뽑는 함수를 이용하여 idx값을 설정하고, 그 idx에 해당하는 data pick

#WatchListView



```

WatchListView.vue
<template>
  <div class="row">
    <h4 style="font-family: 'Rubik Bubbles', cursive;">(putdown the movie that you want to see)</h4>
    <div class="col-2"></div>
    <div class="col-8">
      <div class="input-group mb-3">
        <input type="text" class="form-control"
          placeholder="하나만" aria-label="하나만" aria-describedby="button-addon2"
          v-model="wishItem"
          @keyup.enter="createWishItem"
        >
        <button class="btn btn-outline-secondary" type="button" id="button-addon2"
          @click="createWishItem">Add</button>
      </div>
    </div>
  </div>

```

```

    </div>
    <WatchListItem
      v-for="(wishitem,idx) in wishlist"
      :key = idx
      :wishitem = wishitem
    />
  </div>
</template>

<script>
import WatchListItem from '@components/WatchListItem'
export default {
  name: 'WatchListView',
  components: {
    WatchListItem,
  },
  data() {
    return {
      wishItem : null
    }
  },
  methods: {
    createWishItem() {
      this.$store.commit('CREATE_WISH_ITEM', this.wishItem)
    }
  },
  computed: {
    wishlist() {
      return this.$store.state.wishlist
    }
  }
}
</script>

<style>

</style>

```

```

WatchListItem.vue
<template>
  <div>
    <li
      @click="complete"
      :class="{ 'isSelected' : wishitem.isSelected }"
    >
      {{ wishitem.title }}
    </li>
  </div>
</template>

<script>
export default {
  name: 'WatchListItem',
  props:{
    wishitem : Object
  },
  methods: {
    complete() {
      this.$store.commit('COMPLETE', this.wishitem)
    }
  }
}
</script>

```

```
<style>
.isSelected {
  text-decoration: line-through;
}
</style>
```

- input form으로 데이터를 받아 enter를 누르거나 add click했을 때 createWishItem함수가 실행된다.
→ 이후 store에 만들어둔 리스트에 data를 계속 저장하고, computed로 불러와 data 사용