Main.js -> vue 실행하는 index같은 개념, 라이브러리나 전역적인 설정 추가할 때 여기에 추가

Views -> 페이지

App -> 페이지들의 집합

<template></template>태그로 감싸서 사용

<router-link> - 링크를 걸고 링크에 대한 걸 <router-view/>에 뿌려줌

Router.js – 라우터 정의 되어있음

<template>

<div class="home">

<img alt="Vue logo" src="../assets/logo.png">

<hello-world></hello-world> //사용할 때

</div>

</template>

<script>

// @ is an alias to /src

import HelloWorld from '../components/HelloWorld.vue'

export default {

name: 'home',

components: {

HelloWorld //컴포넌트 불러올 때

}

}

</script>

<template>

<div class="home">

<img alt="Vue logo" src="../assets/logo.png">

<hello-world></hello-world>

<div v-for="aa in aas" :key="aa"> //for문

{{aa}}

</div>

</div>

</template>

<script>

// @ is an alias to /src

import HelloWorld from '../components/HelloWorld.vue'

export default {

name: 'home',

components: {

HelloWorld

},

data () { //데이터 만드는 필드

return {

aas: [

'vue',

'어렵다',

'이것도 쉬운 편이다',

] //aa=[vue, 어렵다, 이것도 쉬운 편이다]

}

}

}

</script>

props

: 부모에서 보내준 데이터

<hello-world :aas="aas"></hello-world>

자식한테 데이터 넘겨줄 때 ( : == v-bind)

Vuex – 중앙 집중형 데이터 관리

Actions – 서버에서 데이터 올때까지 기다림 -> commit -> Mutations – 데이터 변경 -> mutate -> state – 데이터 변경된 부분만 새로고침 ->