



初始化视图：实例创建和挂载

小羊们，我们第一步先实现一个初始化视图的功能，写一个 `createApp()` 暂时什么也不做。

```
// src/minivue/index.js
export function createApp(rootComponent) {
  return {
    mount(selector) {},
  };
}
```

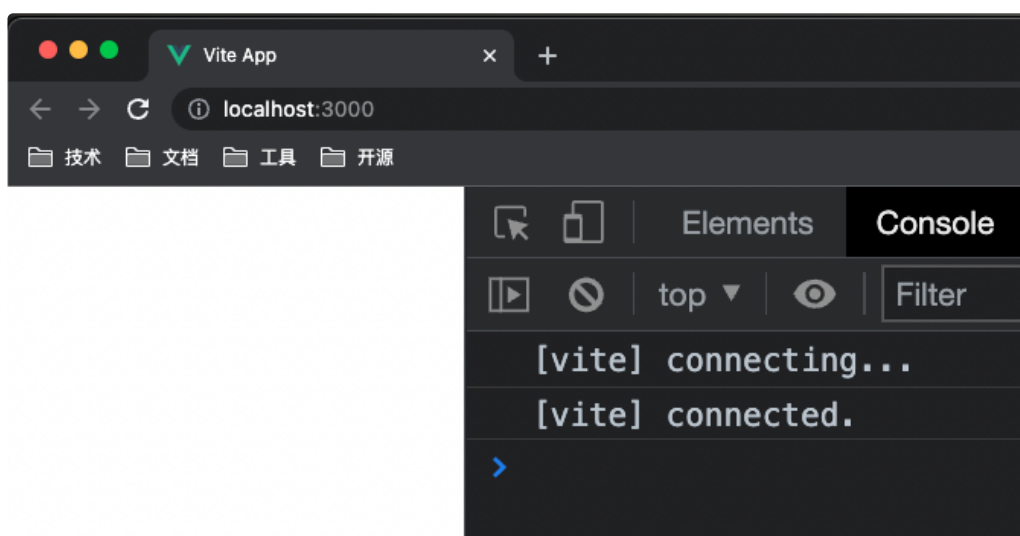
TypeScript

用一下试试

```
// main.js
import { createApp } from './mini-vue'
```

TypeScript

现在没有报错，但是什么也没有！



我们知道 `mount(container)` 作用是将组件 `render()` 的内容追加到 `container`，我们添加这个逻辑：

```
JavaScript

export function createApp(rootComponent) {
  return {
    mount(selector) {
      const container = document.querySelector(selector);
      const el = rootComponent.render.call(rootComponent.data());
      // parent.innerHTML = "";
      container.appendChild(el);
    },
  };
}
```

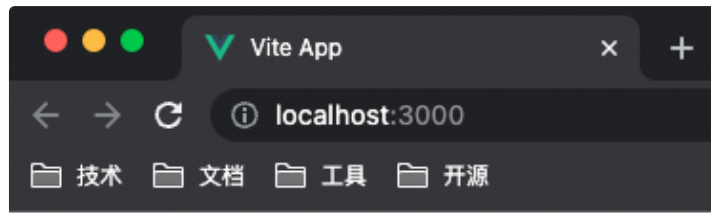
这样我们就需要先简化 App，它现在还不是我们想要的样子：

```
JavaScript

// import App from './App.vue'

createApp({
  data() {
    return {
      title: 'hello, mini-vue!'
    }
  },
  render() {
    const h3 = document.createElement('h3')
    h3.textContent = this.title
    return h3
  }
}).mount('#app')
```

可以看到结果啦！



hello, mini-vue!

总结一下：

- createApp()要返回一个对象，这就是 App 实例
- 实例的 mount 方法会将传入组件的渲染结果追加到宿主元素
- 我们没有编译器，因此需要手动实现组件 render 函数