

开课吧

Vue 3.0训练营

村长@每晚8点

关于我

- 村长：前高伟达架构师，10年前端经验老司机，开课吧声优、奶爸
- github：<https://github.com/57code>
- b站：<https://space.bilibili.com/480140591>
- 掘金：<https://juejin.cn/user/325111174926350>
- 抖音：前端杨村长

关于开课吧

发展历程



坚实的
每一步
是为了下一步
更好



关于开课吧

- 加入W3C，与阿里巴巴、腾讯、苹果、IBM等W3C会员一起维护WWW 标准，全国首家加入W3C教育机构
- CNCC 2020（中国计算机大会）唯一教育合作伙伴，教育和技术实力得到业界广泛认可
- 目前已服务超过400万+付费学员
- 前端会客厅—尤雨溪、贺师俊、周爱民、甄炎鲲、林顺
- 获得A轮5.5亿元融资，该笔融资也是在线教育史上最大一笔

为什么有这次训练营

- Vue3正式版发布
- 精通vue的重要性
- 分享源码学习心得
- 绝佳的学习机会

训练营内容

DAY 1 初始化流程+自定义渲染器实战

- ▼ Vue3源码调试环境准备
- ▼ Vue3整体架构、初始化流程分析
- ▼ 造轮子
- ▼ 实战：实现自定义canvas渲染器

DAY 2 响应式应用+实战

- ▼ 响应式原理Vue2 vs Vue3
- ▼ composition-api新语法实战
- ▼ Vue3数据响应式源码剖析
- ▼ 造轮子：手写reactivity模块

今日目标

- 调试环境搭建
- 项目架构分析
- 初始化流程剖析
- 手写实现
- 自定义渲染器实战

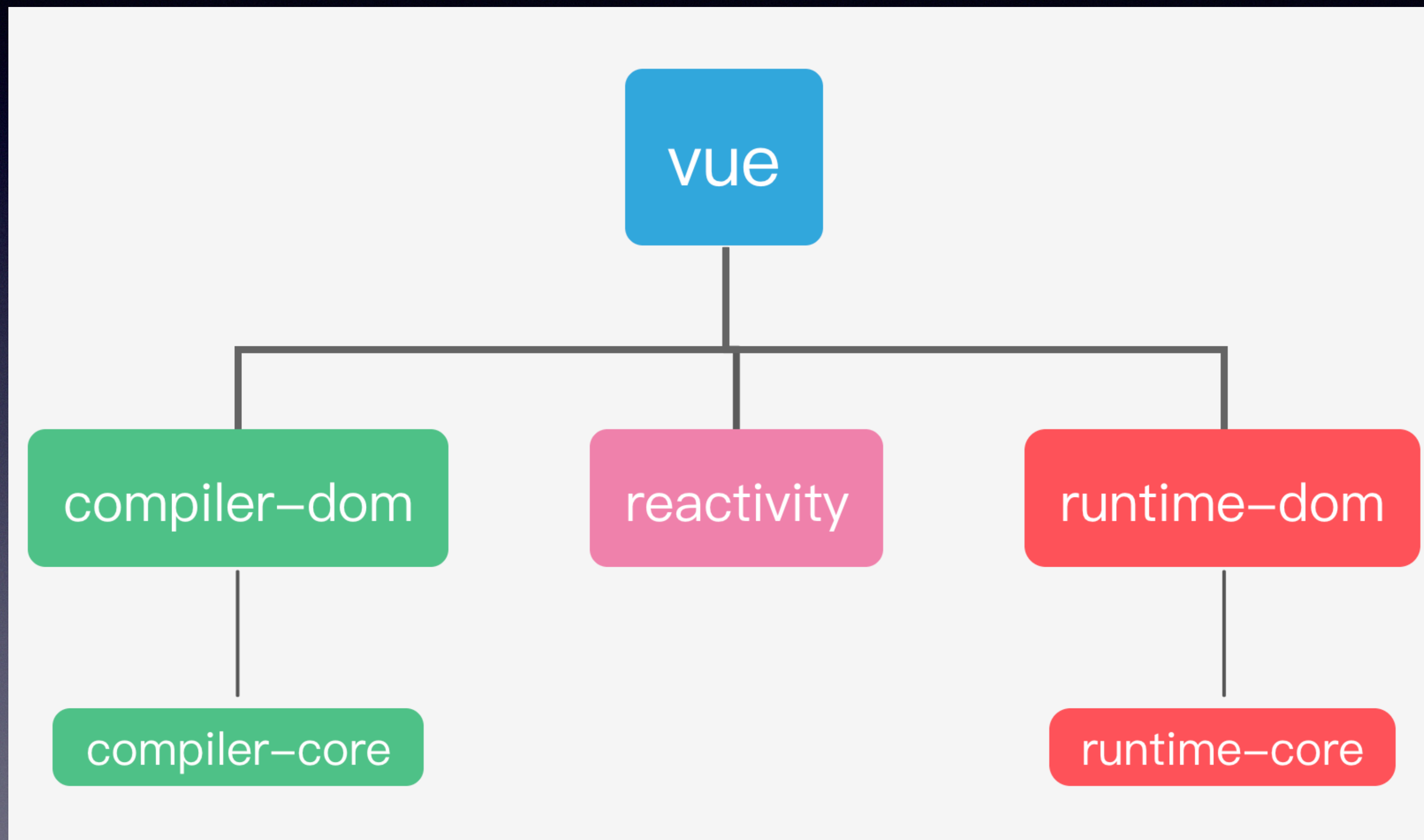
调试环境搭建

- 迁出Vue3源码: `git clone https://github.com/vuejs/vue-next.git`
- 安装依赖: `yarn --ignore-scripts`
- 生成sourcemap文件, package.json

```
"dev": "node scripts/dev.js --sourcemap"
```

- 编译: `yarn dev`

项目架构分析



初始化流程分析

componentEffect	renderer.ts:1234
reactiveEffect	effect.ts:94
effect	effect.ts:63
setupRenderEffect	renderer.ts:1233
mountComponent	renderer.ts:1172
processComponent	renderer.ts:1101
patch	renderer.ts:462
render	renderer.ts:2090
mount	apiCreateApp.ts:221
app.mount	index.ts:70
(anonymous)	init.html:10

手写初始化流程

```
const createAppAPI = (render) => {  
  return function createApp(rootComponent) { }  
}  
  
const createRenderer = (options) => {  
  const render = (vnode, container) => {}  
  return {  
    render,  
    createApp: createAppAPI(render)  
  }  
}  
  
const renderer = createRenderer({})  
  
const Vue = {  
  createApp(options) {  
    return renderer.createApp(options)  
  }  
}
```


自定义渲染器实践

- 范例：编写一个渲染器把数据用canvas以柱状图形式绘制出来
- 核心思路是按照自定义的节点操作逻辑处理视图和数据

自定义渲染器实践

```
const { createRenderer } = Vue

// 创建一个渲染器，给它提供节点和属性操作
const nodeOps = {}
const patchProp = (el, key, prevValue, nextValue) => {}
const renderer = createRenderer({
  ...nodeOps,
  patchProp,
});
```


自定义渲染器实践

```
const draw = (el) => {
  ctx.clearRect(0, 0, canvas.width, canvas.height)
  if (el.tag === 'bar-chart') {
    const { data } = el;
    const barWidth = canvas.width / 10,
          gap = 20,
          paddingLeft = (data.length * barWidth + (data.length - 1) * gap) / 2,
          paddingBottom = 10;

    data.forEach(({ title, count, color }, index) => {
      const x = paddingLeft + index * (barWidth + gap)
      const y = canvas.height - paddingBottom - count
      ctx.fillStyle = color
      ctx.fillRect(x, y, barWidth, count)
      // text
    });
  }
}
```


总结时间

- 总结
- 灵魂拷问：今天我学到了啥？
- 大家学习中经常遇到什么问题？

下次课内容预告

- vue3响应式革新
- vue3 vs vue2
- 手写vue3响应式课前福利实现
- 相关源码剖析
- 加餐：vue3全新vdom架构解析

作业练习

- 大家到群领取今晚作业，巩固学习成果