

极客学院
jikexueyuan.com

JSX语法及特点介绍

JSX语法及特点介绍 — 课程概要

- 什么是JSX
- 如何使用JSX
- 非DOM属性介绍
- JSX解释器架构介绍



什么是JSX

什么是JSX

- JSX的发展历程
- JSX的特点

什么是JSX — JSX的发展历程

JSX = JavaScript XML

此JSX非彼JSX

JSX是：

- 基于ECMAScript的一种新特性
- 一种定义带属性树结构的语法

JSX不是：

- XML或者HTML
- 一种限制

什么是JSX — **JSX**的特点

类XML语法
容易接受

增强JS语义

结构清晰

抽象程度高

代码模块化



如何使用JSX

如何使用JSX

- JSX的语法
- JSX语法实例演示

如何使用JSX — JSX的语法

元素名

子节点
• 首字母大小写

• 嵌套

• 求值表达式

• 驼峰命名

• htmlFor和className

```
1 var HelloMessage = React.createClass({
2   render: function() {
3     return <div className="test">Hello {this.props.name}</div>;
4   }
5 });
6
7 React.render(<HelloMessage name="John" />, mountNode);
```

节点属性

如何使用JSX — **JSX语法实例演示**

注释、CSS样式以及嵌套

条件判断的四种写法

万能的函数表达式

非DOM属性介绍

非DOM属性介绍

- 为什么有非DOM属性
- 如何使用非DOM属性

非DOM属性介绍 — 为什么有非DOM属性

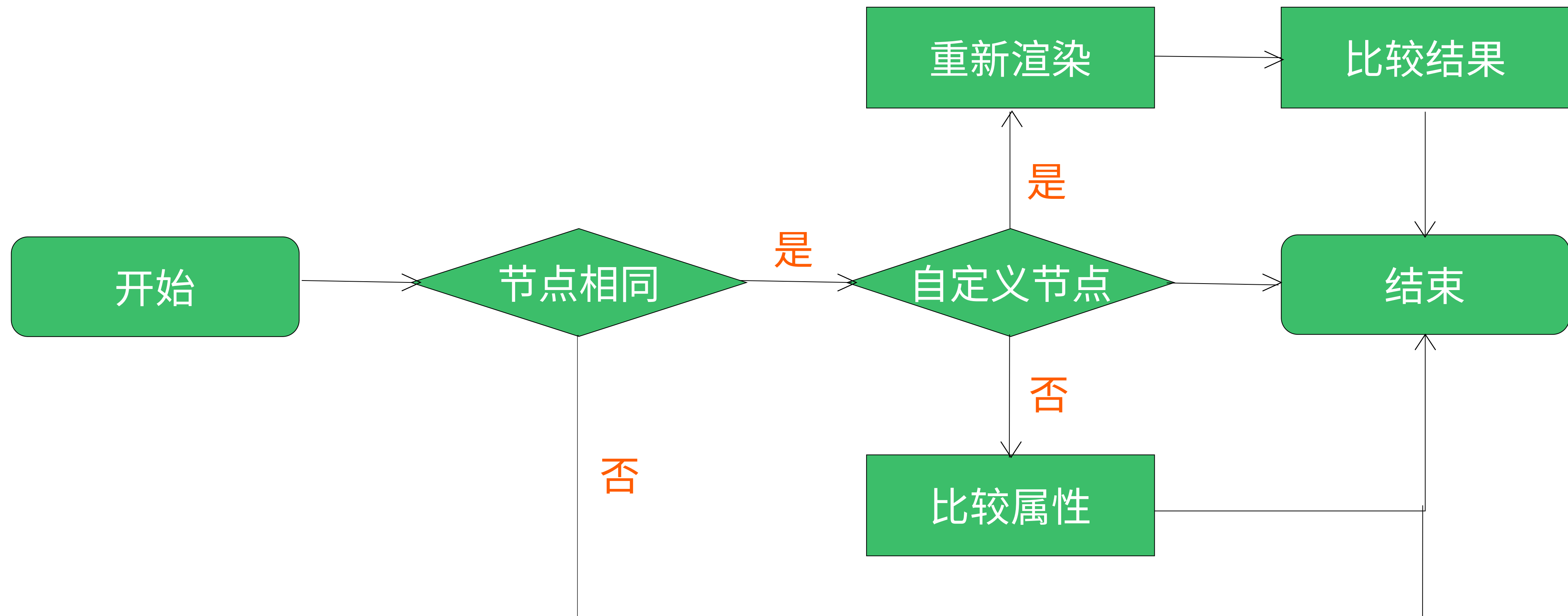
非DOM属性：**dangerouslySetInnerHTML**、**ref**、**key**

dangerouslySetInnerHTML：在JSX中直接插入**HTML**代码

ref：父组件**引用**子组件

key：提高渲染**性能**

非DOM属性介绍 — 为什么有非DOM属性



React diff算法流程图

非DOM属性介绍 — 如何使用非DOM属性

dangerouslySetInnerHTML演示

ref演示

key演示

JSX解释器架构介绍

JSX解释器架构介绍

- 源码阅读方法
- 理解解释器架构

JSX解释器架构介绍 — 源码阅读方法

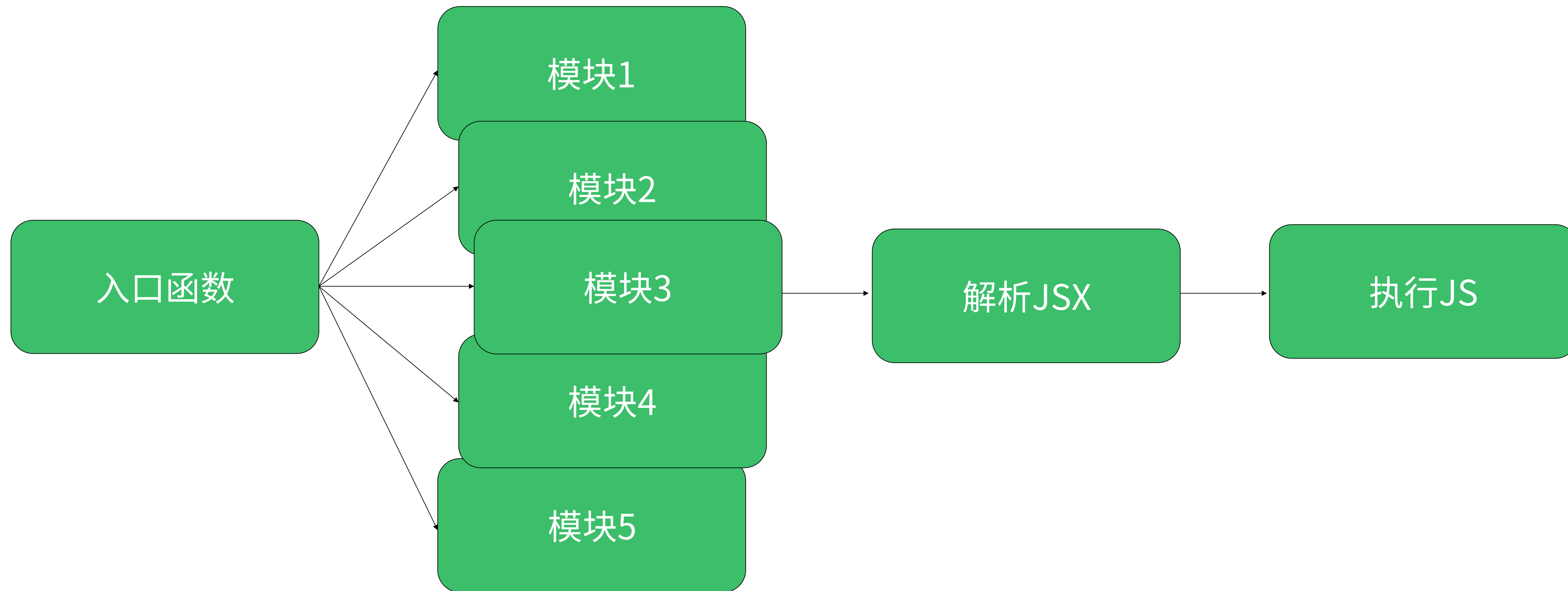
从执行顺序入手

适当忽略细节

重视烂笔头

反复阅读

JSX解释器架构介绍 — 理解解释器架构



课程主标题

本课程中我们学习了JSX，你应当掌握了以下知识：

- JSX的特点
- JSX的用法
- JSX解释器架构

你可以使用这些技巧来编写React组件，如果想继续提高，你可以继续阅读JSX解释器代码并在极客学院学习React系列课程。

极客学院

jikexueyuan.com

中国最大的IT职业在线教育平台

