

# What is Vue?

The Progressive JavaScript Framework

# What is Vue?

- A progressive framework for building user interfaces
- Incrementally adoptable – use as little or as much as you need
- Reactive data binding out of the box
- Component-based architecture
- Excellent TypeScript support

# Single File Components

- `.vue` files combine template, logic, and styles
- Clear separation of concerns in one file
- Scoped styles by default
- Full TypeScript support with `lang="ts"`
- `<script setup>` for cleaner syntax

```
<template>
  <div class="greeting">
    <h1>{{ message }}</h1>
  </div>
</template>

<script setup lang="ts">
  import { ref } from 'vue'

  const message = ref<string>('Hello, Vue!')
</script>

<style scoped>
.greeting {
  color: #42b883;
  font-size: 2rem;
}
</style>
```

# Reactivity with ref()

```
<script setup lang="ts">
import { ref } from 'vue'

const count = ref<number>(0)

function increment(): void {
  count.value++
}

function decrement(): void {
  count.value--
}
</script>

<template>
  <button @click="increment">Increase</button> <br />
  <button @click="decrement">Decrease</button> <br />
  Count: {{ count }}
</template>
```

Increase

Decrease

Count: 0

# Template Directives

```
<template>
  <p v-if="isVisible">Now you see me!</p>
  <ul><li v-for="item in items" :key="item.id">{{ item.name }}</li></ul>
  
  <button @click="handleClick">Click me</button>
  <input v-model="searchQuery" placeholder="Search..." />
</template>
```

# Options API vs Composition API

- **Options API:** The classic Vue 2 style – data, methods, computed in separate blocks
- **Composition API:** The Vue 3 way – organize by feature, not option type
- `<script setup>` is the recommended syntax for Composition API
- Both are fully supported – choose based on preference and project needs

# Options API → Composition API

```
<script setup lang="ts">
import { ref, computed } from 'vue'

const count = ref<number>(0)

const doubled = computed<number>(() => count.value * 2)

function increment(): void {
  count.value++
}
</script>

<template>
  <button @click="increment">{{ count }} ({{ doubled }})</button>
</template>
```



# Questions?

Curious about Vue? What framework do you currently use?



# Thank You!

Stay tuned for more Vue lightning talks!