

Mastering Flowcharts with Mermaid.js

Creating diagrams doesn't have to be complex or require graphic design tools. [Mermaid.js](#) is a JavaScript-based tool that allows developers and writers to create diagrams and visual flows using a simple Markdown-like syntax. Whether you're documenting a process, planning a feature, or teaching a concept, Mermaid can help make your ideas more digestible.

🔧 Getting Started: Adding Mermaid to HTML

You can use Mermaid directly in the browser by loading it from a CDN as a JavaScript module. Add the following script in your HTML `<head>`:

```
<script type="module">
import mermaid from 'https://cdn.jsdelivr.net/npm/mermaid@10/dist/mermaid.esm.min.mjs';
mermaid.initialize({ startOnLoad: true });
</script>
```

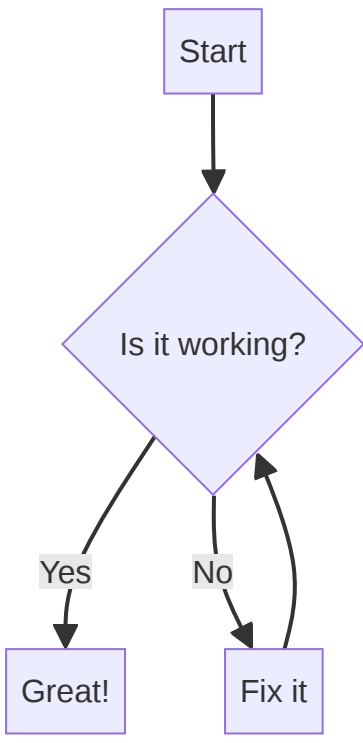
This will automatically render any element with the class `mermaid` on page load.

📌 Basic Syntax: Defining a Flowchart

Mermaid uses a keyword like `graph TD` (Top-Down) or `graph LR` (Left-Right) to define the layout direction.

```
<div class="mermaid">
graph TD
  A[Start] --> B{Is it working?}
  B -->|Yes| C[Great!]
  B -->|No| D[Fix it]
  D --> B
</div>
```

Live Preview:

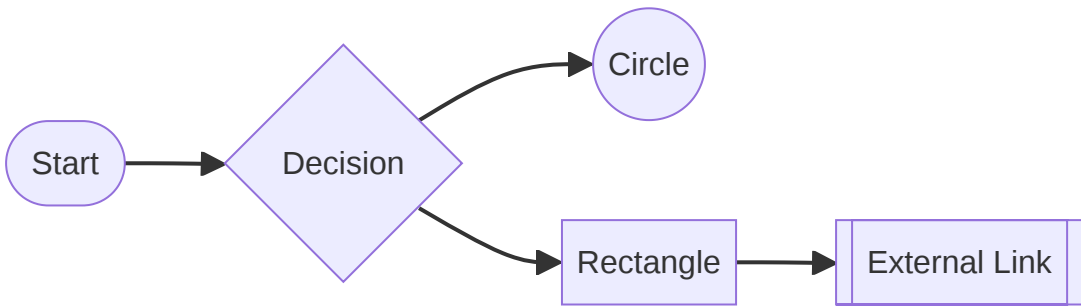


🧱 Node Shapes & Syntax

Mermaid supports multiple types of nodes using different bracket styles:

- `[Text]` – Rectangle (default)
- `((Text))` – Circle (or ellipse)
- `{Text}` – Rhombus (used for decisions)
- `[[Text]]` – Link node
- `[(Text)]` – Rounded edges

```
<div class="mermaid">
graph LR
  A([Start]) --> B{Decision}
  B --> C((Circle))
  B --> D[Rectangle]
  D --> E[External Link]
</div>
```

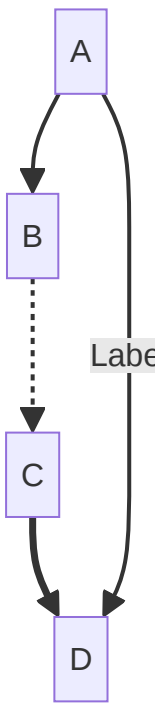


🔗 Link Types

Mermaid supports various arrow types and text labels:

- `-->` : Solid arrow
- `-.->` : Dotted arrow
- `==>` : Thick arrow
- `-->|Text|` : Labeled arrow

```
<div class="mermaid">
graph TD
  A --> B
  B -. C .-> C
  C ==> D
  A -->|Label| D
</div>
```



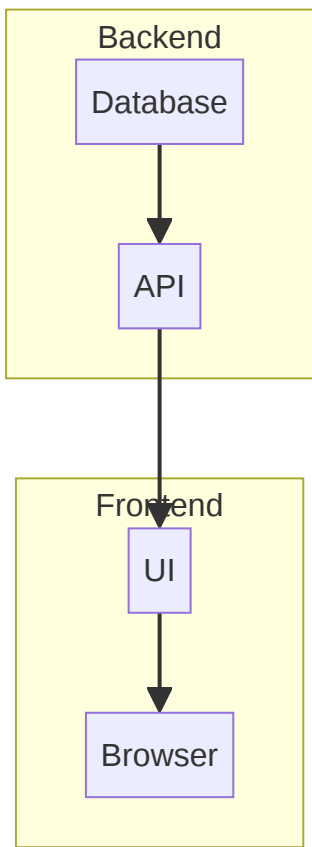
📦 Grouping with Subgraphs

Use `subgraph` to logically group nodes together:

```
<div class="mermaid">
graph TB
  subgraph Backend
    A[Database] --> B[API]
  end

  subgraph Frontend
    C[UI] --> D[Browser]
  end

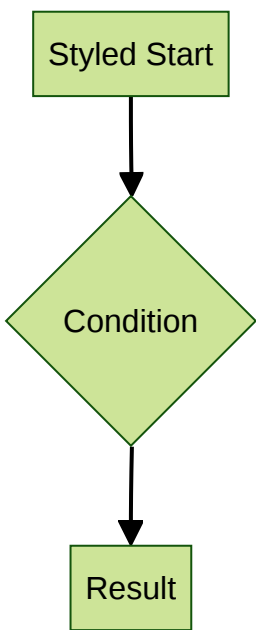
  B --> C
</div>
```



🎨 Styling (Optional)

Mermaid supports CSS-like styling and themes using directives or global config.

```
<div class="mermaid">
%%{ init: { "theme": "forest" } }%%
graph TD
  A[Styled Start] --> B{Condition}
  B --> C[Result]
</div>
```



📦 Real-World Use Cases

- ✅ Documenting CI/CD pipelines
- ✅ Visualizing algorithms or data flow
- ✅ Explaining decision-making logic
- ✅ UML-lite for dev documentation
- ✅ Auto-generating diagrams from code pipelines (with tools like Docusaurus)

📖 Further Reading

Dive deeper into the official Mermaid flowchart syntax guide: <https://mermaid.js.org/syntax/flowchart.html?id=flowcharts-basic-syntax>

🚀 Final Thoughts

Mermaid is a fantastic, no-frills tool for turning code into clarity. It's especially helpful in technical environments where lightweight diagrams are needed without switching tools or using proprietary software.