

# Notes

## Definition List

`Lorem ipsum dolor sit amet`

Sed sagittis eleifend rutrum. Donec vitae suscipit est. Nullam tempus tellus non sem sollicitudin, quis rutrum leo facilisis.

`Cras arcu libero`

Aliquam metus eros, pretium sed nulla venenatis, faucibus auctor ex. Proin ut eros sed sapien ullamcorper consequat. Nunc ligula ante.

Duis mollis est eget nibh volutpat, fermentum aliquet dui mollis. Nam vulputate tincidunt fringilla. Nullam dignissim ultrices urna non auctor.

## Partially completed list

- `Lorem ipsum dolor sit amet, consectetur adipiscing elit`
- `Vestibulum convallis sit amet nisi a tincidunt`
  - `In hac habitasse platea dictumst`
  - `In scelerisque nibh non dolor mollis congue sed et metus`
  - `Praesent sed risus massa`
- `Aenean pretium efficitur erat, donec pharetra, ligula non scelerisque`

## Tables

Suite Scenarios Supported events Hazard Warnings Smoke and Fire Detection  
"Smoke event detected Fire event detected Sparks detected Open flames detection"  
No smoking/no-vaping zones "Smoking event detected Vaping event detected" Spills & Leaks detection (Liquids) "Water puddle detected Water leak from equipment detected Spill event detected Slippery sign detected" Gas leak detection Gas leak detected Missing fire-extinguisher Missing fire extinguisher Blocked exit monitoring Blocked exit detected Equipment temperature "Temperature exceeds limit Temperature subceeds limit" Slip/trip and fall detection Blocker on pathway detected. Equipment rust-and-corrosion Rust or corrosion event detected

---

## Hazard Warnings

Scenario name	Supported Events
smoke-and-fire-detection 	Fire event detected Smoke Event Detected Sparks Detected Open Flames Detected
no-smoking-no-vaping-zones 	Smoking event detected Vaping event detected
smoke-and-fire-detection 	Fire event detected Smoke Event Detected Sparks Detected Open Flames Detected

Smoke and Fire Detection No smoking/no-vaping zones Spills & Leaks detection (Liquids) Gas leak detection Missing fire-extinguisher Blocked exit monitoring Equipment temperature Slip/trip and fall detection Equipment rust-and-corrosion

## Callout



### Note

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.



### Abstract

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.

### **Warning**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.

### **Danger**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.

### **Success**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.

### **Question**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.

### **Tip**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.

### **Quote**

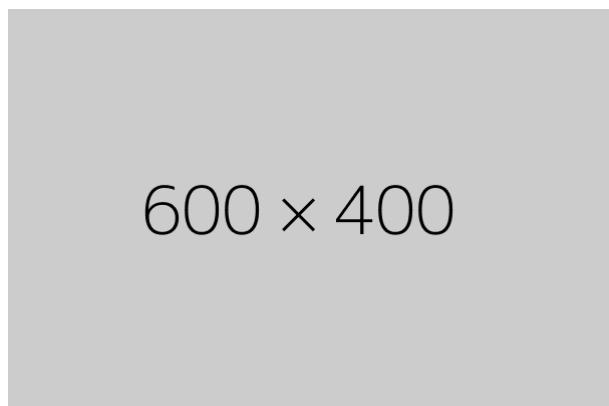
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.



### VisionAI Comment

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.

## Images



*Image caption*



*Image caption*

## Side-by-side code-blocks

### Material for MkDocs

```
name: ci # (1)!
on:
  push:
    branches:
      - master # (2)!
      - main
  permissions:
    contents: write
jobs:
  deploy:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v3
      - uses: actions/setup-python@v4
        with:
          python-version: 3.x
      - uses: actions/cache@v2
        with:
          key: ${{ github.ref }}
          path: .cache
      - run: pip install mkdocs-material # (3)!
      - run: mkdocs gh-deploy --force
```

1. You can change the name to your liking.
2. At some point, GitHub renamed `master` to `main`. If your default branch is named `master`, you can safely remove `main`, vice versa.
3. This is the place to install further [MkDocs plugins] or Markdown extensions with `pip` to be used during the build:

```
pip install \
  mkdocs-material \
  mkdocs-awesome-pages-plugin \
  ...
```

### Insiders

```
name: ci
on:
  push:
    branches:
      - master
      - main
  permissions:
    contents: write
jobs:
  deploy:
```

```

runs-on: ubuntu-latest
if: github.event.repository.fork == false
steps:
  - uses: actions/checkout@v3
  - uses: actions/setup-python@v4
    with:
      python-version: 3.x
  - uses: actions/cache@v2
    with:
      key: ${{ github.ref }}
      path: .cache
  - run: apt-get install pngquant # (1)!
  - run: pip install git+https://${{GH_TOKEN}}@github.com/squidfunk/mkdocs-material-insiders.git
  - run: mkdocs gh-deploy --force
env:
  GH_TOKEN: ${{ secrets.GH_TOKEN }} # (2)!
```

## Code blocks annotations

Material for MkDocs is published as a [Python package] and can be installed with `pip`, ideally by using a [virtual environment]. Open up a terminal and install Material for MkDocs with:

```

theme:
  features:
    - content.code.annotate # (1)
```

1. 🤖 I'm a code annotation! I can contain `code`, **formatted text**, images, ... basically anything that can be written in Markdown.

## Version/tags flags

[↗ 9.0.0][Code copy button support] · ⚡ Feature flag

## Adding a title to a code-block

```

bubble_sort.py

def bubble_sort(items):
    for i in range(len(items)):
        for j in range(len(items) - 1 - i):
            if items[j] > items[j + 1]:
                items[j], items[j + 1] = items[j + 1], items[j]
```

## Hover, Important information

TODO:

### Icon with tooltip

```
{title="Important information" }
```



hello this is important.



## Adding footnotes

This is how you add footnotes.

## Add buttons

[Subscribe to our newsletter](#)

another button

[Subscribe to our newsletter](#)

## Data tables

### This is table 1

Method	Description
`GET`	:material-check: Fetch resource
`PUT`	:material-check-all: Update resource
`DELETE`	:material-close: Delete resource

Method	Description
GET	✓ Fetch resource
PUT	✗ Update resource
DELETE	✗ Delete resource

## Text highlighting

- This was marked
- This was inserted
- ~~This was deleted~~

## Text block with title

### Emoji

Here you can add anything..



## Mermaid

### Sequence Diagram

```
sequenceDiagram
    actor user as User
    participant p as Button

    user --> p: click
```

### Flowchart

```
graph TD
    A[Hard edge] -->|Link text| B(Round edge)
    B --> C{Decision}
    C -->|One| D[Result one]
    C -->|Two| E[Result two]
```

## Another chart

```
flowchart TD
    Start -->|Decision 1| A
    A -->|Yes| B
    A -->|No| C
    B -->|Action 1| End
    C -->|Decision 2| D
    D -->|Yes| End
    D -->|No| A
```

### Pi chart

```
pie title Accuracy
    "Dogs" : 386
    "Cats" : 85
    "Rats" : 15
```

## User Journey

```
---
title: My working day
---
journey
    section Go to work
        Make tea: 5: Me
        Go upstairs: 3: Me
        Do work: 1: Me, Cat
    section Go home
        Go downstairs: 5: Me
        Sit down: 5: Me
```

## Class Diagram

```
---
title: Animal example
---
classDiagram
    note "From Duck till Zebra"
    Animal <|-- Duck
    note for Duck "can fly\n-can swim\n-can dive\n-can help in debugging"
    Animal <|-- Fish
    Animal <|-- Zebra
    Animal : +int age
    Animal : +String gender
    Animal: +isMammal()
    Animal: +mate()
    class Duck{
        +String beakColor
        +swim()
        +quack()
    }
    class Fish{
        -int sizeInFeet
        -canEat()
    }
    class Zebra{
        +bool is_wild
        +run()
    }
```

## Sequence Diagram

```
---
title: Sequence Diagram Example
---
sequenceDiagram
```

```
Consumer-->API: Book something
API-->BookingService: Start booking process
break when the booking process fails
    API-->Consumer: show failure
end
API-->BillingService: Start billing process
```

---

1. This is a footnote. [←](#)
2. This is another footnote. [←](#)