

## Defining an Anchor

**anchor-name:** `--my-anchor`

Creates a named anchor point

## Positioning and Sizing

**position-anchor:** `--anchor-name`

Sets default anchor for element

**position-area:** `top bottom center`  
`span-all`

Placement in the 3×3 containing block

**justify-self:** `anchor-center`

Inline-axis center on the anchor

**Also on:** `align-self justify-items`  
`align-items place-self`  
`place-items`

## Fallback Positioning

**position-try:** `<fallbacks> <order>`

**position-try-fallbacks:** `flip-block`  
`flip-inline`  
`flip-start`

**position-try-order:** `normal`  
`most-height`  
`most-width`

**position-visibility:** `always`  
`anchors-visible`  
`no-overflow`

Hide or keep anchored element when anchor is out of view

## Function Values

**Position:** `anchor(top)`  
`anchor(--name top)`  
`inset/top/left/right/bottom`

**Size:** `anchor-size(width)`  
`anchor-size(block)`  
`width/height/inline-size/block-size`

## Scope

**anchor-scope:** `none` `all` `--name`  
Limits anchor visibility to subtree

## Position Area Example

```
.trigger {
  anchor-name: --btn;
}

.tooltip {
  position: absolute;
  position-anchor: --btn;
  position-area: top;
  justify-self: anchor-center;
}
```

## Position Area Grid

|             |               |              |
|-------------|---------------|--------------|
| left top    | center top    | right top    |
| left center | center/anchor | right center |
| left bottom | center bottom | right bottom |

## Fallback

default → **tip** → flip-inline

`btn` `tip` `btn` `tip` `btn`

flip-block

## Function Example

```
.tooltip {
  position: absolute;
  position-anchor: --btn;
  bottom: anchor(top);
  width: anchor-size(width);
}
```

## anchor() Sides

top

left **center/anchor** right

bottom