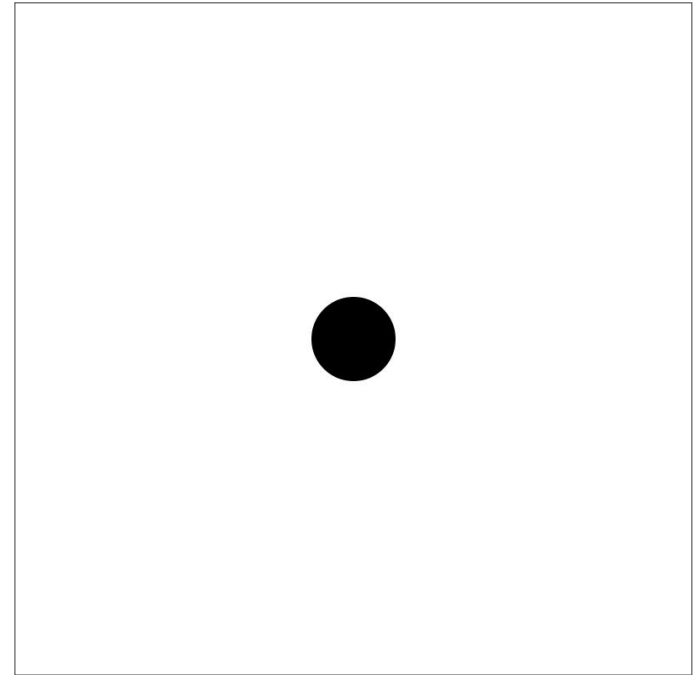


# BEDINGUNGEN

Beispiel

```
if (mouseX < width/2)
{
    fill (255, 0, 0);
}
ellipse (width/2, height/2, 100, 100);
```

Wenn die x-Position der Maus kleiner ist als die Hälfte der Fenstergröße, dann wird die Füllfarbe rot.



# BEDINGUNGEN

Allgemeiner Aufbau


```
if (Bedingung)
{
    // Inhalt
}
ellipse (width/2, height/2, 100, 100);
```

Wenn die Bedingung wahr ist, wird der Inhalt ausgeführt.

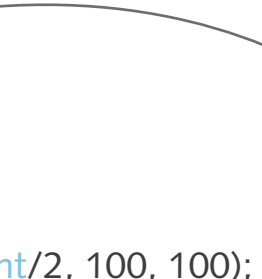
Wenn die Bedingung falsch ist, wird der Block übersprungen.

Der Inhalt einer Bedingung wird mit { und } gekennzeichnet.

```
if (Bedingung wahr)
{
    // Inhalt
}
ellipse (width/2, height/2, 100, 100);
```



```
if (Bedingung falsch)
{
    // Inhalt
}
ellipse (width/2, height/2, 100, 100);
```



# BEDINGUNGEN

Allgemeiner Aufbau

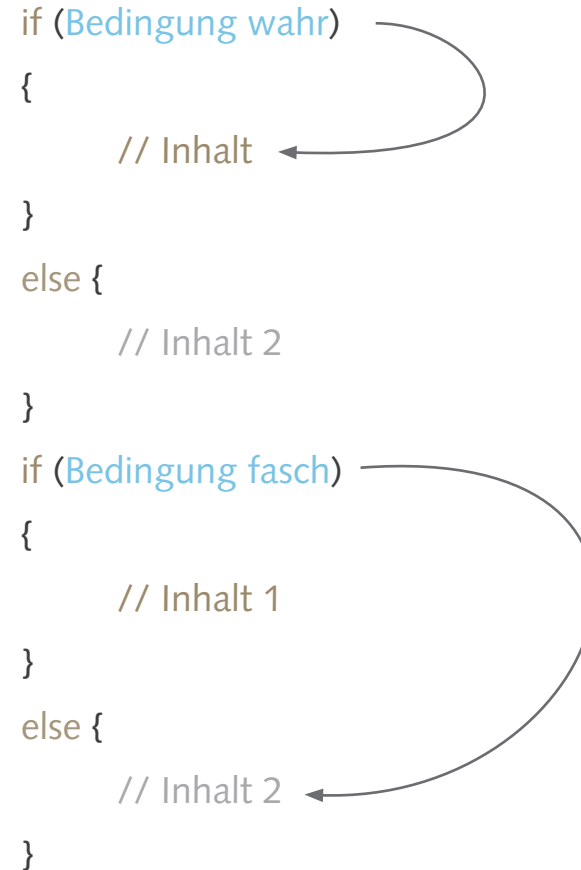
```
if (Bedingung)
{
    // Inhalt 1
}
else
{
    // Inhalt 2
}
```

```
ellipse (width/2, height/2, 100, 100);
```

Wenn die Bedingung wahr ist, wird der **Inhalt 1** ausgeführt.

Wenn die Bedingung falsch ist, wird der **Inhalt 2** ausgeführt.

```
if (Bedingung wahr)
{
    // Inhalt
}
else {
    // Inhalt 2
}
if (Bedingung falsch)
{
    // Inhalt 1
}
else {
    // Inhalt 2
}
```



# OPERATOREN

## Vergleichsoperatoren

<	kleiner als
>	größer als
==	gleich
<=	kleiner gleich
>=	größer gleich
!=	ungleich

# BEDINGUNGEN

Allgemeiner Aufbau

```
if (Bedingung) {  
    // Inhalt 1  
}  
else if (Bedingung) {  
    // Inhalt 2  
}  
else if (Bedingung) {  
    // Inhalt 3  
}  
  
    (...)  
else {  
    // Inhalt X  
}
```

# OPERATOREN

## Logische Operatoren

&&      und  
||        oder  
!         nicht

## Beispiel

```
if (mouseX > width/3 && mouseX < width*2/3)  
{  
    fill (255, 0, 0);  
}
```

```
if (mouseX < width/3 || mouseX > width*2/3)  
{  
    fill (255, 0, 0);  
}
```