

## Inhaltsverzeichnis

|                          |          |
|--------------------------|----------|
| <b>Eine Uhr</b>          | <b>2</b> |
| <b>Anforderungen</b>     | <b>2</b> |
| <b>Media - Dateien</b>   | <b>2</b> |
| <b>Vorgehen</b>          | <b>2</b> |
| <b>Pygame Grundlagen</b> | <b>3</b> |
| Das Koordinatensystem    | 3        |
| Position                 | 3        |
| Vector2                  | 4        |
| Farben                   | 4        |

## Eine Uhr



### Anforderungen

- Uhr mit rundem Zifferblatt und 12 Stunden Anzeige
- Sekunden-, Minuten- und Stundenzeiger
- Links unten Datum im Format TT.MM.JJJJ
- Rechts unten Zeit im Format HH:MM:SS

### Media - Dateien

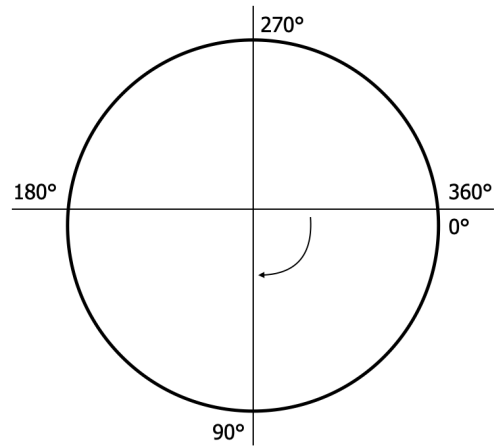
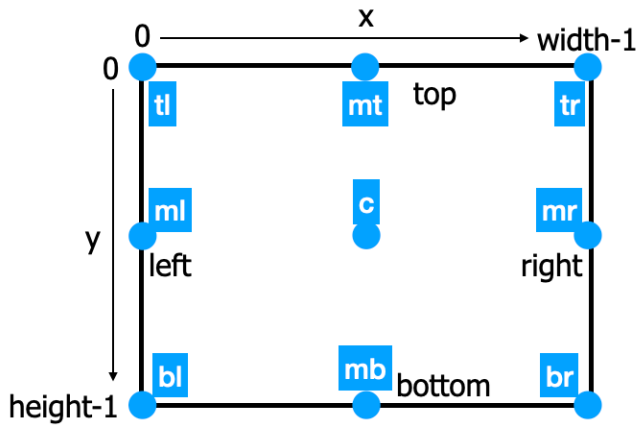
keine

### Vorgehen

- Projekt aus Vorlage erstellen. Es wird nur mygame.py benötigt.
- Keine Media - Dateien notwendig.
- Erstellen eines fixen Hintergrundes (self.background) mit dem äusseren Kreis.
- Minutenstriche und Ziffern.
- Sekundenzeiger, der jede Sekunde springt.
- Minutenzähler, der sich kontinuierlich bewegt.
- AUFGABE: Erstelle den Stundenzeiger. Kann auch der Sekundenzeiger kontinuierlich bewegt werden?
- Label für Datum links unten
- AUFGABE: Erstelle die Zeitanzeige rechts unten.

## Pygame Grundlagen

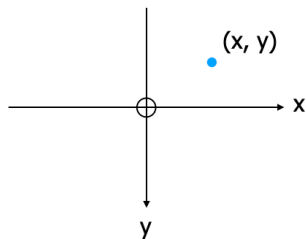
### Das Koordinatensystem



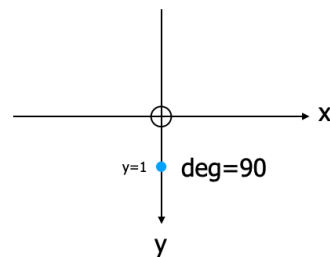
### Position

Gibt eine Position als x- und y-Koordinate an. Zusätzlich stehen noch weitere Optionen zur Verfügung.

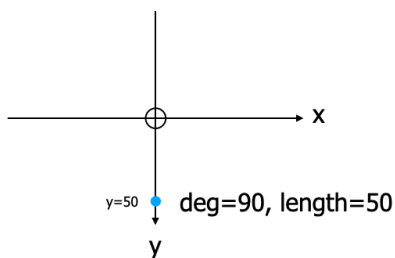
`pos = pgf.Position(x, y)`



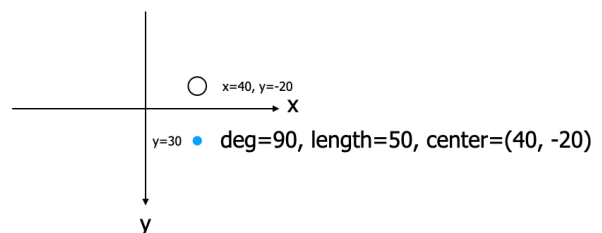
`pos = pgf.Position.from_deg(deg=90)`



`pos = pgf.Position.from_deg(deg=90, length=50)`



`pos = pgf.Position.from_deg(deg=90, length=50, center=(40, -20))`



Die Eigenschaften der Position können abgefragt oder verändert werden.

```
x = pos.x,  
y = pos.y  
x, y = pos  
laenge = pos.length  
winkel = pos.deg  
winkel = pos.rad
```

```
pos.x = 5  
pos.y = 10  
pos.length = 50  
pos.deg = 45  
pos.rad =  $\pi / 2$ 
```

Im Prinzip ist **Position** ein **Vector2** und stellt alle Operationen von Vector2 zur Verfügung.

### Vector2

Das ist eine zweidimensionaler Vektor, der von Pygame zur Verfügung gestellt wird.

<https://www.pygame.org/docs/ref/math.html#pygame.math.Vector2>

### Farben

Farben können als RGB - Werte oder Strings übergeben werden.

```
color = (r, g, b)          # Die Komponenten bestehen aus Werten zwischen 0 und 255  
color = (r, g, b, a)       # a ist der Alpha - Wert (Transparenz, 100=volle Farbe)  
color = '#F0F8FF'         # RGB im Hex - Format  
color = 'blue'
```

[https://www.pygame.org/docs/ref/color\\_list.html](https://www.pygame.org/docs/ref/color_list.html)

<https://www.pygame.org/docs/ref/color.html>

[https://github.com/pygame/pygame/blob/main/src\\_py/colordict.py](https://github.com/pygame/pygame/blob/main/src_py/colordict.py)

<https://www.webucator.com/article/python-color-constants-module/>