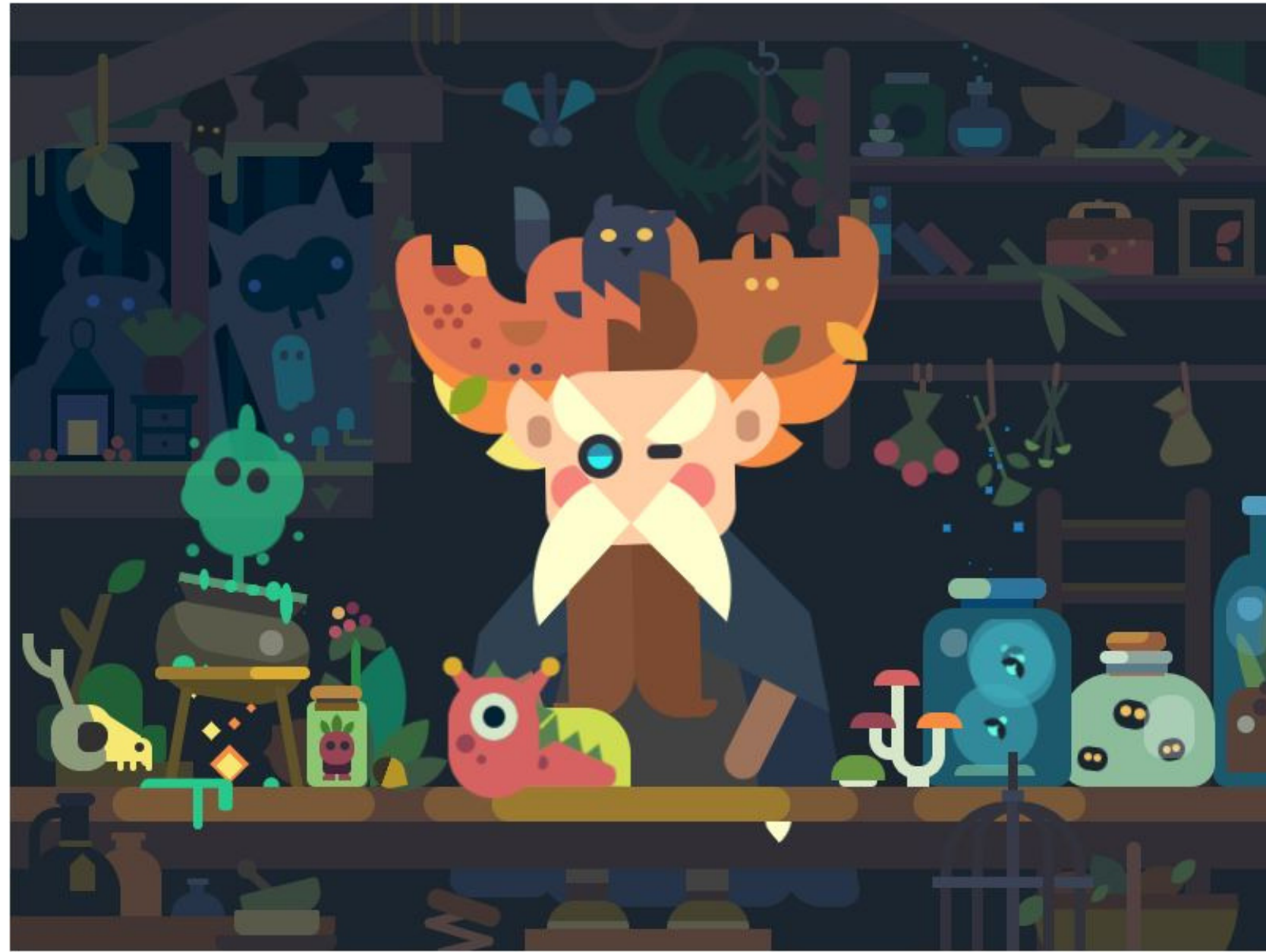


# Les animations CSS3



# la propriété "transform"

La propriété transform permet d'appliquer une transformation 2D ou 3D a un élément.

Exemple de valeur possible avec la propriété transform :

**transform:** translateX(x)

**transform:** translateY(y)

**transform:** translate(x,y)

**transform:** scale(x,y)

**transform:** rotate(angle)

**transform:** skew(angle)

```
.square{  
  width: 100px;  
  height: 100px;  
  background-color: red;  
  border: 1px solid black;  
  transform: translateY(50px);  
}
```

# la propriété "transition"

La propriété transition permet d'effectuer une animation entre un état de début et de fin.

**transition-property** : transform // all // height // margin // font-size



**transition-duration** : 0.5s // 1s // 500ms

**transition-timing-function** : ease // linear // step-end // step(3,end)

**transition-delay** : 0.5s // 1s // 500ms

**transition**:<property> <duration> <timing-function> <delay>

# Exemple d'utilisation

```
.square{  
  width: 100px;  
  height: 100px;  
  background-color:  red;  
  border: 1px solid  black;  
  transition: width 0.5s ease-in-out;  
}  
  
.square:hover{  
  width: 200px;  
}
```

# Les animations avec @keyframes

L'utilisation de keyframes permet de gérer les animations avec plus de précision qu'avec les transitions, il est important de définir préalablement le nom de l'animation grâce à la propriété "animation".

```
div{
  animation-duration:5s;
  animation-name: squaresize;
}

@keyframes squaresize {
  from{
    width:50px;
    height:50px;
  }

  to{
    width:100px;
    height:100px;
  }
}
```

```
@keyframes squaresize {
  0%{
    width:100px;
    height:100px;
  }

  25%{
    width:50px;
    height:50px;
  }

  50%{
    width:100px;
    height:100px;
  }

  100%{
    width:75px;
    height:75px;
    transform:translateX(100px);
  }
}
```