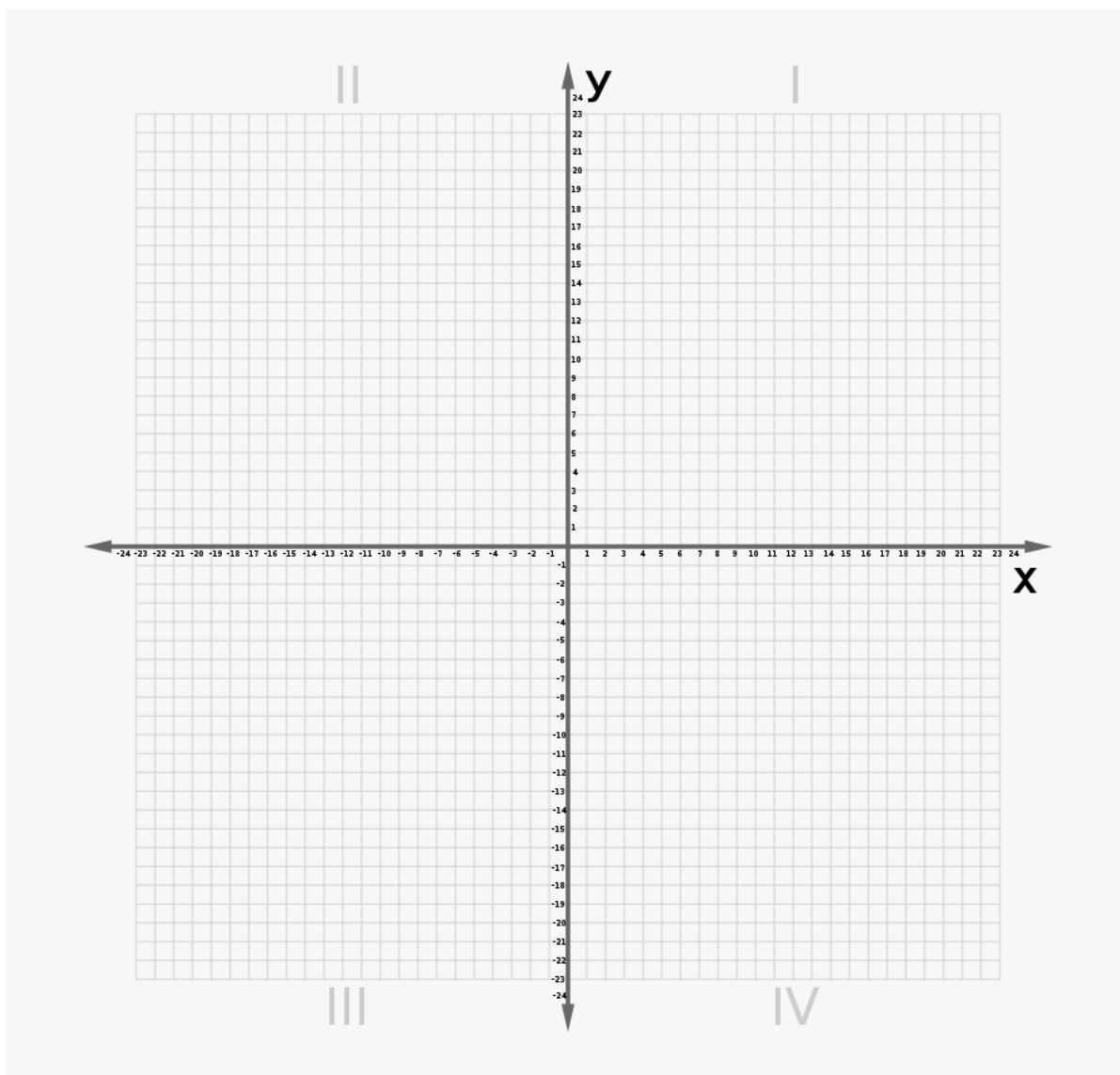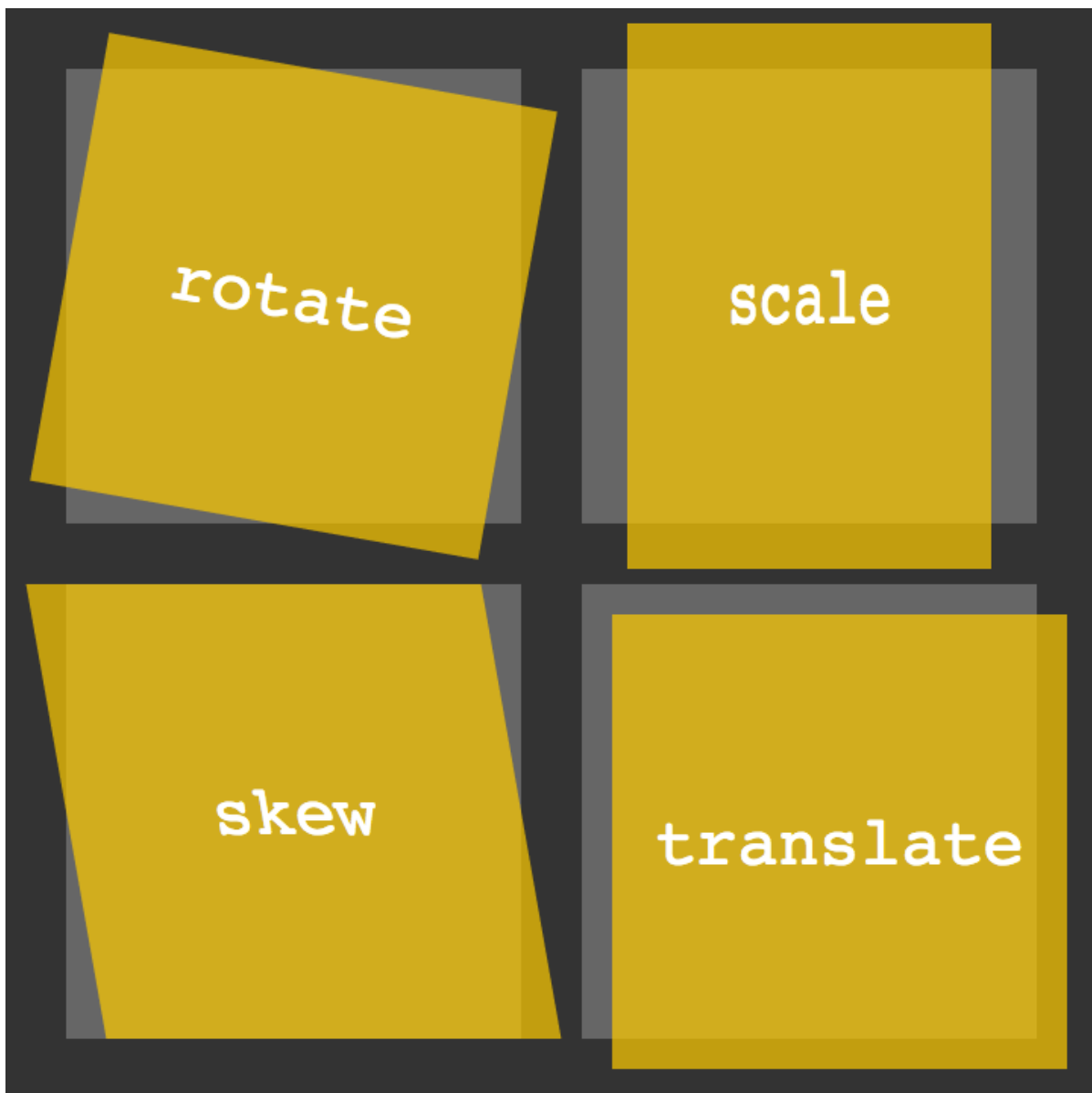# 49. 2D transforms

Reja:

▼ **2D**

💡 **2D -** geometrik model bo'lib elementimizni joylashuvini aniqlash uchun 2ta nuqta yetarli bo'ladi

▼ **Transform**

💡 **Transform -** veb sahifamizdagi elementlarni aylantirish, siljitish, o'lchovini o'zgartirish, va og'dirish uchun ishlatilinadigan CSS xossasi. Uning 2 ko'rinishi mavjud 2D va 3D

Quyidagi "method"lari mavjud

▼ rotate()

> 💡 **Rotate -** elementimizni soat strelkasi yoki unga teskari bo'lgan yo'nalishda burish (aylantirish) uchun ishlatiladi. Musbat qiymat soat strelkasi boy'icha, manfiy esa teskari yo'nalishda buradi

Umumiy ko'rinishi

```
selector-nomi {
  transform: rotate(daraja);
}
```

Soat strelkasi bo'yicha burish

```
<style>
  .box {
    width: 500px;
    height: 500px;
    background-color: red;
    transform: rotate(40deg);
  }
</style>

--------------------
<div class="box">40 darajaga burilgan quti</div>
```

Soat strelkasi teskari ravishda burish

```
<style>
  .box {
    width: 500px;
    height: 500px;
    background-color: red;
    transform: rotate(-40deg);
  }
</style>

--------------------
<div class="box">40 darajaga burilgan quti</div>
```

💡 **Transform origin -** o'zgartirish (transform) amal qiladigan nuqta

transform-origin

The transform-origin CSS property sets the origin for an element's transformations. The transformation origin is the point around which a transformation is applied. For example, the transformation origin of the function is the center of rotation. This property is applied by first translating the element by the value of

🔗 https://developer.mozilla.org/en-US/docs/Web/CSS/transform-origin

```
<style>
  .parent-box {
    margin-top: 100px;
    border: 5px solid blue;
  }
  .box {
    height: 100px;
    background-color: red;
    transform: rotate(45deg);
    transform-origin: top right;
  }
</style>

---------------
<div class="parent-box">
  <div class="box">40 darajaga burilgan quti</div>
</div>
```
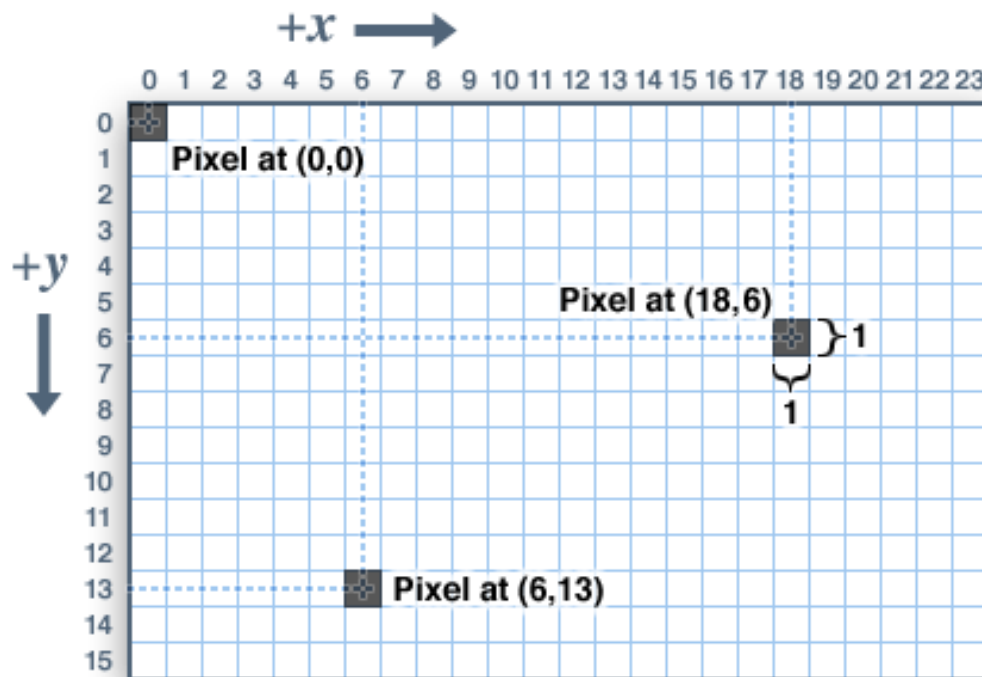
▼ translate()

> 💡 **Translate -** elementimizni xozirgi joyidan siljitish (surish) uchun ishlatiladi. Musbat qiymat o'ng va pastga qarab, manfiy esa chap va yuqoriga qarab siljitish uchun ishlatiladi

Umumiy ko'rinishi

```
selector-nomi {
  transform: translate(x-qiymat, y-qiymat);
}
```

Misol

```
<style>
  .box {
    width: 500px;
    height: 500px;
    background-color: red;
    transform: translate(20px, 40px);
  }
</style>

--------------------
<div class="box">20px o'ngga, 40px pastga surish</div>
```

💡 "**Translate**"ning yana **translateX()** va **translateY()** ko'rinishlari ham mavjud

▼ **scale()**

💡 **Scale -** elementimizni o'lchovini oshirish yoki kamaytirish uchun ishlatiladi. 0 bilan 1 orasida berilgan qiymat uning o'lchovini kamaytiradi, 1dan katta qiymat esa o'lchovini oshiradi

Umumiy ko'rinishi

```css
selector-nomi {
  transform: scale(x-qiymat, y-qiymat);
}
```

Element o'lchovini kattalashtirishga misol

```html
<style>
  .box {
    width: 500px;
    height: 500px;
    background-color: red;
    transform: scale(2,3);
  }
</style>

--------------------
<div class="box">kengligini 2 barobar, balandligini 3 barobarga oshirish</div>
```

Element o'lchovini kichiklashtirishga misol

```
<style>
  .box {
    width: 500px;
    height: 500px;
    background-color: red;
    transform: scale(0.5, 0.25);
  }
</style>

-------------------
<div class="box">kengligini 2 barobar, balandligini 4 barobarga kamaytirish</div>
```
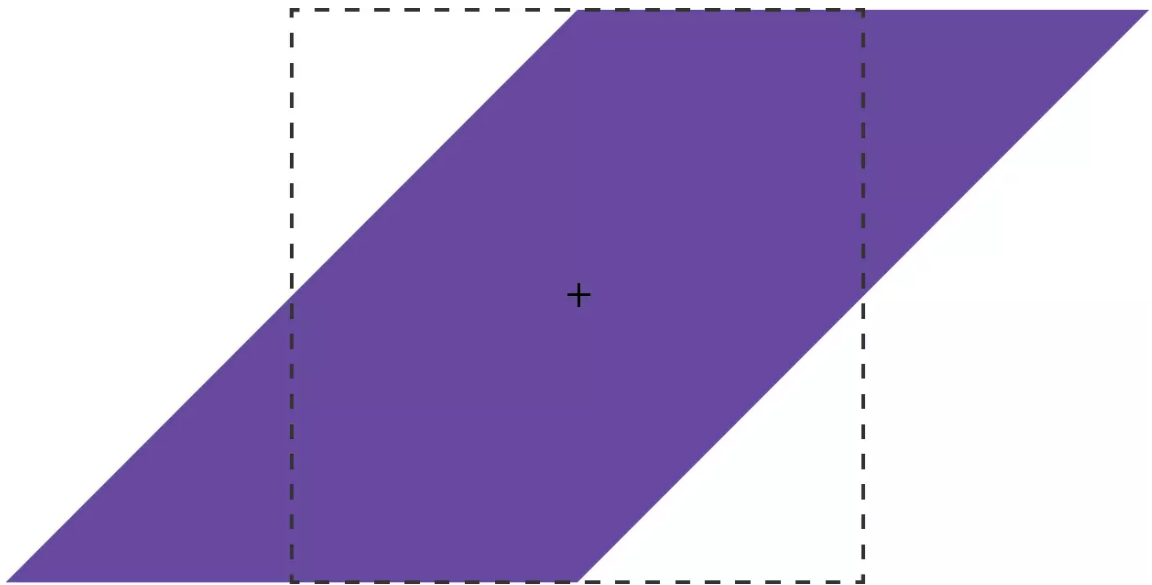
💡 **"Scale"**ning yana **scaleX()** va **scaleY()** ko'rinishlari ham mavjud

▼ skew()

💡 **Skew -** elementimizni ma'lum bir darajaga (degree)ga og'dirish uchun ishlatiladi. Manfiy qiymat og'ish yo'nalishini teskari tomonga o'zgartiradi

Umumiy ko'rinishi

```
selector-nomi {
  transform: skew(x-qiymat, y-qiymat);
}
```

Element o'lchovini kattalashtirishga misol

```
<style>
  .box {
    width: 500px;
    height: 500px;
    background-color: red;
    transform: skew(20deg,30deg);
  }
</style>

--------------------
<div class="box">x o'qi bo'yicha 20 darajaga, y o'qi bo'yicha esa 30 darajaga og'dirish</div>
```

💡 "**Skew**"ning yana **skewX()** va **skewY()** ko'rinishlari ham mavjud

References:

https://infinityisreallybig.com/2019/05/14/points-coordinates-skew-coordinates-and-graphs/

https://www.htmldog.com/references/css/properties/transform/

https://learn.adafruit.com/adafruit-gfx-graphics-library/coordinate-system-and-units

https://idg.net.ua/blog/uchebnik-css/ispolzovanie-css/transform/scale

https://www.sitepoint.com/how-to-use-2d-transformation-functions-in-css/