

# 数据可视化 CSS3

刘军 liujun

# 目录

## content



**1** 邂逅数据可视化

**2** 2D动画

**3** 3D动画

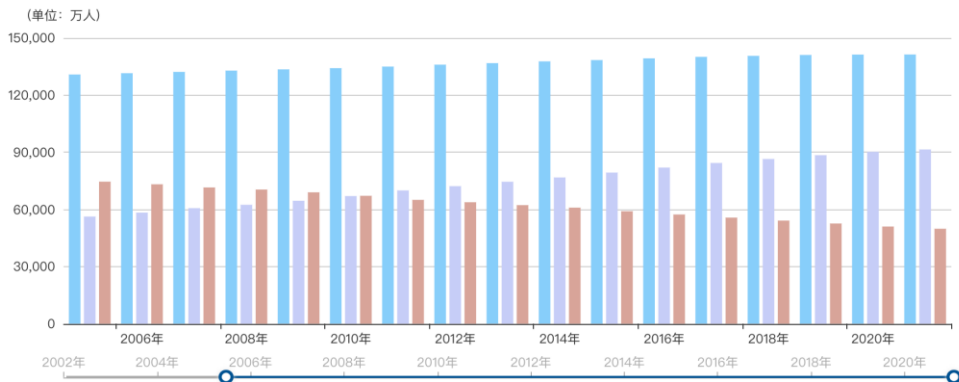
**4** 2.5D和3D动画实战

**5** 动画的优化

# 邂逅数据可视化

- **数据可视化**（英语：Data visualization），主要旨在借助于**图形化手段**，清晰有效地**传达与沟通信息**。
  - 为了清晰有效地传递信息，**数据可视化通常使用柱状图、折线图、饼图、玫瑰图、散点图等图形来传递信息**。
  - 也可以使用**点、线、面、地图**来对数字数据进行编码展示，以便在视觉上快速传达关键信息。
  - 可视化可以帮助用户**分析和推理**数据，让复杂的数据更容易理解和使用，有利于做出决策。

序号	统计时间	年末人口 (万人)	城镇人口 (万人)	乡村人口 (万人)
1	2005年	130756	56212	74544
2	2006年	131448	58288	73160
3	2007年	132129	60633	71496
4	2008年	132802	62403	70399
5	2009年	133450	64512	68938



○ 全选 
 ■ 年末人口 (万人) 
 ■ 城镇人口 (万人) 
 ■ 乡村人口 (万人) 
 ○ 城镇人口占总人口比重 (%)



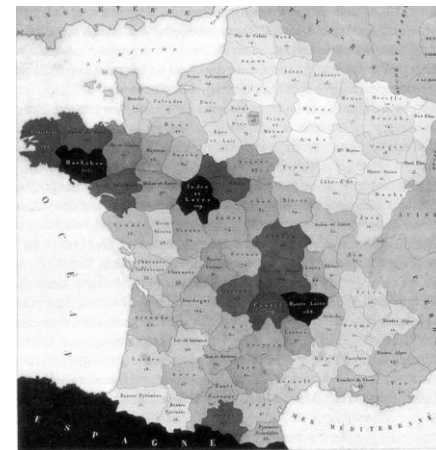
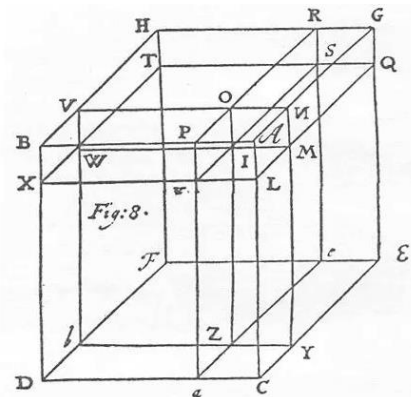
# 可视化-萌芽阶段

## ■ 17世纪以前

- 早在17世纪以前，可视化就开始萌芽了，其中**最早的地图**在公元前6200年于土耳其地区出现。
- 现代考古发现**我国最早的地图实物**，是出土于甘肃天水放马滩战国墓地一号墓中的《放马滩地图》

## ■ 17-19世纪

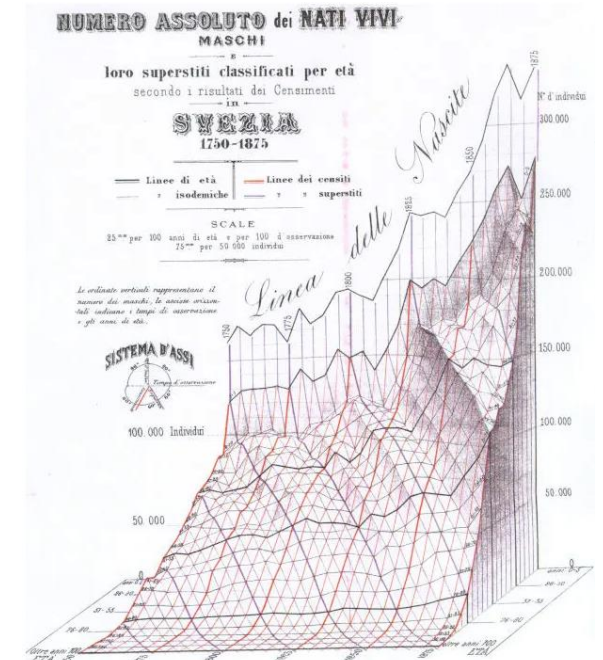
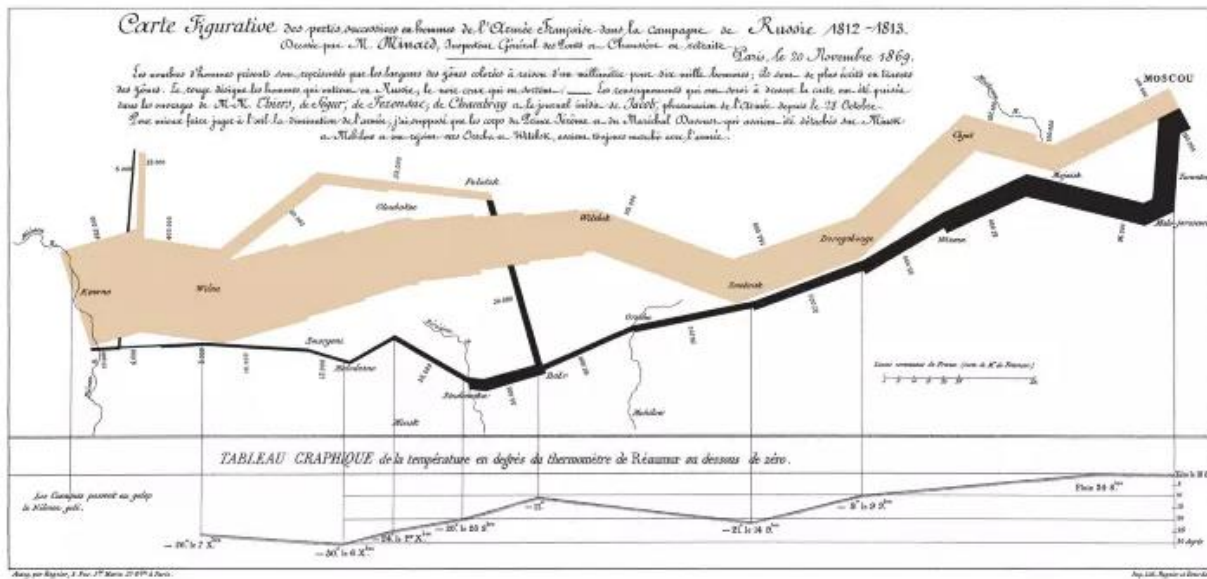
- 17世纪末随着**几何兴起、坐标系、以及人口统计学开端**，人类开始了可视化思考的新模式，从此**标记可视化的开端**。
- 1800-1849年：随着工艺设计的完善，统计图形爆炸性增长，**包括柱状图, 饼图, 直方图, 折线图等**。
- 1826年，查尔斯·杜品发明了使用连续黑白底纹来显示法国**识字分布**，这可能是第一张现代形式主题统计地图。



# 可视化-黄金阶段

## ■ 19世纪中

- 1850-1899: 人们开始认识到数字信息对社会计划, 工业化, 商业和运输的重要性, 此时**统计理论**开始诞生。
- 1869年查尔斯·约瑟夫·米纳德, 发布的拿破仑对1812年俄罗斯东征事件流图, **被誉为有史以来最好的数据可视化**。
  - ✓ 他的流图呈现了拿破仑军队的位置和行军方向、军队汇集、分散和重聚的时间和地点等信息。
- 1879年 Luigi Perozzo 绘制立体图 (三维人口金字塔)。**标记着可视化开始进入了三维立体图**。

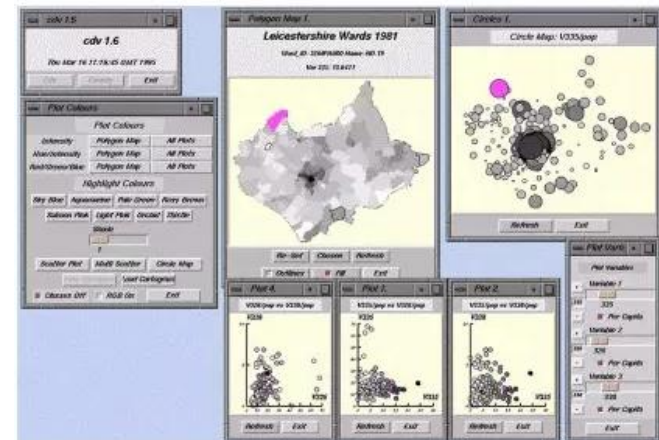
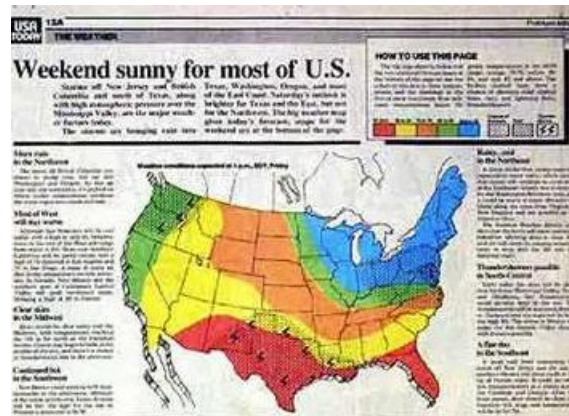
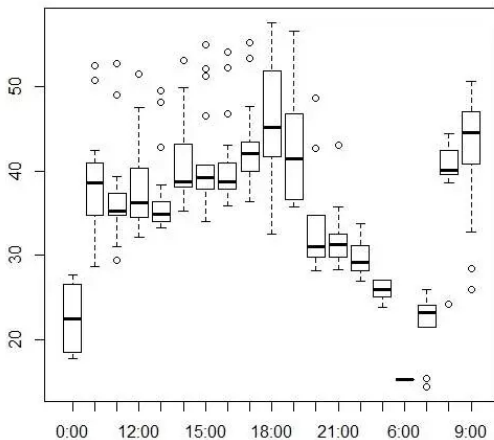




# 可视化-重生阶段

## ■ 20世纪

- 1950-1974年：引领这次大潮的，首先是一个划时代的事件——**计算机的诞生**。
- **计算机的出现彻底地改变了数据分析工作，计算机高分辨率和交互式的图形分析，提供了手绘时代无法实现的表现能力。**
- 随着统计应用的发展，**数理统计把数据可视化变成了一门科学**（如：计算机图形学、统计学、分析学），并运用到各行各业。
- 1969年 John W. Tukey 在探索数据分析的图形时，发明箱型图。
- 1982年乔治·罗里克（George Rorick）绘制彩色天气图开创了报纸上的**彩色信息图形时代**。
- 1996年 Jason Dykes 发明了制图**工具：一种地图可视化工具包**，可以实时查看数据的图形工具。



# 可视化 - 分析学阶段

## ■ 2004年至今

- 以前可视化难以应对**海量、高维、多源的动态数据的分析**，进入21世纪，随着计算机的升级，对于以前难以应对数据，可以借用计算机来综合可视化、图形学、数据挖掘理论与方法来研究新的科学理论模型。通过这种模型来辅助用户从海量、复杂、矛盾的数据中快速挖掘出有用的数据，做出有效决策，这门新兴学科称为**可视化分析学**。
- 可视化分析现在已大量应用在地图、物流、电力、水利、环保、交通、医学、监控、预警等领域。可视化分析降低了数据理解的难度，突破了常规统计分析的局限性。如下交通拥挤分析图。随着大数据的应用，如今可视化开发也变得越来越重要了。



- 随着近几年大数据的快速发展，数据可视化技术也迅速被普及。目前数据可视化的应用非常广：
  - 如淘宝双十一活动时，借助于数据可视化展示公司实时交易数额，并可以实时动态观察。
  - 交管部门可实现对交通形态、卡口数据统计、违章分析、警力部署、出警分析、行车轨迹分析等智能交通大数据分析。
  - 企业各层可以借助数据可视化工具，可以直接在手机等设备上远程查看业务运营数据状况和关键指标。
  - 医院可以利用数据可视化工具，对医疗卫生数据进行可视化分析和研究应用，进而获取医疗卫生数据隐藏的价值。
  - 等等

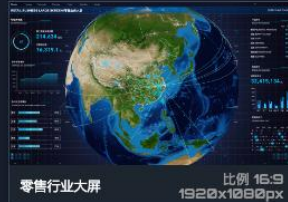




# 数据可视化-应用

data.aliyun.com/create\_screen?workspaceId=182058&project\_id=297091&type=pc

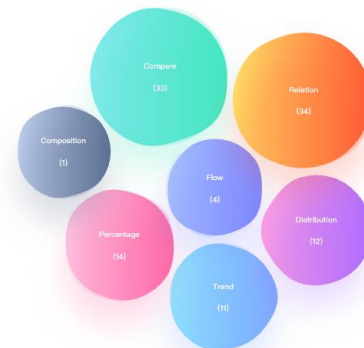
取消创建



# 可视化-解决方案

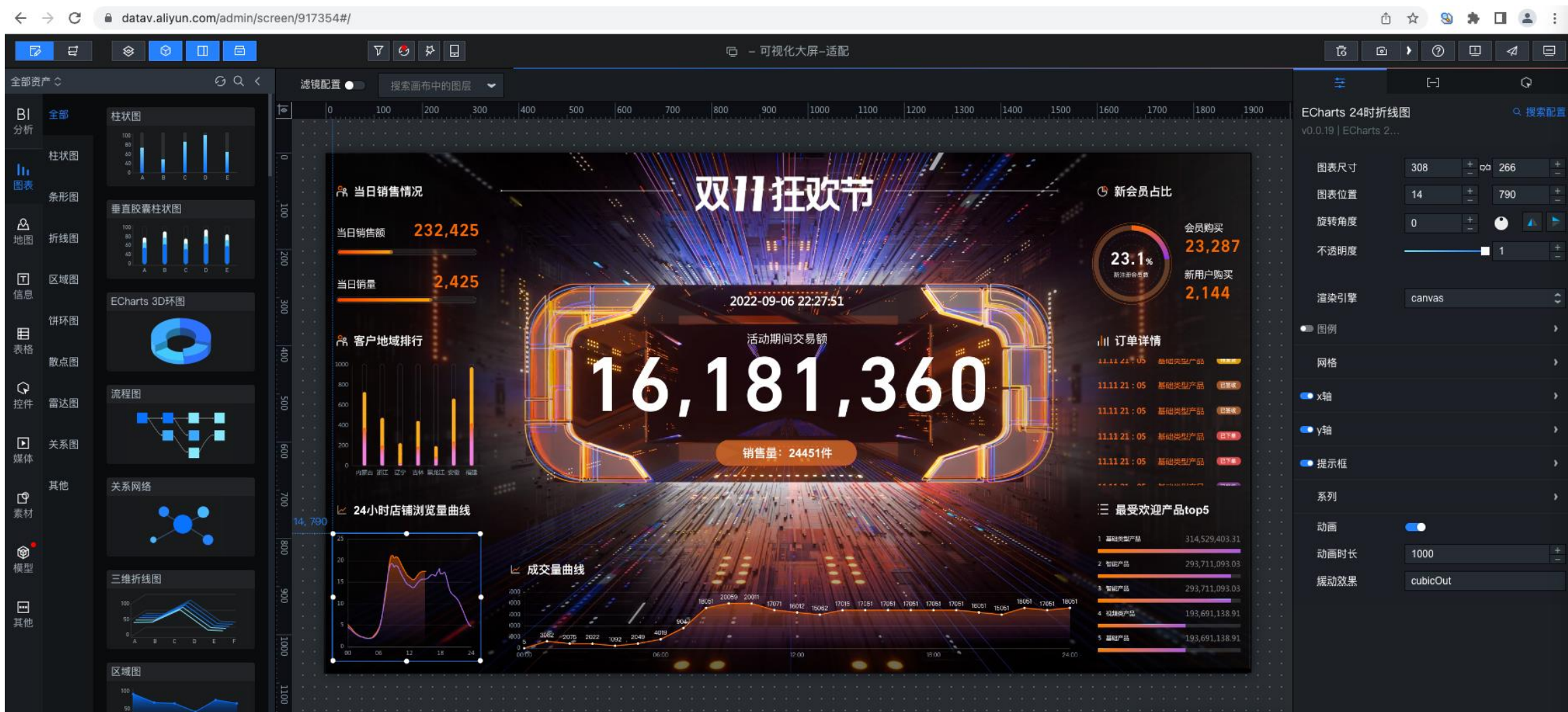
## ■ 前端可视化技术

- 底层图形引擎： Skia 、 OpenGL 等。
- W3C提供： CSS3、 Canvas、 SVG、 WebGL。
- 第三方的可视化库： ZRender、 Echarts、 AntV 、 Highcharts、 D3.js 、 Three.js 和 百度地图、 高德地图等等。
- 低代码可视化平台： 阿里云 (DataV) 、 腾讯云图、 网易有数 (EasyScreen) 、 帆软等。





# 低代码平台 阿里云 (DataV)



# 2D动画 - transform

- CSS3 transform属性允许你**旋转，缩放，倾斜或平移**给定元素。
- Transform是形变的意思（通常也叫变换），transformer就是变形金刚
- 常见的函数transform function有：
  - **平移**：translate(x, y)
  - **缩放**：scale(x, y)
  - **旋转**：rotate(deg)
  - **倾斜**：skew(deg, deg)
- 通过上面的几个函数，我们就可以改变某个元素的2D形变



## ■ CSS3 transform属性允许你在二维或三维空间中直观地变换元素。

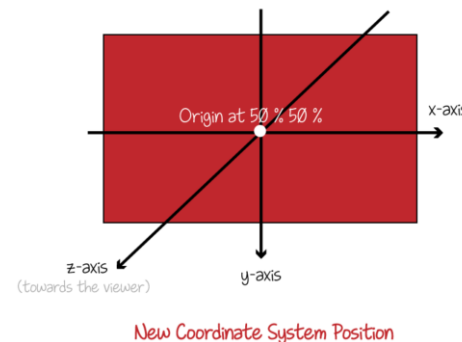
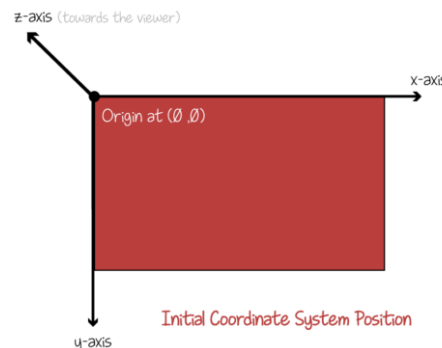
- transform属性会转换元素的坐标系，使元素在空间中转换。
- 用transform属性变换的元素会受transform-origin属性值的影响，该属性用于指定形变的原点。

## ■ 元素的坐标系

- CSS 中的每个元素都有一个坐标系，其原点位于元素的左上角，左上角这被称为初始坐标系。
- 用transform时，坐标系的原点默认会移动到元素的中心。
- 因为transform-origin属性的默认值为50% 50%，即该原点将会作为变换元素的中心点。
- 用transform属性旋转或倾斜元素，会变换或倾斜元素的坐标系。并且该元素所有后续变换都将基于新坐标系的变换。
- 因此，transform属性中变换函数的顺序非常重要——不同的顺序会导致不同的变换结果。

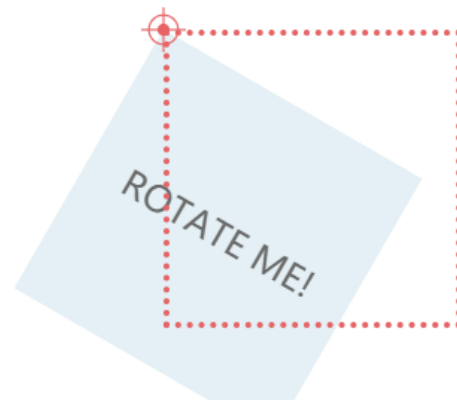
## ■ 例如：

- 如果将一个元素绕 y 轴旋转 90 度，那么它的 x 轴将指向屏幕内部，即远离你。
  - ✓ 此时如再沿着 x 轴平移，元素不会向右移动，它会向内远离我们。
- 因此，要注意编写转换函数的顺序，其中transform属性中的第一个函数将首先应用，最后一个函数将最后应用。



# transform-origin

- **transform-origin: 变形的原点 (即坐标系0, 0点)**
- **一个值:**
  - 设置 x轴 的原点, y轴为默认值 50%。
- **两个值:**
  - 设置 x轴 和 y轴 的原点
- **三个值:**
  - 设置 x轴、 y轴 和 z轴 的原点
- **必须是<length>, <percentage>, 或 left, center, right, top, bottom关键字中的一个**
  - left, center, right, top, bottom关键字
  - length: 从左上角开始计算
  - 百分比: 参考元素本身大小



```
transform-origin: top left;
```



# 3D动画 - transform

- CSS3 transform属性不但允许你进行2D的**旋转**，**缩放或平移**指定的元素，还支持**3D变换元素**。
- 常见的函数transform function有：
  - **平移**: translate3d(tx, ty, tz)
    - ✓ translateX(tx)、translateY(ty)、translateZ(tz)
  - **缩放**: scale3d(sx, sy, sz)
    - ✓ scaleX(sx)、scaleY(sy)、scaleZ(sz)、
  - **旋转**: rotate3d(x, y, z, a)
    - ✓ rotateX(x)、rotateY(y)、rotateZ(z)
- 通过上面的几个函数，我们可以改变某个元素的3D形变。
- 3D形变函数会创建一个合成层来启用GPU硬件加速，比如： translate3d、 translateZ、 scale3d 、 rotate3d ...

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/787046035126010002>