**1、定义画布**

首先我们现在html文件里面插入<canvas>标签，定义画布的尺寸，我这里定义画布的尺寸为800\*600像素。同时在内部样式表里面设置canvas的背景色（方便画图时观看）；

接下来的核心就是在原生JS环境下，绘制风车；通过JS DOM操作方法获取到canvas元素对象，并通过getContex（"2d"）获取2D绘图上下文，通过这个方法就像是要告诉浏览器“我们要在这个画布上绘制2d图形”；

**2、绘制风车底座**

风车的底座的几何图形看似就像一个细长细长的梯形，我们可以画出一个梯形出来，然后填充颜色，这里为了达到相对较好的效果，使用了颜色渐变填充的方法

**3、绘制叶子**

接下来的部分将是这个动画中最关键的地方，首先我们分析一下叶子的结构，三片叶子夹角为120°，而且每片叶子的形状是相同的；他们有一个圆心，你心中或许也有疑问，先画圆心还是先画叶子？叶子的形状应该怎么画呢？叶子可不可复制粘贴呢？答案当然是可以的，Let's do it！

思路分析：

1）、由于3片叶子的形状是一模一样，我们只需要画出一片叶子，第二第三片叶子直接copy就行了，聪明的我们是不是应该给这个叶子的画法封装一个函数呀？就叫它bind(  )函数吧！！每次调用它就可以了！哎！你们TM太机智了

2）、三片叶子有一个圆心，绘制叶子的时候为了方便取坐标值，我们将圆心从画图的左上角移动梯形顶部，这样我们绘制叶子会方便很多！这里使用了translate（）方法，移动坐标系！

3）、最难的一点就是理解这里动画是怎么实现的，因为动画原理会影响到我们画叶子的文档结构：

首先我们先新建一个绘图环境，我们称它为环境1，我们在环境1上画完第一片叶子；然后在   第一个绘图环境前提下 　旋转120°新建第一个绘图环境2，再此基础上调用画叶子的函数bind(  )，绘制二片叶子;第三片叶子的绘制方法如法炮制，在环境2的基础上旋转120°，新建环境3，调用绘制叶子函数bind(  )画第三片叶子;

如果要实现动画，我们只需要旋转第一片叶子的绘图环境1，第二片叶子和第三片叶子都是参照环境1为基准画出来的，是不是也跟着动起来了呢？？        弹幕：666666

4）、最后就是一些基本的外观样式调试的啦！比如颜色渐变啊，透明度啊，之类的！

这里我声明了一个变量 var num = 0;，作为环境1旋转度数变化的一个参数。

**4、设置动画**

动画这部分就比较简单了，设置定时器，清除画布，调用函数