import { cn } from "@/lib/utils"

import { ElementType, ComponentPropsWithoutRef } from "react"

interface StarBorderProps<T extends ElementType> {

as?: T

color?: string

speed?: string

className?: string

children: React.ReactNode

}

export function StarBorder<T extends ElementType = "button">({

as,

className,

color,

speed = "6s",

children,

...props

}: StarBorderProps<T> & Omit<ComponentPropsWithoutRef<T>, keyof StarBorderProps<T>>) {

const Component = as || "button"

const defaultColor = color || "hsl(var(--foreground))"

return (

<Component

className={cn(

"relative inline-block py-[1px] overflow-hidden rounded-[20px]",

className

)}

{...props}

>

<div

className={cn(

"absolute w-[300%] h-[50%] bottom-[-11px] right-[-250%] rounded-full animate-star-movement-bottom z-0",

"opacity-20 dark:opacity-70"

)}

style={{

background: `radial-gradient(circle, ${defaultColor}, transparent 10%)`,

animationDuration: speed,

}}

/>

<div

className={cn(

"absolute w-[300%] h-[50%] top-[-10px] left-[-250%] rounded-full animate-star-movement-top z-0",

"opacity-20 dark:opacity-70"

)}

style={{

background: `radial-gradient(circle, ${defaultColor}, transparent 10%)`,

animationDuration: speed,

}}

/>

<div className={cn(

"relative z-1 border text-foreground text-center text-base py-4 px-6 rounded-[20px]",

"bg-gradient-to-b from-background/90 to-muted/90 border-border/40",

"dark:from-background dark:to-muted dark:border-border"

)}>

{children}

</div>

</Component>

)

}

/\*\* @type {import('tailwindcss').Config} \*/

module.exports = {

theme: {

extend: {

animation: {

'star-movement-bottom': 'star-movement-bottom linear infinite alternate',

'star-movement-top': 'star-movement-top linear infinite alternate',

},

keyframes: {

'star-movement-bottom': {

'0%': { transform: 'translate(0%, 0%)', opacity: '1' },

'100%': { transform: 'translate(-100%, 0%)', opacity: '0' },

},

'star-movement-top': {

'0%': { transform: 'translate(0%, 0%)', opacity: '1' },

'100%': { transform: 'translate(100%, 0%)', opacity: '0' },

},

},

},

},

}