

对象数组



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌



对象数组

```
// 1. 定义接口
interface Person {
    stuId: number
    name: string
    gender: string
    age: number
}

// 2. 基于接口构建对象
let p1: Person = {
    stuId: 1,
    name: '小丽',
    gender: '女',
    age: 12
}
```

对象

```
// 1. 定义接口
interface Person {
    stuId: number
    name: string
    gender: string
    age: number
}

// 2. 基于接口构建对象数组
let pArr: Person[] = [
    { stuId: 1, name: '小丽', gender: '女', age: 12 },
    { stuId: 2, name: '小王', gender: '男', age: 11 },
    { stuId: 3, name: '大强', gender: '男', age: 13 },
    { stuId: 4, name: '小张', gender: '男', age: 11 },
    { stuId: 5, name: '小美', gender: '女', age: 12 },
]
```

对象数组



总结

什么是对象数组？

```
// 1. 定义接口
interface Person {
  stuId: number
  name: string
  gender: string
  age: number
}

// 2. 基于接口构建对象数组
let pArr: Person[] = [
  { stuId: 1, name: '小丽', gender: '女', age: 12 },
  { stuId: 2, name: '小王', gender: '男', age: 11 },
  { stuId: 3, name: '大强', gender: '男', age: 13 },
  { stuId: 4, name: '小张', gender: '男', age: 11 },
  { stuId: 5, name: '小美', gender: '女', age: 12 },
]
```

通过下标即可访问，通过 for of 可以遍历

ForEach-渲染控制



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌

ForEach – 渲染控制

ForEach 可以基于数组的个数，渲染组件的个数。（简化代码）

语法: `ForEach(arr, (item, index) => {})`

```
@State titles:string[] = ['电子产品', '精品服饰', '母婴产品',  
'影音娱乐', '海外旅游']  
  
build() {  
  Column() {  
    ForEach(this.titles, (item: string, index) => {  
      Text(item)  
        .fontSize('24')  
        .fontWeight(700)  
        .fontColor(Color.Orange)  
        .width('100%')  
        .padding(15)  
    })  
  }  
}
```

电子产品

精品服饰

母婴产品

影音娱乐

海外旅游

51CTO

近200+自动驾驶数据集全面调研！一览如何数据闭环全流程

2024-01-31 09:59:43

MySQL Shell 8.0.32 for GreatSQL编译二进制包

2024-01-31 09:55:53

在Redis中如何实现分布式事务的一致性？

2024-01-31 09:54:51

如何 循环渲染 列表结构?

ForEach



总结

```
@State titles:string[] = ['电子产品', '精品服饰', '母婴产品',  
'影音娱乐', '海外旅游']  
  
build() {  
    Column() {  
        ForEach(this.titles, (item: string, index) => {  
            Text(item)  
                .fontSize('24')  
                .fontWeight(700)  
                .fontColor(Color.Orange)  
                .width('100%')  
                .padding(15)  
        })  
    }  
}
```

ForEach 案例 – 新闻列表

语法: `ForEach(arr, (item, index) => {})`

```
@State articles: Article[] = [
  {
    title: '近200+自动驾驶数据集全面调研！一览如何数据闭环全流程',
    createTime: '2024-01-31 09:59:43'
  },
  {
    title: 'MySQL Shell 8.0.32 for GreatSQL编译二进制包',
    createTime: '2024-01-31 09:55:53'
  },
  {
    title: '在Redis中如何实现分布式事务的一致性?',
    createTime: '2024-01-31 09:54:51'
  }
]
```

51CTO

近200+自动驾驶数据集全面调研！一览如何数据闭环全流程

2024-01-31 09:59:43

MySQL Shell 8.0.32 for GreatSQL编译二进制包

2024-01-31 09:55:53

在Redis中如何实现分布式事务的一致性?

2024-01-31 09:54:51

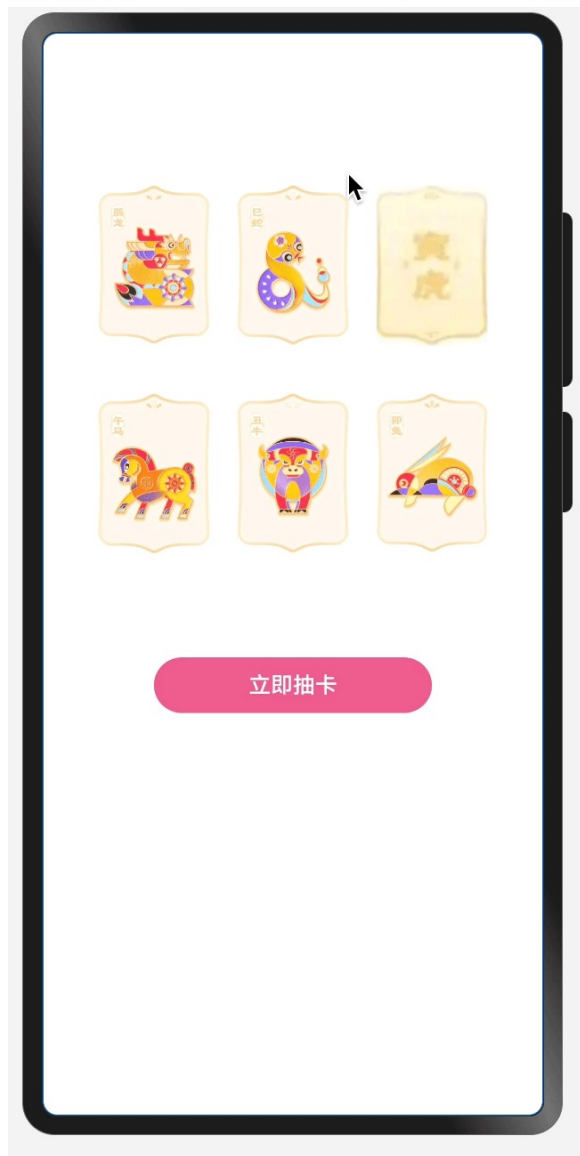
生肖抽奖卡



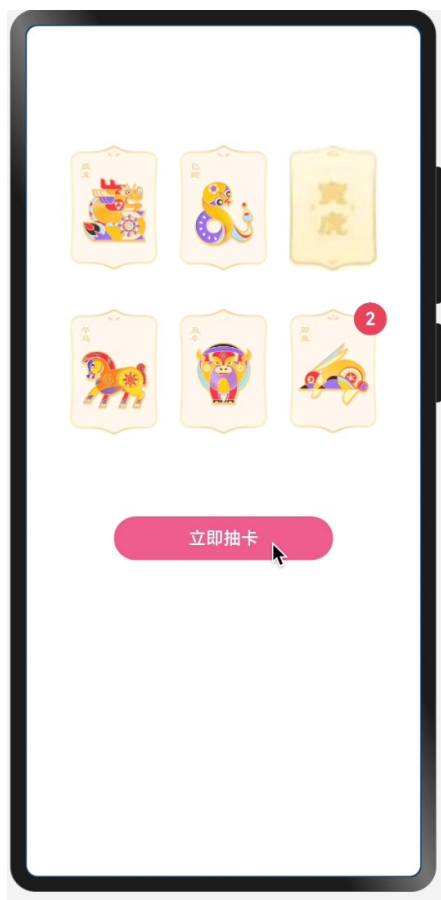
黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌

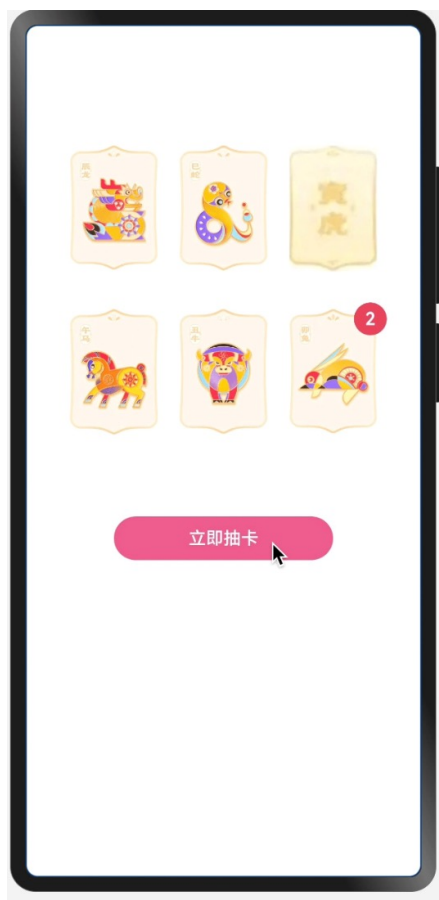
阶段案例 – 生肖抽奖卡



阶段案例 - 生肖抽奖卡



阶段案例 – 生肖抽奖卡 - 初始布局

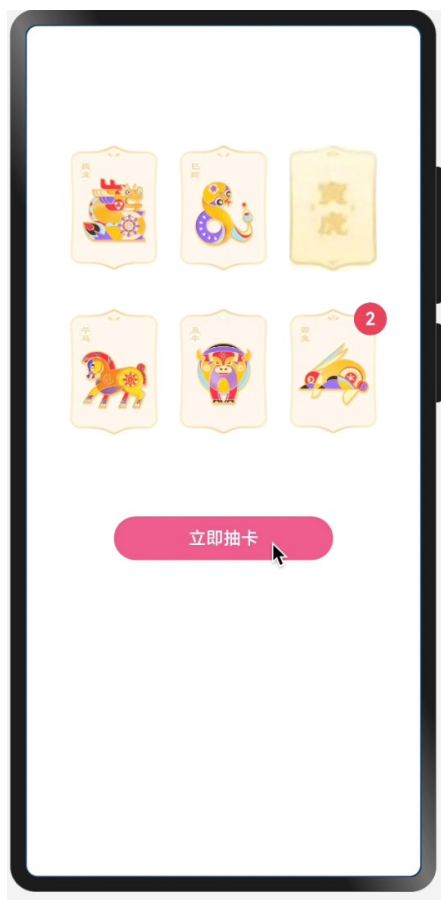


知识点:

1. Badge 角标组件
2. Grid 布局

```
Badge({  
  count: 1, // 角标数值  
  position: BadgePosition.RightTop, // 角标位置  
  style: {  
    fontSize: 12, // 文字大小  
    badgeSize: 16, // 圆形大小  
    badgeColor: '#FA2A2D' // 圆形底色  
  }  
}) {  
  Image($r('app.media.bg_01'))  
    .width(80)  
}
```

阶段案例 - 生肖抽奖卡 - 初始布局

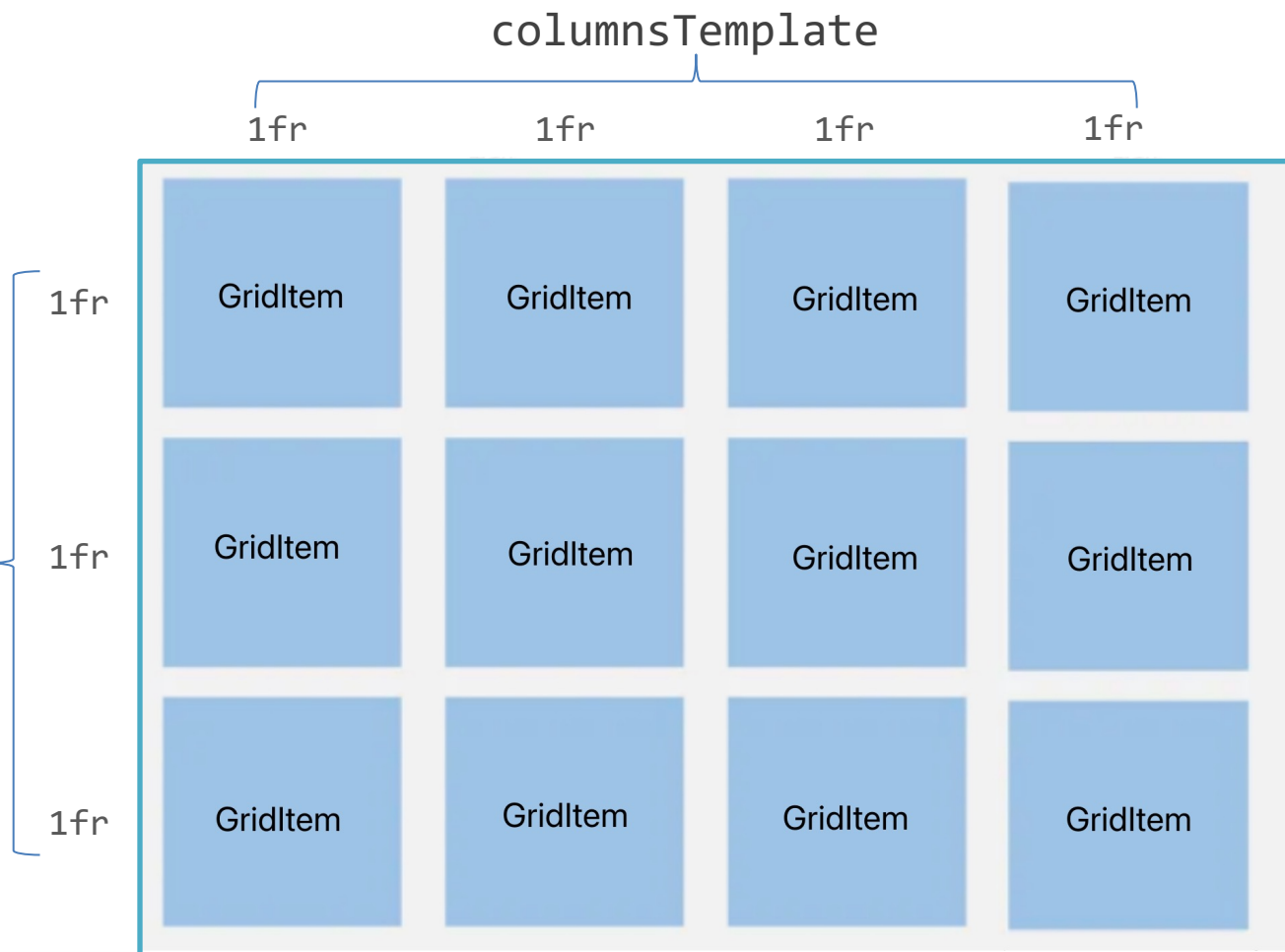


知识点:

1. Badge 角标组件

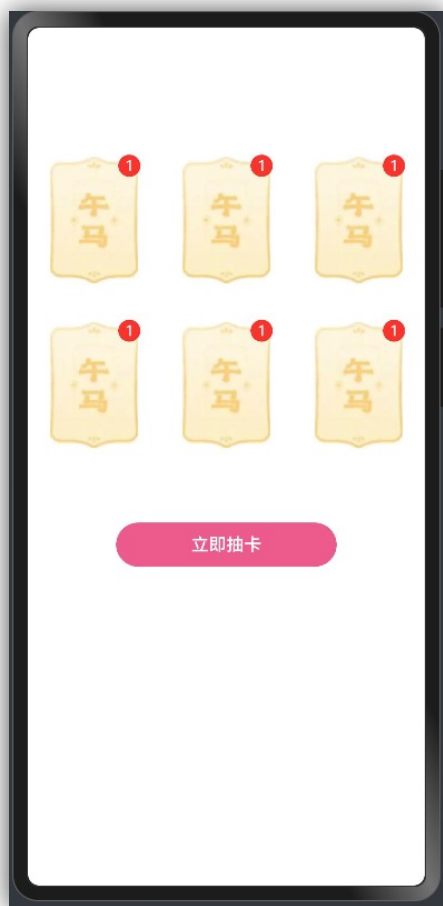
2. Grid 布局

rowsTemplate



Grid组件

阶段案例 – 生肖抽奖卡 – 数据动态渲染

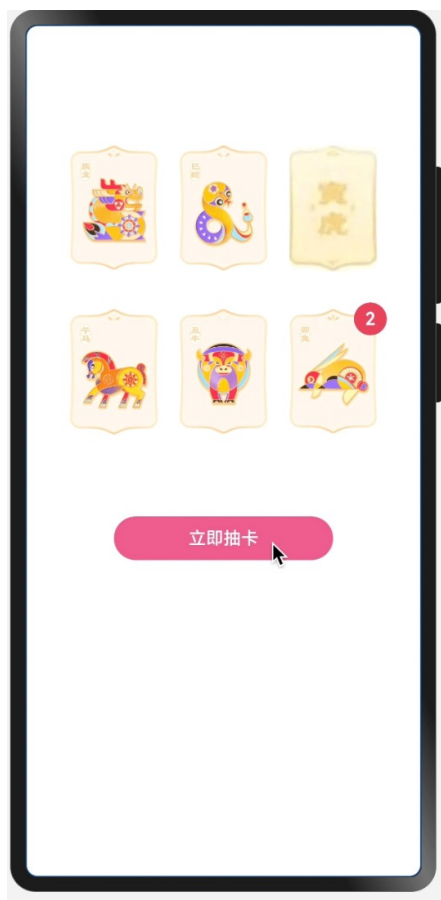


每个列表项 – 两个数据:

1. 图片的地址
2. 抽中的数量

```
GridItem() {  
  Badge({  
    count: 1,  
    position: BadgePosition.RightTop,  
    style: {  
      fontSize: 14,  
      badgeSize: 20,  
      badgeColor: '#fa2a2d'  
    }  
  }) {  
    Image($r('app.media.bg_01'))  
      .width(80)  
  }  
}
```

阶段案例 - 生肖抽奖卡



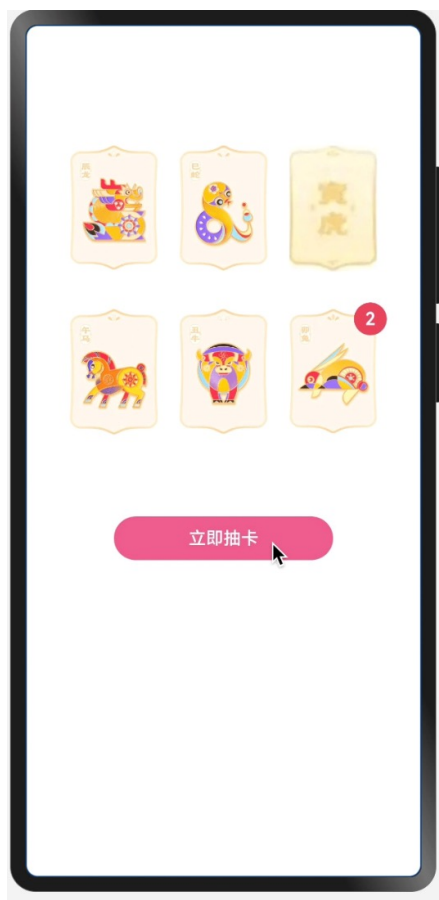
阶段案例 – 生肖抽奖卡 – 抽卡遮罩层



思路分析:

1. 布局角度: 层叠布局 Stack
2. 结构角度: Column > Text + Image + Button

阶段案例 – 生肖抽奖卡 – 显隐效果控制



需求1: 点击抽卡, 才显示遮罩层

需求2: 图片, 要有缩放显示动画

遮罩显隐控制:

1. 透明度 **opacity**: 0 => 1

2. 层级 **zIndex**: -1 => 99

图片动画:

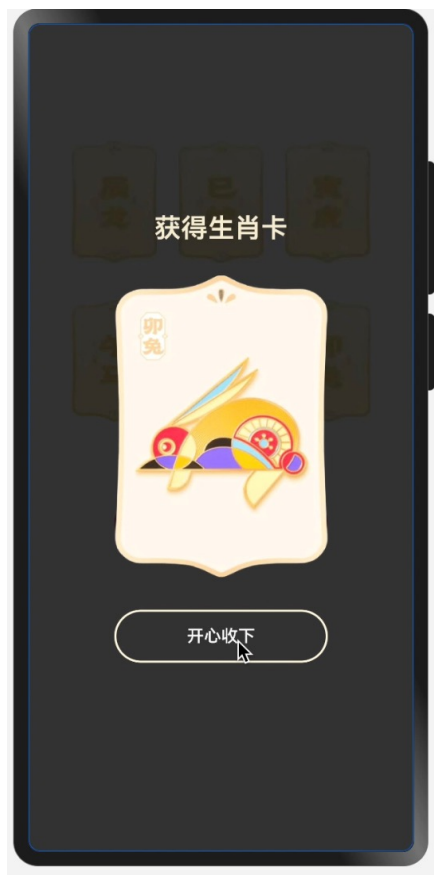
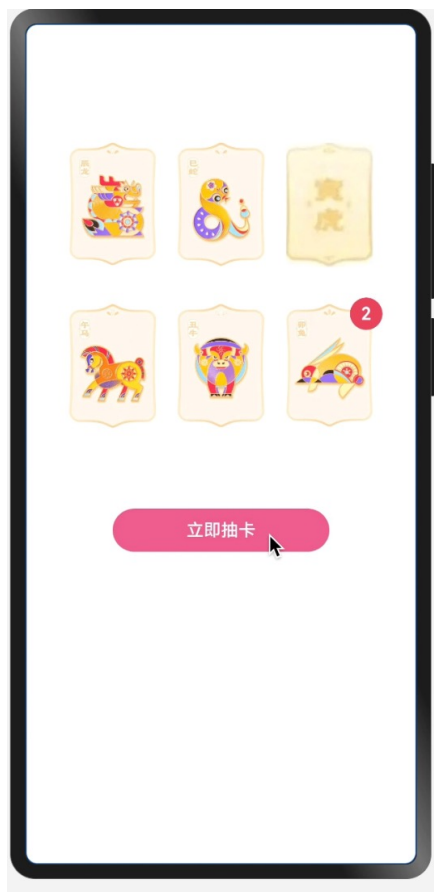
缩放 **scale**: 0 => 1

```
.animation({  
  duration: 500  
})
```


阶段案例 – 生肖抽奖卡 – 随机卡片



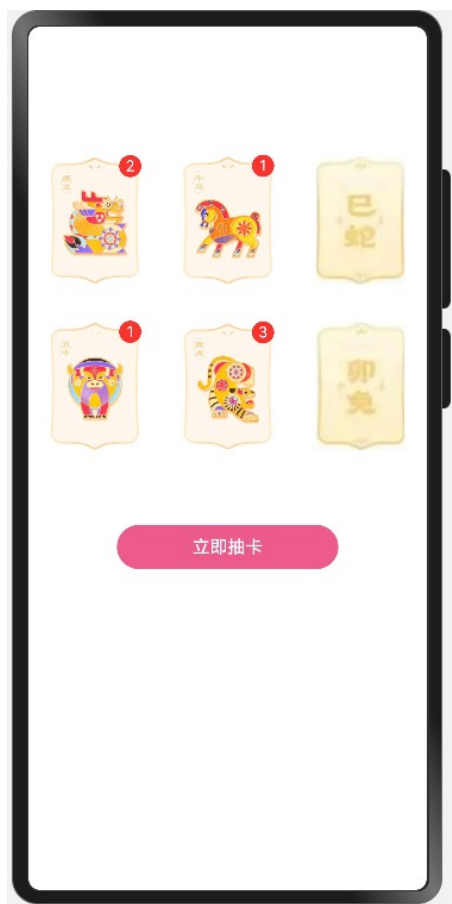
阶段案例 – 生肖抽奖卡 – 随机卡片



需求:

1. 取随机数 `Math.random`
2. 控制展示 (换图)
3. 点击收下, 卡片数累加

阶段案例 – 生肖抽奖卡



阶段案例 – 生肖抽奖卡 – 抽大奖遮罩层



思路分析:

1. 布局角度: 层叠布局 Stack
2. 结构角度: Column > Text + Image + Button

阶段案例 – 生肖抽奖卡 – 抽大奖显隐控制



需求说明：六张卡片集齐，显示 - 中大奖页面

思路：

1. 准备一个变量，控制显隐
2. 每次收下卡片，判断是否集齐，集齐显示中奖页面

阶段案例 – 生肖抽奖卡 – 随机奖品 & 再来一次



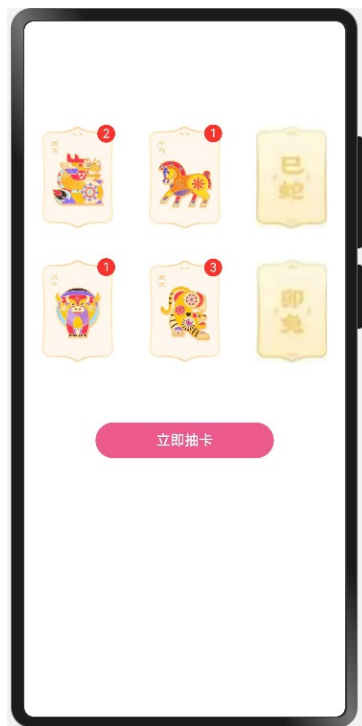
需求1：奖品随机

需求2：再来一次

思路：

1. 奖品随机 → 准备一个**奖品数组**，`Math.random` 随机取下标
2. 再来一次 → 重置数据

总结



知识点总结:

1. Badge 角标组件
2. Grid布局、Stack布局
3. 数组对象动态渲染、动态更新
4. 遮罩层动画、图像动画效果 animation
5. 随机抽奖 Math.random, Math.floor
6. 假设成立法, 判断是否中奖



传智教育旗下高端IT教育品牌