**一、步骤**

1. **开发基本的菜单结构：一级菜单、二级菜单**
2. **用户体验问题（优化）**

**问题1：用户无法通过斜线操作快速选中二级菜单中的内容，只能水平移动到二级菜单，体验不好如何解决？**

**答案：**

**加入延迟来优化：切换子菜单的时候，用setTimeout设置延迟。**

**问题2：当鼠标在一级菜单上频繁引动时会有一个连续切换的效果，用户体验不好！如何解决？**

**答案：**

**debounce去抖技：当事件被频繁触发时，只执行一次处理，一般是最后一次！**

**问题3：延迟引入之后产生了新的体验问题：当用户不想进入二级菜单的时候，没办法很快的在一级菜单中切换，会有一个延迟！如何解决？**

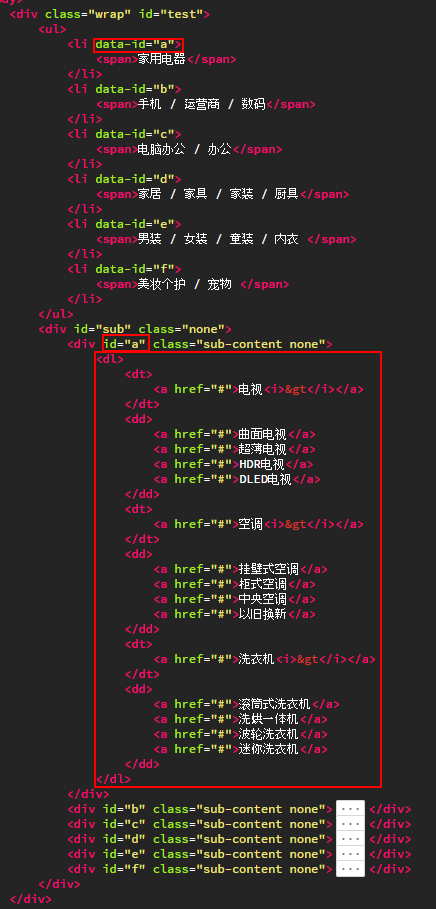
**答案：**

**基于用户行为预测的切换技术**

**二、思路与技巧**

**1. HTML**

* **给每个li添加属性data-id，给每个二级菜单添加属性id，使它们两者相等。**
* **二级菜单默认隐藏，通过JS来控制其显示与隐藏。**
* **二级菜单的结构采用dl dt dd比较符合语义化**

****

1. **CSS**

* **通常二级菜单采用绝对定位，只要知道其父容器的宽高，就可以很容易让二级菜单出现在指定的位置。**
* **使用BFC来使得子元素浮动仍然让能撑开父元素的高度。**

****

* **什么时候给元素添加类，什么时候用伪类？**

1. **当样式的切换需要用到JS的时候，就给元素添加类；**
2. **如果不需要的就可以用伪类，例如:hover。**
3. **JS**

* **使用事件代理（委托），把事件绑定在父容器上，而不是选中所有列表项循环为每一个列表项绑定事件！**
  + **原理：事件冒泡**
  + **好处：无论以后增加还是删除列表项都不需要修改代码，提高性能。**
* **mouseenter和mouseover的区别：**

**1. 使用mouseover/mouseout时，如果鼠标移动到子元素上，即便没有离开父元素，也会触发父元素的mouseout事件；**

**2. 使用mouseenter/mouseleave时，如果鼠标没有离开父元素，在其子元素上任意移动，也不会触发mouseleave事件。**

* **加入延迟来优化从而解决问题1**
  + **切换子菜单的时候，用setTimeout设置延迟。**
  + **当鼠标还在子菜单中的时候不触发事件。**
* **debounce去抖技术来解决问题2**
* **当事件被频繁触发时，只执行一次处理，一般是最后一次！**
  + **实现**

**1. 设置计时器**

**2. 如果事件触发的时候计时器还没有执行，就清掉。**

**3. 在计时器回调结束之后将计时器设置为null。**

* **基于用户行为预测的切换技术**

1. **mousemove一般绑定在document上，这是一个常识。**
2. **鼠标离开猜的时候需要对mousemove进行解绑，所以要独立出来。**

**三、基于用户行为预测的切换技术**

**1. 思路**

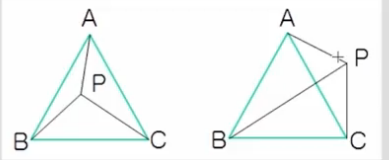
* **通过判断鼠标是否在三角形内可以知道用户是否希望移动到二级菜单，如果是则延迟，如果不是则不延迟。**
* **当用户希望移动到二级菜单中的时候，鼠标的轨迹一定是在当前这个点和二级菜单上下边缘构成的三角形之内。**
* **跟踪鼠标的移动**
* **用鼠标当前位置，和鼠标上一次位置与子菜单上下边缘形成的三角形区域进行比较**

**2. 如何判断鼠标在三角形内**

* **向量： Vab = Pb – Pa**
* **二维向量叉乘公式：a(x1, y1) \* b(x2, y2) = x1\*y2 – x2\*y1**
* **用叉乘法判断点在三角形内**

**如果向量PA\*PB，PB\*PC，PA\*PC它们叉乘的结果符号都是相同的时候就证明P在三角形内；**

**如果符号不同P就在三角形外面。**

****

**这里P是鼠标的当前位置；**

**A就是鼠标上一次的位置；**

**B和C分别是二级菜单的上边缘和下边缘；**

**3. 最终效果**

* **鼠标自然的移动和点击到子菜单**
* **切换时无延迟**