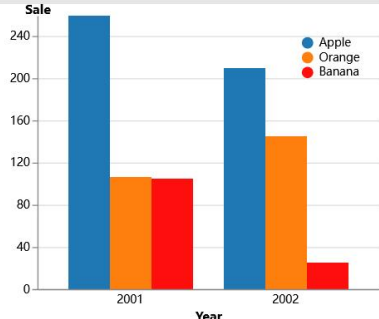
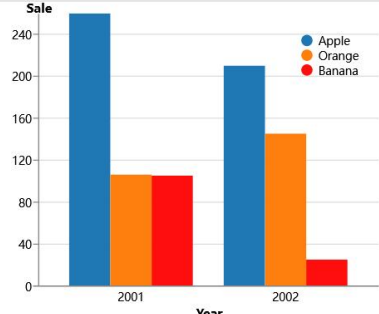


山东大学计算机科学与技术学院

大数据分析实践课程实验报告

学号：202300130220		姓名：刘傲宇		班级：数据科学与大数据技术班	
实验题目：canis 实践					
实验学时：2			实验日期：2025/11/14		
实验目标：					
通过 Canis 官网文档，系统掌握图表模板、动画参数、交互设计等基础概念，理解 JSON 配置文件中选择器、动画类型、时序控制等核心要素的定义逻辑；操作 Canis 编辑器，完成自带示例项目的加载与运行，精准观察默认动画的轴绘制顺序、数据点加载方式及过渡效果等表现特征。					
实验步骤与内容：					
1、基础概念系统性学习					
以 Canis 官网文档为核心学习资料，重点梳理工具的核心概念与配置逻辑，为实操奠定理论基础。					
明确 Canis 的核心组成包括图表模板（基于数据增强型 SVG，即 DSVG）、动画参数体系和交互设计模块。其中图表模板是动画的载体，需包含数据属性与视觉元素的关联；动画参数是控制动画表现的核心；交互设计则实现用户与图表的动态联动。					
2、内置示例项目加载与运行					
选取 Canis 编辑器自带的“Fruit Sale”等示例项目，完成加载与运行操作，重点观察默认动画的表现特征。					
在“内置模板”列表中选择“Fruit Sale”项目，点击播放按钮，等待编辑器完成图表渲染与动画初始化。					
<div><div><div>CANIS</div><div><div>0. Fruit Sale</div><div><div><div>1</div><div>2</div><div>3</div><div>4</div><div>5</div><div>6</div><div>7</div><div>8</div><div>9</div><div>10</div><div>11</div><div>12</div><div>13</div><div>14</div><div>15</div><div>16</div><div>17</div><div>18</div><div>19</div><div>20</div><div>21</div><div>22</div><div>23</div><div>24</div><div>25</div><div>26</div><div>27</div></div><div><pre>{ "charts": [{ "source": "../charts/groupedBar.dsvg" }], "animations": [{ "selector": ".rectangle", "grouping": { "groupBy": "position", "reference": "start after previous", "delay": 200, "grouping": { "groupBy": "id", "delay": 100 } }, "effects": [{ "type": "wipe bottom", "duration": 300 }] }] }</pre></div></div></div></div><div><div>Chart</div><div></div></div><div><div>Result Animation</div><div></div></div><div><div>console</div><div><div>[LOG]: rendering...</div><div>[LOG]: The duration of the generated animation is: 1600ms</div><div>[LOG]: Done rendering.</div></div></div></div>					

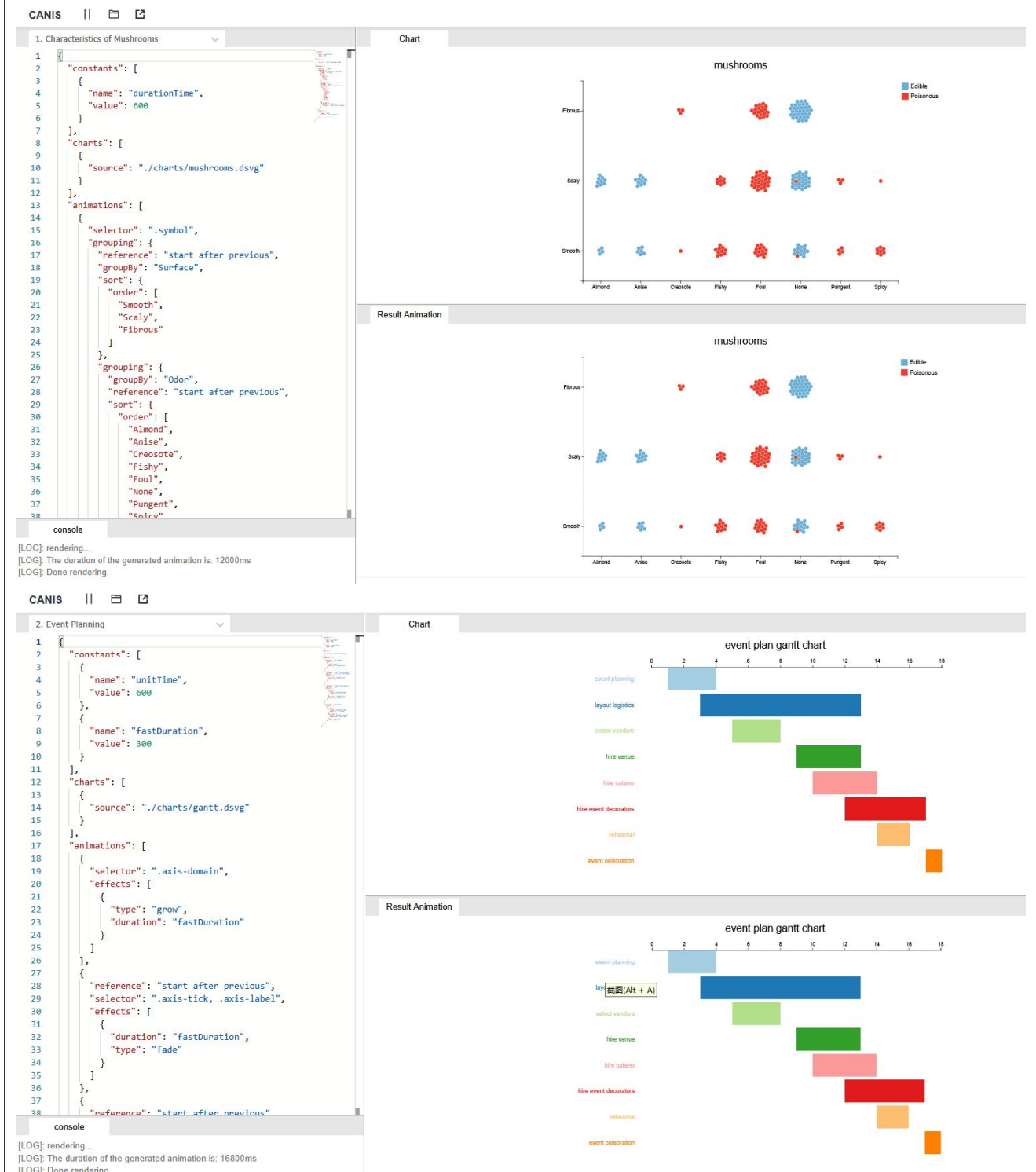
从三个维度记录动画表现：

轴的绘制顺序：首先完成 X 轴（水果类别）与 Y 轴（销售额）的静态渲染，随后轴标签以“淡入”效果依次显现，确保视觉焦点先聚焦于坐标轴定义。

数据点的动态加载：柱状图数据以“从下到上”的“wipe up”效果逐组加载，每组对应一种水果的月度销售额，加载顺序遵循数据集的时间字段排序。

过渡效果展现：组间切换时存在 50ms 的轻微延迟，配合“easeInOut”缓动效果，避免动画生硬跳转，增强数据变化的连贯性。

另外，尝试其他几个示例项目：





3、尝试调整核心参数以观察不同动画效果。

- 选择器（selector）：修改影响的目标元素（如数据点、矩形、文字等）。
- 分组（grouping）：依据不同字段（如时间、分类）分组数据，影响动画的播放顺序。
- 动画类型（type）：如“fade”（淡入淡出）或“wipe left”（从左到右显现）。
- 时序效果（timing）：通过增加或减少动画时长、设置延迟（delay），调整动画节奏。

如，改变 grouping 所依据的数据属性，使动画时序将数据划分依据从按类型变成按时间；将“type”设置为“fade”就能实现动画以淡出的方式进行数据状态切换。

结论分析与体会：

Canis 的核心价值在于“平衡易用性与可视化效果”，其设计理念更偏向“数据故事讲述”

而非“专业数据建模”，适合市场、运营等业务人员制作演示用数据动画，而非开发人员构建企业级可视化系统；

Canis 的 JSON 配置体系层次分明，各参数的作用边界清晰，这种设计既保证了灵活性，又避免了配置冲突，体现了领域特定语言的专业性；

在小型报告、产品演示、教学课件等场景中，Canis 能大幅提升工作效率；但面对大规模数据、复杂交互或多浏览器兼容需求时，仍需优先选择 D3.js、ECharts 等更成熟的工具。