1.项目所需安装配置介绍  
· 操作系统：Windows/Mac

· 开发工具：VSCode

· 技术栈：Vue.js (3.x), Leaflet, Element-Plus, Pinia

· 项目依赖：npm install vue leaflet element-plus pinia

2.项目运行环境  
VScode，nodejs，npm，Vue3

3. 项目运行方式：

导入项目，进入项目目录，打开控制台，  
输入npm install 安装项目所需依赖，  
然后输入npm run dev 运行项目  
接着打开本地的网页，即可看到网页。

4. 项目目录结构描述：

|-- src

    |-- assets        # 静态资源

    |-- components    # 公共组件

    |-- views         # 任务界面

        |-- Task1     # 海洋热浪时空分布特征分析

        |-- Task2     # 海洋热浪与多变量因素关联分析

        |-- Task3     # 海洋热浪生态影响评估

    |-- router.js     # 路由配置

|-- store.js      # 全局状态管理

5. 项目关键模块的介绍

6. 项目AI赋能点的介绍  
  
代码生成与优化：

使用方式：我向 AI 提供了初始代码和问题描述（例如日期选择器无法选择、地图重复），AI 生成了修复方案（TaskOne.vue 的 localTimeRange 和 watch 逻辑）。

赋能点：AI 快速分析代码错误（this 语法问题、响应式绑定失效），提供可执行的 Vue.js 解决方案，节省调试时间。

示例：AI 建议将 fetchHeatwaveData 从 methods 移到 setup，并添加经度归一化逻辑。

数据分析与验证：

使用方式：我上传了 heatwave.geojson，AI 分析其格式是否符合标准，指出经度超范围问题。

赋能点：AI 自动检测 GeoJSON 数据异常（经度 > 180°），提出修正方法（减 360），提高了数据处理的准确性。

示例：AI 检查 "coordinates": [[199.937106918239, 34.21487603305785]]，建议归一化。

技术方案设计：

使用方式：我询问如何限制 Leaflet 地图，AI 提供了 maxBounds 和 noWrap 的详细方案。

赋能点：AI 基于 Leaflet 文档和最佳实践，设计了地图限制方案，避免了我手查阅复杂文档的时间。

示例：AI 建议 maxBoundsViscosity: 1.0，确保边界固定。

前端后端交互指导：

7. 参考他人的项目或开发技术和可视化等资源的引用：

8. 项目运行过程中会出现的问题与解决方案：  
通过本次实验，我完成了海洋热浪数据的可视化展示，解决了日期选择和地图重复的技术难题，并初步掌握了通过热点实现前后端交互的方法。AI 在代码生成、问题诊断、技术指导和文档编写中发挥了关键作用，显著提高了实验效率和成果质量。