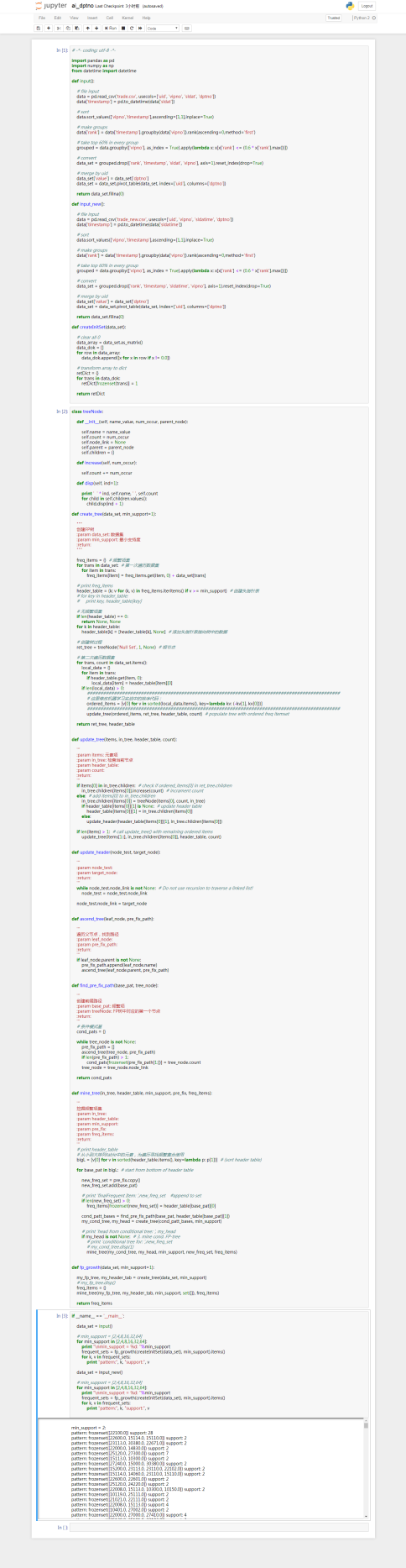
## 代码运行

[点击查看原图](ai_dptno.png)



### 讨论分析

1. 本题采用FP-Growth算法计算商品购买记录中，以单个用户进行单次购买为单位的频繁项集。
2. 算法参考： *Machine Learning in Action*, Peter Harrington
3. FP-Growth算法对于每个频繁项，构造它的[条件投影](https://baike.baidu.com/item/%E6%9D%A1%E4%BB%B6%E6%8A%95%E5%BD%B1/3393313)数据库和投影[FP-tree](https://baike.baidu.com/item/FP-tree/9801330)。对每个新构建的FP-tree重复这个过程，直到构造的新FP-tree为空，或者只包含一条路径。当构造的FP-tree为空时，其前缀即为频繁模式；当只包含一条路径时，通过枚举所有可能组合并与此树的前缀连接即可得到频繁模式。
4. 具体步骤包含：构建FP-tree、频繁挖掘模式。
5. 实际输入为全部用户的单笔订单样本，此处itemno为dptno。
6. 输出为对应频繁项集合。
7. 分析输出可见，频繁集中大部分或者说support较高的频繁项都是单个itemno构成，相当于该itemno在数据集中出现的次数统计。这样的数据符合频繁集要求但是在之后的预测计算中将予以过滤。

### 性能比较

完成两个数据集的频繁集挖掘及输出，总计运行时间为：36.340s

考虑到dptno相较于bndno有着更少的可能性，即dptno会出现更复杂的频繁模式，因此运行耗时相较于bndno有明显增加。