## salarykit\_dataset\_ele每日定时处理：

日数据都是当天的，36文件是累计的，但是ext文件取值最后一天，所以也是当天的。

在生成下面表的时候，没有做时间判断，所以应该是累计数据。

ele\_day\_13\_std\_ext：

ele\_day\_13\_std[运单号, 评价等级=’非常满意’]

ele\_day\_34\_std\_lookup\_map[账单时间（日期）, 骑手ID, 站点ID, 运单号]

std\_qplus\_vendor\_dc\_map[vendor\_dc\_id（站点ID）,dc\_id]

按运单号关联，得到非常满意的配送员信息

\*小括号中的名称代表修改后的名字

ele\_worker\_order\_common\_kpi\_by\_daily\_t2：

ele\_day\_32\_std\_ext[账单时间（日期）, 骑手ID, 站点ID, 虚假配送单, 配送原因取消单]

ele\_day\_36\_std\_ext[账单时间（日期）, 骑手ID, 站点ID, 违规（不包含罚单）, 申诉]

ele\_day\_34\_std[账单时间（日期）, 骑手ID, 站点ID, 业务类型=‘配送收入’, 订单号, 详情=‘完成单-超时’].groupby([日期, 骑手ID, 站点ID]).[订单号（超时单量）].agg(count)

ele\_day\_13\_std\_ext[日期, 骑手ID, 站点ID, 评价等级, 运单号].groupby([日期, 骑手ID, 站点ID]).[运单号（好评单量）].agg(count)

ele\_day\_34\_std[账单时间（日期）, 骑手ID, 站点ID, 业务类型=’配送收入’, 订单号, 详情=[差评,完成单,完成单-超时]].groupby([日期, 骑手ID, 站点ID]).[运单号（完成单量）].agg(count)

以最后一张表[日期, 骑手ID, 站点ID]为基准聚合

计算[准时单量] = [完成单量] - [超时单量]，计算出勤

ele\_worker\_order\_common\_kpi\_by\_monthly\_t2：

以ele\_worker\_order\_common\_kpi\_by\_daily\_t2表为基础，把每天的累计求和

## salarykit\_dataset\_ele每月定时处理：

ele\_worker\_order\_common\_kpi\_by\_daily\_month：

ele\_month\_33\_std\_ext[账单时间, 骑手ID, 站点ID, 虚假配送单量,配送原因取消单量]

ele\_month\_37\_std\_ext[账单时间, 骑手ID, 站点ID, 违规（不包含罚单）,申诉]

ele\_worker\_order\_common\_kpi\_by\_daily\_t2[日期, 骑手ID, 站点ID, 好评单量]

ele\_month\_35\_std[业务类型=’配送收入’,详情=’完成单-超时’].groupby([日期, 骑手ID, 站点ID]).[订单号（超时单量）].agg(count)

ele\_month\_35\_std[账单时间（日期）, 骑手ID, 站点ID, 业务类型=’配送收入’, 订单号, 详情=[差评,完成单,完成单-超时]].groupby([日期, 骑手ID, 站点ID]).[运单号（完成单量）].agg(count)

以最后一张表[日期, 骑手ID, 站点ID]为基准聚合

计算[准时单量] = [完成单量] - [超时单量]，计算出勤

ele\_worker\_order\_common\_kpi\_by\_monthly\_month:

以ele\_worker\_order\_common\_kpi\_by\_daily\_month表为基础，把每天的累计求和