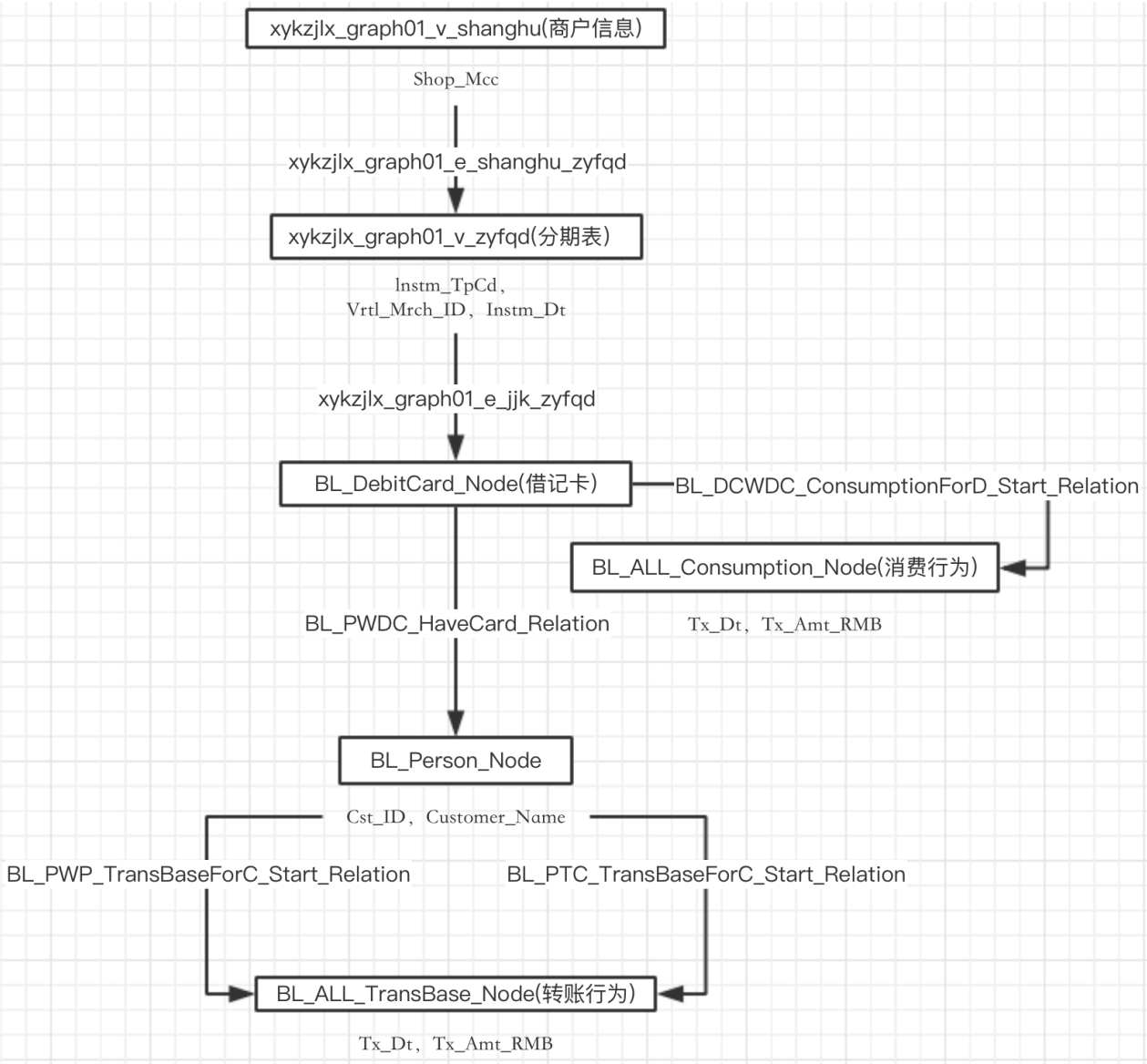


反套现练习

反套现TXZC_XFXD_005:

规则: 当月，装修分期客户在3日内（第一笔交易起算）累计交易≥5万元，其名下借记卡在5日内（第一笔交易起算）单笔或累计有接近交易额95%-100%的进账款项。

相关vertex及其edge如下:



相关属性:

属性	属性中文名	类型	备注
Instm_TpCd	分期类型代码	Spring	Vrtl_Mrch_ID='666666714'时:'or'9'表示现金分期; Vrtl_Mrch_ID='666666698'or'666666714'时: 'F'表示装修分期; 'D'表示分期通; 'E'表示购车分期;
Vrtl_Mrch_ID	虚拟商户编号	Spring	
Instm_Dt	分期日期	Spring	
Shop_Mcc	商户MCC	Spring	=7299:未列入其他代码的个人服务（其他房地产服务） =8912:装修、装潢、园艺 =4722:旅行社 =0001:信用卡预授权/冻结 =5714:帏帐、窗帘、室内装潢商店 =5211:木材和各类建材卖场 =5039:未列入其他代码的建材批发(批发商)
Tx_Amt_RMB	交易金额	double	
Tx_Dt	交易日期		
Cst_ID	借记卡卡号		
Customer_Name	客户姓名		

练习一：

请根据给定的规则，思考该如何找出满足规则的客户及其交易日期、金额，写出你的逻辑。

练习二：

```
/* data_time = 2019-01-01; */
start = {xykzjlx_graph01_v_zyfqd.*};
L0 = select t from start:t
      where trim(t.Instm_TpCd) == "D"
            and trim(t.Vrtl_Mrch_ID) in ("6666666698","6666666714")
            and to_datetime(trim(t.Instm_Dt)) between data_time and
datetime_add(data_time,interval 1 month);
L0_1 = select s
      from start:s-(xykzjlx_graph01_e_shanghu_zyfqd)-
>xykzjlx_graph01_v_shanghu:t
      where trim(s.Vrtl_Mrch_ID) == "6666666698"
            and trim(t.Shop_Mcc) == "7299"
            and to_datetime(trim(s.Instm_Dt)) between data_time and
datetime_add(data_time,interval 1 month);
L0 = L0 Union L0_1;
```

Q：最终得到的L0是什么？（大概描述即可）

练习三：

请仿照练习一的代码，找出分期类型为装修分期的分期。

练习四：

```
/* spend_limit = 50000; */
SetAccum<DATETIME> @fq_time;
MapAccum<datetime,SumAccum<double>> @spend_group,@trans_group;
OrAccum @suspect;
L1 = select t from L0:s-(xykzjlx_graph01_e_jjk_zyfqd)->BL_DebitCard_Node:t
      accum t.@fq_time += to_datetime(s.Instm_Dt);

L2 = select s
```

```

        from L1:s-(BL_DCWDC_ConsumptionForD_Start_Relation)-
>BL_ALL_Consumption_Node:t
        accum
            foreach fq_time in s.@fq_time do
                if t.Tx_Dt between fq_time and datetime_add(fq_time,interval 3
day) then
                    s.@spend_group += (fq_time->t.Tx_Amt_RMB)
                end
            end
        post-accum
            foreach(fq_time,spend_money) in s.@spend_group do
                if spend_money > spend_limit then
                    s.@suspect += True
                end
            end
        having s.@suspect == True;

L3 = select t from L2:s-(BL_PWDC_HaveCard_Relation)->BL_Person_Node:t
        accum t.@spend_group += s.@spend_group;

```

Q: @spend_group用来存储什么? L1,L2,L3分别代表什么?

练习五:

```

/* FILE f(filepath);
product_num = ''; */
L4 =
    select s
    from L3:s-
((BL_PWP_TransBaseForC_Start_Relation|BL_PTC_TransBaseForC_Start_Relation))
->BL_ALL_TransBase_Node:t
where t.Tx_Dt between data_time
    and datetime_add(data_time,interval 1 month)
accum
    foreach (fq_time,spend_money) in s.@spend_group do
        if t.Tx_Dt between fq_time and datetime_add(fq_time,interval 5 day) then
            if t.Tx_Amt_RMB <spend_money and t.Tx_Amt_RMB> spend_money * 0.95 then

```

```

        f.println("TXZC_XFXD_004_2" + product_num,s.Cst_ID,s.Customer_Name,
            datetime_format(fq_time,"%Y-%m-%d"),spend_money,
            "单笔",
            datetime_format(t.Tx_Dt,"%Y-%m-%d"),t.Tx_Amt_RMB)
    end,
    s.@trans_group += (fq_time->t.Tx_Amt_RMB)
end
end
post-accum
foreach (fq_time,trans_money) in s.@trans_group do
    if trans_money <s.@spend_group.get(fq_time)
        and trans_money> s.@spend_group.get(fq_time) * 0.95 then
        f.println("TXZC_XFXD_004_2" + product_num,s.Cst_ID,s.Customer_Name,
            datetime_format(fq_time,"%Y-%m-
%d"),s.@spend_group.get(fq_time),
            "累计",
            datetime_format(fq_time,"%Y-%m-%d"),trans_money )
        end
    end;
end;

```

Q: @trans_group用来存储什么? 哪些内容最终被写入文件?

练习六:

请根据练习二三四五写出找出满足规则TXZC_XFXD_005客户的流程, 并与你对练习一的回答进行比较, 提出你的建议或想法。

参考答案TXZC_XFXD_005

练习二:

分期类型为分期通且分期日期在2019年1月内的分期。

练习三:

```

L0 = select t from start:t
      where trim(t.lnstm_TpCd) == "F"
            and trim(t.Vrtl_Mrch_ID) in ("666666698","666666714")
            and to_datetime(trim(t.Instm_Dt)) between data_time and
                  datetime_add(data_time,interval 1 month);
L0_1 = select s
        from start:s-(xykzjlx_graph01_e_shanghu_zyfqd)-
>xykzjlx_graph01_v_shanghu:t
        where trim(s.Vrtl_Mrch_ID) == "666666698"
              and trim(t.Shop_Mcc) in ("8912","4722","0001","5714","5211","5039")
              and to_datetime(trim(s.Instm_Dt)) between data_time and
                    datetime_add(data_time,interval 1 month);
L0 = L0 Union L0_1;

```

练习四：

L1：办理装修分期的借记卡，并记录其分期日期；

L2：找出在3日内（第一笔交易起算）累计交易≥5万元的借记卡：对每张卡查看其所有交易信息，在每张卡的分期日期中，若有交易日期在其分期日期3日内，则累加其交易金额，记录在@spend_group中。若在@spend_group中有金额>5万，则标记该卡到@suspect为true。

L3：找到L2借记卡对应的个人，并复制属性@spend_group。

@spend_group用于记录每张借记卡的每个分期日期3日内的总交易金额。

练习五：

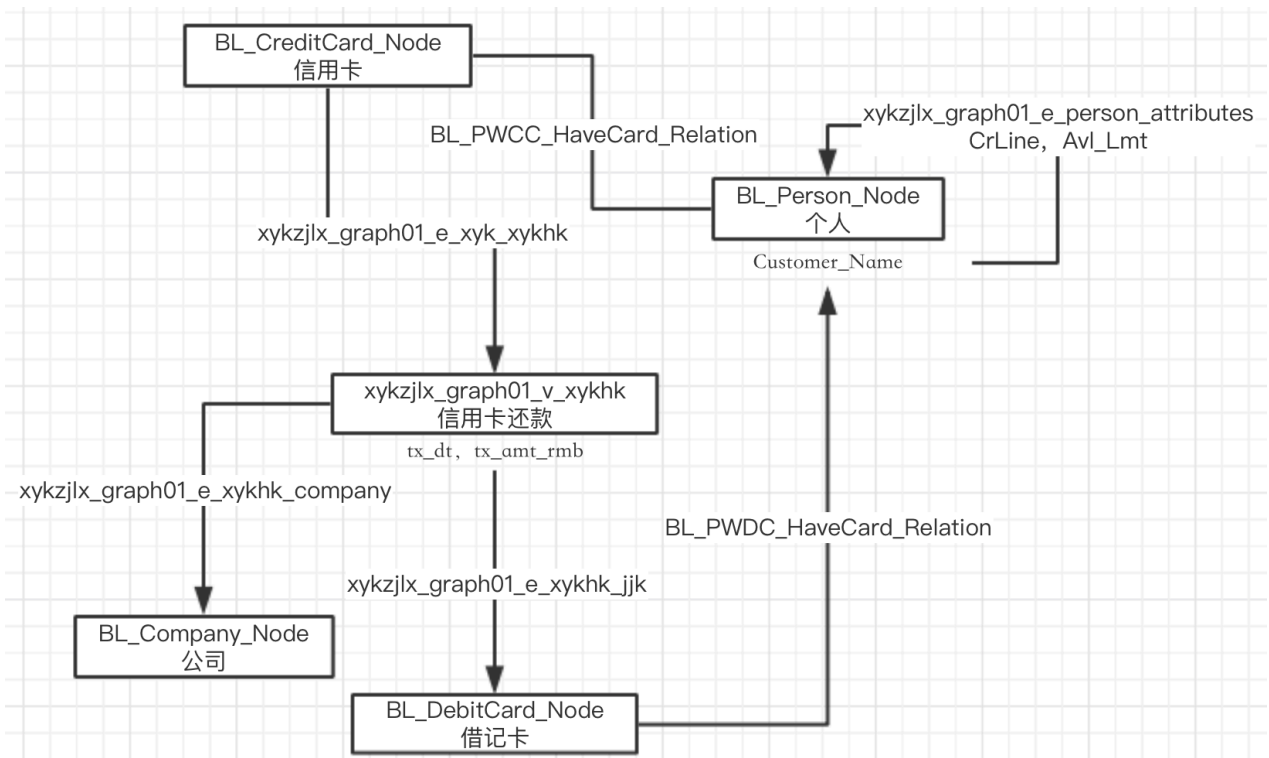
@trans_group存储在分期5日内单笔或累计有接近交易额95%-100%的进账款项信息（进账日期、金额）。以下内容被写入文件：

"TXZC_XFXD_004_2", 借记卡卡号，客户姓名，分期日期，分期3日内总交易金额，单笔/累计，分期日期，分期5日内总进账金额。

反套现TXZC_XHxD_013

当期，一张借记卡或一个对公账户向≥10名信用卡客户还款，且单名信用卡客户累计被还款金额≥2万元，且当期或上一期账单有过额度使用率在60%以上。

相关vertex及其edge如下：



相关属性：

属性	属性中文名	类型	备注
CrLine	每月总额度	MapAccum<int,double>	邓君猜的
Avl_Lmt	每月剩余额度	MapAccum<int,double>	邓君猜的
tx_dt	交易日期	DATE	
tx_amt_rmb	折人民币金额	DECIMAL(18,2)	
Cust_Name	客户姓名		

练习一：

请根据给定的规则，思考该如何找出满足规则的客户及其还款金额，写出你的逻辑。

练习二：

```

/* data_time = 2019-01-01 */
time1 = datetime_to_epoch(datetime_sub(datetime_add(data_time,interval 1
month),interval 1 day)); # 当期最后一天
time0 = datetime_to_epoch(datetime_sub(data_time,interval 1 day)); # 上一期最后
一天
start = {BL_CreditCard_Node.*};

```

```

L0 = select t from start:s-(BL_PWCC_HaveCard_Relation)->BL_Person_Node:t; # 信用卡客户

L1 = select t from L0:s-(xykzjlx_graph01_e_person_attributes:e)->BL_Person_Node:t
      accum t.@line += e.CrLine
      having t.@line.get(time1) > 0;

L2 = select t from L1:s-(xykzjlx_graph01_e_person_attributes:e)->BL_Person_Node:t
      accum t.@xf_line += e.Avl_Lmt
      having t.@xf_line.get(time1)/t.@line.get(time1) < 0.4;

```

Q: 请指出L1, L2分别代表什么?

练习三:

```

L2_1 = select t from L2:s-(BL_PWCC_HaveCard_Relation)->BL_CreditCard_Node:t
      accum t.@person += s;

L3 = select t from L2_1:s-(xykzjlx_graph01_e_xykhk_xykhk)->xykzjlx_graph01_v_xykhk:t
      where t.tx_dt between data_time and datetime_add(data_time,interval 1 month)
      accum t.@person += s.@person;

```

Q: 请指出L3代表什么?

练习四:

```

/* money_limit = 20000, person_num = 10 */
#对私

L4 = select t from L3:s-(xykzjlx_graph01_e_xykhk_jjk)->BL_DebitCard_Node:t
      accum t.@hk_group += (s.@person.get(0)->s.tx_amt_rmb);

L5 = select t from L4:s-(BL_PWDC_HaveCard_Relation)->BL_Person_Node:t
      accum t.@hk_group = s.@hk_group
      post-accum
      foreach (person,hk_money) in t.@hk_group do
        if hk_money > money_limit then

```



```

        t.@suspect_group += (person->hk_money)
    end
end
having t.@suspect_group.size() >= person_num;

L6 = select t from L5:t
post-accum
    foreach (person,hk_money) in t.@suspect_group do
        f.println("TXZC_XHxD_013_person",t,t.Customer_Name,person,hk_money)
    end;

```

Q：为什么L5中hk_money > money_limit这一判断放在post-accum中而不是在accum中？此处输出到文件中的是哪一种客户？

练习五：

请参照练习三代码，找出满足规则的对公账户，并写入文件。新涉及vertex与edge如下：

BL_Company_Node (vertex, 对公账户)

xykzjlx_graph01_e_xykhk_company (edge, 连接vertex信用卡还款xykzjlx_graph01_v_xykhk与vertex对公账户BL_Company_Node)

练习六：

请根据练习二三四五写出找出满足规则TXZC_XHxD_013客户的流程，并与你对练习一的回答进行比较，提出你的建议或想法。

参考答案TXZC_XHxD_013

练习二：

L1有额度信用卡客户， L2大额度使用率信用卡客户

练习三：

L3表示还款日期中2019年1月的信用卡还款， 并标记还款人@person。

练习四：

hk_money在accum中计算， accum计算结束后每个顶点的hk_group才是包括全部还款行为的map集合， 故应在post-accum中遍历。此处输出到文件的是满足规则的个人客户。

练习五：

```
#对公
L7 = select t from L3:s-(xykzjlx_graph01_e_xykhk_company)->BL_Company_Node:t
where t.outdegree()<10000000
accum t.@hk_group += (s.@person.get(0)->s.tx_amt_rmb)
post-accum
  foreach (person,hk_money) in t.@hk_group do
    if hk_money > money_limit then
      t.@suspect_group += (person->hk_money)
    end
  end
having t.@suspect_group.size() >= person_num;

L8 = select t from L7:t
post-accum
  foreach (person,hk_money) in t.@suspect_group do
    f.println("TXZC_XHxD_013_company",t,t.Cust_Name,person,hk_money)
  end;
```