Импорт из RK CRM

Для создания файла со списком счетов при импорте из r_keeper CRM 7 можно воспользоваться SQL-скриптом. Важно: скрипт не обрабатывает ситуацию, когда у одного счёта в CRM сразу несколько карт. Такие карты не будут обработаны.

```
/*
Выгрузка из CRM для PROBONUS. (в пробонусе надо убрать требование заполнения полей
анкеты)
 @allow_blocked_cards - выгружать ли заблокированные карты
 @allow_persons_wo_cards - выгружать ли владельцев без карт после фильтрации
@allow_blocked_cards
PEOPLE_ID - id владельца в CRM
NAME,FULL_NAME - полное имя из карточки владельца
CARD, CARD2, CARD3 карты владельца
RESTSUM сумма всех активных бонусных счетов
из бонусных счетов для выгрузки выбирается последний по алфавиту:
BONUSNAME - имя счета
BONUSNO - код бонуса для RK
для скидок аналогично
личные данные держателя карты не учитываются, только данные владельца
*/
declare @allow_blocked_cards int = 0;
declare @allow_persons_wo_cards int = 0;

WITH ACCOUNTS_table AS (
select RTRIM(LTRIM(cp.FULL_NAME)) AS FULL_NAME, cp.PEOPLE_ID,
cat.ACCOUNT_TYPE_ID, cat.ACCOUNT_CLASS, cat.NAME AS BONUS_NAME, cpa.BALANCE
from CARD_PEOPLES cp
  left join CARD_PEOPLE_ACCOUNTS cpa ON cpa.PEOPLE_ID=cp.PEOPLE_ID and
cpa.DELETED=0 and cpa.BLOCKED=0
  left join CARD_ACCOUNT_TYPES cat ON
cpa.ACCOUNT_TYPE_ID=cat.ACCOUNT_TYPE_ID and cat.DELETED=0 and
cat.IS_ACTIVE=1
  --left join CARD_CARDS cc ON cc.PEOPLE_ID=cp.PEOPLE_ID and cc.DELETED=0
and cc.STATUS=1 -- csActive = 1; csInactive = 2; csDisabled = 3; csBlocked =
4; csIssue = 5; // физическая выдача владельцу
where cp.deleted=0 and cpa.DELETED=0 and cat.IS_ACTIVE=1 --and cc.STATUS=1
),
ACCOUNTS_PEOPLES AS (
select distinct(PEOPLE_ID), FULL_NAME from  ACCOUNTS_table
),
RESULT_TABLE AS (
SELECT
 ap.PEOPLE_ID AS PEOPLE_ID
 ,ap.FULL_NAME AS NAME
 ,(SELECT TOP (1) cc.CARD_CODE FROM CARD_CARDS cc WHERE
cc.PEOPLE_ID=ap.PEOPLE_ID and cc.DELETED=0 and (cc.STATUS=1 OR
```

```
(@allow_blocked_cards=1))) AS CARD
 ,(SELECT cc.CARD_CODE FROM CARD_CARDS cc WHERE cc.PEOPLE_ID=ap.PEOPLE_ID
and cc.DELETED=0 and (cc.STATUS=1 OR (@allow_blocked_cards=1)) ORDER BY
CARD_CODE OFFSET 1 ROWS FETCH NEXT 1 ROWS ONLY) AS CARD2
 ,(SELECT cc.CARD_CODE FROM CARD_CARDS cc WHERE cc.PEOPLE_ID=ap.PEOPLE_ID
and cc.DELETED=0 and (cc.STATUS=1 OR (@allow_blocked_cards=1)) ORDER BY
CARD_CODE OFFSET 2 ROWS FETCH NEXT 1 ROWS ONLY) AS CARD3
 --,(SELECT cc.CARD_CODE FROM CARD_CARDS cc WHERE cc.PEOPLE_ID=ap.PEOPLE_ID
and cc.DELETED=0 and cc.STATUS=1 ORDER BY CARD_CODE OFFSET 3 ROWS FETCH NEXT
1 ROWS ONLY) AS CARD4
 ,(SELECT SUM(BALANCE) from ACCOUNTS_table at1 WHERE
at1.PEOPLE_ID=ap.PEOPLE_ID and at1.ACCOUNT_CLASS=1) AS RESTSUM
 --,(select TOP(1) ACCOUNT_TYPE_ID FROM ACCOUNTS_table at_1 where
at_1.PEOPLE_ID=ap.PEOPLE_ID and ACCOUNT_CLASS=1 and BONUS_NAME like 'Бонус%'
ORDER BY BONUS_NAME DESC) as BONUS_ACC_TYPE_ID
 --бонусы
 ,(select EXTERNAL_ID FROM CARD_SCHEMES where CARD_SCHEMES.DELETED=0 AND
CARD_SCHEMES.SCHEME_ID =
     (select top(1) SCHEME_ID from CARD_ACCOUNT_LEVELS where
CARD_ACCOUNT_LEVELS.DELETED=0 and CARD_ACCOUNT_LEVELS.BLOCKED=0
     AND  ACCOUNT_TYPE_ID =
     (select TOP(1) ACCOUNT_TYPE_ID FROM ACCOUNTS_table at_1
     where at_1.PEOPLE_ID=ap.PEOPLE_ID and ACCOUNT_CLASS=1 and BONUS_NAME
like 'Бонус%' ORDER BY BONUS_NAME))
  ) AS BONUSNO
 ,(select TOP(1) BONUS_NAME FROM ACCOUNTS_table at_1 where
at_1.PEOPLE_ID=ap.PEOPLE_ID and ACCOUNT_CLASS=1 and BONUS_NAME like 'Бонус%'
ORDER BY BONUS_NAME) as BONUS_NAME
--скидка
  ,(select EXTERNAL_ID FROM CARD_SCHEMES where CARD_SCHEMES.DELETED=0 AND
CARD_SCHEMES.SCHEME_ID =
     (select top(1) SCHEME_ID from CARD_ACCOUNT_LEVELS where
CARD_ACCOUNT_LEVELS.DELETED=0 and CARD_ACCOUNT_LEVELS.BLOCKED=0
     AND  ACCOUNT_TYPE_ID =
     (select TOP(1) ACCOUNT_TYPE_ID FROM ACCOUNTS_table at_1
     where at_1.PEOPLE_ID=ap.PEOPLE_ID and ACCOUNT_CLASS=2 ORDER BY
BONUS_NAME))
  ) AS DISCOUNTNO
 ,(select TOP(1) BONUS_NAME FROM ACCOUNTS_table at_1 where
at_1.PEOPLE_ID=ap.PEOPLE_ID and ACCOUNT_CLASS=2 ORDER BY BONUS_NAME) as
DISCOUNT_NAME

 --контакты
 ,(select top(1) contact_value FROM CARD_CONTACTS cc1 WHERE
cc1.PEOPLE_ID=ap.PEOPLE_ID
     and cc1.CONTACT_TYPE IN (select CONTACT_TYPE_ID from CARD_CONTACT_TYPES
where FLAGS=6)) AS TEL1
 ,(select contact_value FROM CARD_CONTACTS cc1 WHERE
cc1.PEOPLE_ID=ap.PEOPLE_ID
     and cc1.CONTACT_TYPE IN (select CONTACT_TYPE_ID from CARD_CONTACT_TYPES
where FLAGS=6) ORDER BY contact_value OFFSET 1 ROWS FETCH NEXT 1 ROWS ONLY)
```

```
AS TEL2
 ,(select TOP(1) contact_value FROM CARD_CONTACTS cc1 WHERE
cc1.PEOPLE_ID=ap.PEOPLE_ID
    and cc1.CONTACT_TYPE IN (select CONTACT_TYPE_ID from CARD_CONTACT_TYPES
where FLAGS=5)) AS EMAIL
 --допданные
 ,(select NOTES from CARD_PEOPLES cp2 where cp2.PEOPLE_ID=ap.PEOPLE_ID) AS
EXTINFO
 ,CONVERT(NVARCHAR,(select BIRTHDAY from CARD_PEOPLES cp2 where
cp2.PEOPLE_ID=ap.PEOPLE_ID),104) AS BIRTHDAY
 ,(select SEX from CARD_PEOPLES cp2 where cp2.PEOPLE_ID=ap.PEOPLE_ID) AS SEX
 from ACCOUNTS_PEOPLES AS ap
)
--select * from RESULT_TABLE
select
NAME,
ISNULL(CAST(CARD as nvarchar),'') AS CARD,
ISNULL(CAST(CARD2 as nvarchar),'') AS CARD2,
ISNULL(CAST(CARD3 as nvarchar),'') AS CARD3,
--ISNULL(CAST(CARD4 as nvarchar),'') AS CARD4,
ISNULL(RESTSUM,0) AS RESTSUM,
ISNULL(BONUSNO,'') AS BONUSNO,
ISNULL(BONUS_NAME,'') AS BONUS_NAME,
ISNULL(DISCOUNTNO,'') AS DISCOUNTNO,
ISNULL(DISCOUNT_NAME,'') AS DISCOUNT_NAME,
ISNULL(TEL1,'') AS TEL1,
ISNULL(TEL2,'') AS TEL2,
ISNULL(EXTINFO,'') AS EXTINFO,
ISNULL(BIRTHDAY,'') AS BIRTHDAY,
ISNULL(SEX,'') AS SEX

from RESULT_TABLE
where /*not bonus_NAME is null*/
(not card is null
OR not card2 is null
OR not card3 is null
--and not card4 is null
OR (@allow_persons_wo_cards=1))
 order by NAME
```