**Пререквезиты – выполнить код по импорту необходимых файлов**

options mprint mlogic symbolgen;

%macro tech\_import(mpObjectName=course\_lifecycle,

mpObjectType=CSV,

mpOutputLibref = work

);

%local lmvObjectName

lmvObjectType

lmvOutputLibref

;

%let lmvObjectName= &mpObjectName.;

%let lmvObjectType = %lowcase(&mpObjectType.);

%let lmvOutputLibref = &mpOutputLibref.;

%let mvObjectsCount = %sysfunc(countw(&lmvObjectName., %str( )));

%do i=1 %to &mvObjectsCount.;

%let mvObjectSelected = %scan(&lmvObjectName., &i., %str( ));

%put &=mvObjectSelected;

proc import datafile="/var/data/mipt/&mvObjectSelected..&lmvObjectType."

DBMS=&lmvObjectType.

out=&lmvOutputLibref..&mvObjectSelected.

replace

;

getnames=yes;

run;

%end;

%mend tech\_import;

%tech\_import(mpObjectName=course\_tt course\_tt\_sku course\_promo course\_sales\_inv course\_sum\_sls\_tt\_date course\_sku course\_inventory course\_lifecycle course\_movesale\_part1 course\_movesale\_part2 course\_movesale\_part3 course\_movesale\_part4,

mpObjectType=CSV,

mpOutputLibref = work

);

**Задание 1.** Создать таблицу **valid\_structure\_movesale:**

proc sql noprint;

create table work.valid\_structure\_movesale

(

Date\_day numeric,

LOCATION\_ID numeric,

PRODUCT\_ID numeric,

Sls\_Retail\_AD\_Rub numeric,

Sls\_Retail\_BD\_Rub numeric,

Sls\_Retail\_Qty numeric,

Small\_Scale\_Wholesales numeric,

Small\_Scale\_Wholesales\_AD\_RUB numeric,

VERSION\_ID numeric,

location\_id\_char numeric,

product\_id\_char numeric,

return\_ad\_rub numeric,

return\_qty numeric

)

;

quit;

**Задание 2.** Получить список всех объектов в новую таблицу WORK.GET\_OBJECTS из таблицы sashelp.vtable – библиотека ‘work’, маска названия таблицы – ‘movesale\_’

**Задание 3.** Развернуть полученный список объектов из таблицы **GET\_OBJECTS** в макропеременные – список таблиц поместить в **mvTABLE\_NAMES,** список библиотек в **mvLIB\_NAMES.**

(\*HINT – SELECT COUNT(VAR\_NM) INTO :mvCNT FROM TABLE\_NM – выдаст единственное значение (рассчитанное количество строк) в макропеременную mvCNT.   
Select VAR\_NM into :mvVAR\_NAMES\_LIST separated by “ “ from TABLE\_NM --- поместит список ВСЕХ существующих значений переменной VAR\_NM через разделитель by “ “. )

**Задание 4.** Посчитать количество имен таблиц (библиотек) из списка &mvVAR\_NAMES\_LIST – ищем по списку символьных функций SAS SUPPORT. (Необходимо пробежаться по полученному списку и посчитать количество «строк») [**http://support.sas.com/documentation/cdl/en/lefunctionsref/63354/HTML/default/viewer.htm#p0w6napahk6x0an0z2dzozh2ouzm.htm**](http://support.sas.com/documentation/cdl/en/lefunctionsref/63354/HTML/default/viewer.htm#p0w6napahk6x0an0z2dzozh2ouzm.htm)

Полученное значение записать в новую макропеременную **mvLIST\_CNT**.

Данное значение будет использоваться в цикле.

**Задание 5.** Получить структуру таблицы **valid\_structure\_movesale** в таблицу **valid\_cont. (\*proc contents с опцией out=).**

**Задание 5.1.** Создать таблицу **WORK.MOVESALE\_VALID**, используя структуру таблицы **valid\_structure\_movesale**.

**Задание 6.** Обработка циклом от 1 до i. Достать из списка библиотек и списка таблиц i-тый элемент.

Для выбранной пары библиотека+имя таблицы получить структуру таблицы **invalid\_cont. (&lmvLibName..&lmvTableNm.)**

**Задание 6.1. Проверка структуры таблиц по именам переменных.**

Соединить полученные таблицы со структурами (valid\_cont T1, invalid\_cont T2), выбрав только T1.NAME, COALESCE(T2.NAME, 'INVALID COLUMN') AS CLMN\_FLAG, T1.LENGTH, T1.TYPE. Результат записать в новую таблицу **CHECK\_FOR\_NEW\_COLUMNS.**

(Соединение происходит в данном случае по 1 полю)

**Задание 6.2. Проверка структуры таблиц по формату (имя, длина, тип).**

Создать новую таблицу **T\_WITH\_VALID\_FLG** на основе соединения полученной на шаге 6.1. таблицы с валидной структурой, выбрав следующие поля:

T1.NAME,T1.CLMN\_FLAG, COALESCE(T2.NAME, 'INVALID FORMAT') AS ERR\_DESC

Выбрав только те поля, которые прошли проверку в 6.1. (Использовать полученный флаг).

(Соединение происходит в данном случае по 3м полям)

**Задание 6.3.** Выбрать только валидные данные из таблицы **T\_WITH\_VALID\_FLG в таблицу ONLY\_VALID\_DATA.**

**Задание 6.4.** Посчитать количество полученных строк в таблице **ONLY\_VALID\_DATA** и сравнить с количеством строк в таблице **valid\_cont.** (Количество строк = количество переменных). В случае, если количество совпадает, то добавить через PROC APPEND текущую i-ую таблицу к таблице **WORK.MOVESALE\_VALID.**

**6.5.** Завершение цикла.

**6.6. \*\*Бонусный уровень.** Непрошедшие проверку имена таблиц добавить в новую таблицу WORK.IMPORT\_INVALID со структурой REGIST\_DATE datetime., TABLE\_NM $64, ERR\_DESC $200.   
Записывая имя некорректной таблицы в поле table\_nm, а сообщение «Ошибка при импорте таблицы “&table\_nm”. Некорректный формат входной таблицы» записать в ERR\_DESC, проставив текущую дату+время в поле REGIST\_DATE.