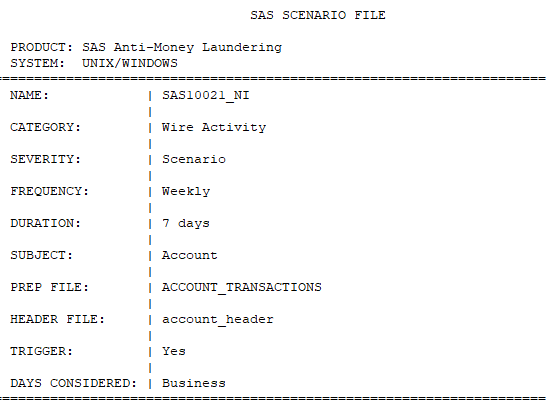
**SCENARIO DOCUMENTATION** :

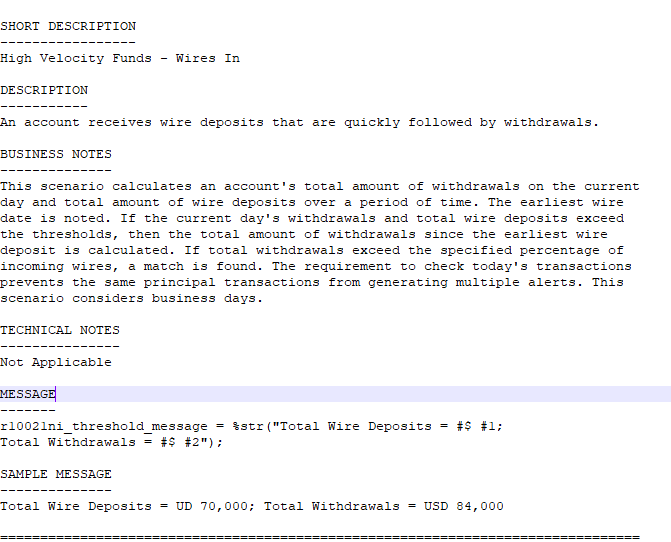
The scenario is divided into 8 parts and each of them are explained briefly.

**Part 1 :**



**Description :**

* Name - defines name of the scenario
* Category - Category of the scenario
* Severity - registered code functions as scenario or risk factors(scenario).
* Frequency - scenario running/execution frequency(weekly).
* Duration - weekly = 7days
* Subject - default entity level
* Prep file - prep file used
* Header file - header file used
* Days considered - scenario considering business or calendar days.

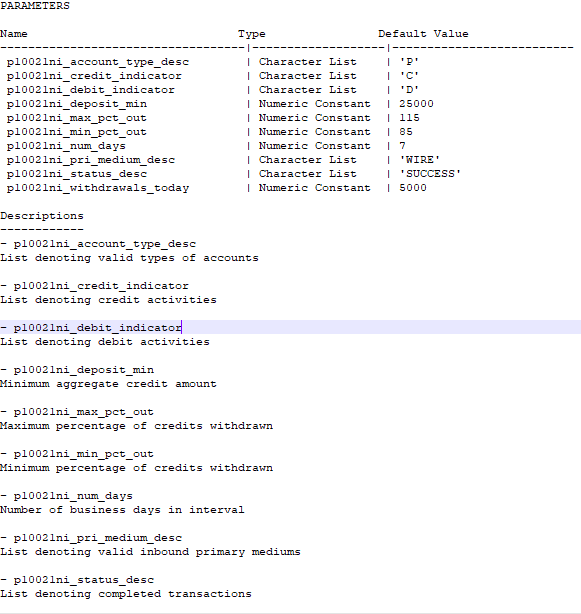
**Part2 :**

**Description :**

* Short\_description - short, abstract desc about scenario. Only 100 characters are available.
* Business notes - A description of how the logic achieves the scenario’s requirements.
* Above, the business note is explaining that:
* The scenario will calculate total amount of withdrawals,debit, on current day and total amount of wire deposits,credit, over a period of time.
* As the frequency is weekly, earliest wire date is recorded.
* Condition is set to calculate the total amount of withdrawals:

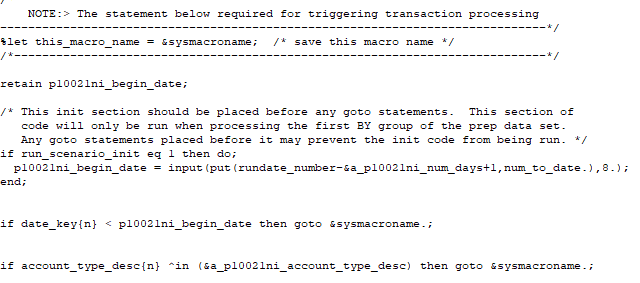
If withdrawal on current day + total wire deposits > threshold.

* Now if this total withdrawal exceeded a certain percentage,of incoming wires, then a match is found.
* Sample message - how the message will appear in the investigation UI.
* Message - how to print the message or what message will be printed on the UI.

**Part3:**

**Description:**

Parameters used,their name, default values, and their description.

**Part4:**

**Description :**

* %let this\_macro\_name = &sysmacroname;  */\* making a macro var and saving this macro name \*/*
* retain p10021ni\_begin\_date;  */\*keeping the value once assigned\*/*
* */\* this statement will run run\_scenario\_init equal to 1\*/*

if run\_scenario\_init eq 1 then do;

* */\* calc. begin date for the parameter; put() will convert num to char; Input() will convert that character value to SAS date format; num\_to\_date format is also used\*/*

p10021ni\_begin\_date = input(put(rundate\_number-&a\_p10021ni\_num\_days+1,num\_to\_date.),8.);

end;

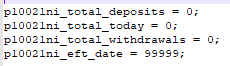
* */\*if date\_key used is less than begin\_date,then macro is called\*/*

if date\_key{n} < p10021ni\_begin\_date then goto &sysmacroname.;

* */\*if current acc\_desc NOT IN given macro variable,then macro is called\*/*

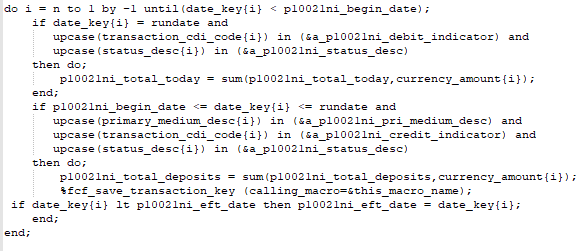
if account\_type\_desc{n} ^in (&a\_p10021ni\_account\_type\_desc) then goto &sysmacroname.;

**Part 5 :**

****

**Description :** initializing parameters values.

**Part 6 :**



**Description :**

In this do-until loop: (which runs from ‘n’ to 1,increment by -1).

In the first IF statement: if current\_date = rundate of scenario;

we calculate total transaction on the current day, and put that value into a parameter(if conditions satisfy)(for diff cdi\_code : credit,debit,inquiry; and status\_desc of transaction).

In the Second IF statement : if current\_date is between begin\_date and rundate;

We calculate total\_deposits in the account on the current day.

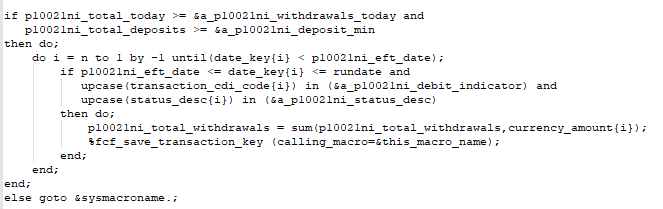
And then call the macro :

*%fcf\_save\_transaction\_key(calling\_macro=&this\_macro\_name);*

And in the last IF statement : if the date\_key is less than p10021ni\_eft\_date,then

the assignment is done.

**Part 7 :**



**Description:** only run if the first IF statement verifies.

IF statement : it will check if the values of total\_today and total\_deposits greater than minimum aggregate current day withdrawals and min. aggregate credit amount respectively(for diff cdi\_code : credit,debit,inquiry).

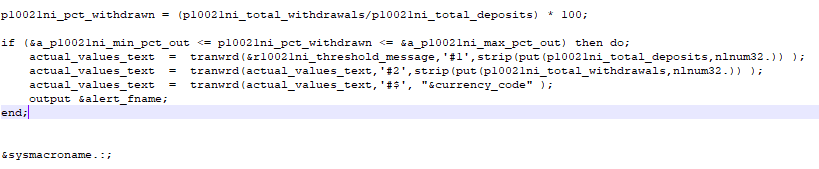
Do-until loop : starts from n to 1; increment by -1

IF statement : check the value of date\_key;if conditions fulfills, then

We calculate the value of total\_withdrawals, and call the macro.

ELSE statement : transferring control flow to macro var;

**Part 8 :**



**Description :**

Calculating withdrawn amt percentage.

IF statement : if the withdrawn percentage is between Minimum percentage of credits withdrawn and Maximum percentage of credits withdrawn, then tranwrd function will replace the source value with total\_deposits, total\_withdrawals and currency\_code in &alert\_fname;

Resolved macro variable.