

*_BEGIN_HEADER_*****

Sponsor: PPD Admin
Protocol: Bios Global Library ADaM
Program Name: IS02SAS.SAS
Sas Version: SAS Versions 9.1.3 and 9.2
Purpose: Macro to convert Date in ISO 8601 format to DATE9 format.
Usage Notes: %IS02SAS(isodate=, datec=[, daten=][, timec=])
 where isodate = Input date in ISO 8601 format
 datec = Output Character Date variable name
 daten = Output Numeric Date variable name

(optional)

[timec = Output Character Time variable name]

(optional)

Files Used: None
Files Created: None
Author Initials: (b) (6)
Date Program Created: 22JUN2009
Copyright: PPD 2009

Modification History:

Date	Initials	Modification Description
------	----------	--------------------------

08Oct2009	(b) (6)	Updated to make as generic macro for ADaM Global Bios Library use
15Oct2009	(b) (6)	Updated macro to create time and missing date when only time is available
15Oct2009	(b) (6)	If entire iso is null, but time is available then datec =
15Oct2009	(b) (6)	Added leading zeros to hrs.

*_END_HEADER_*****
*****;

%MACRO IS02SAS(isodate=, datec=, daten=, timec=) /STORE des="V1.1";

** Check if parameters ISO Date and Ouput Date variables are provided **;
%if &isodate. ne and &datec. ne %then
%do;

length	_yr	\$4
	_mn	\$3
	_dy	\$2
	_daypart	\$10
	&datec.	\$9;

```

_yr      = "";
_mn      = "";
_dy      = "";
_empdate = 0;
_daypart = "";
&datec.  = "";

** If input ISO date is non-missing, then extract the date part **;
if not missing(&isodate.) then
do;
    ** If Time present, get start position and extract the date
part *;
    _tpos = index(&isodate., "T");

** (b) (6) if _tpos = 1 then only time is available **;
    if _tpos > 1 then _daypart = substr(&isodate., 1, _tpos-1);
else if _tpos = 1 then _daypart = "-----";
else _daypart = substr(&isodate., 1);
end;

if not missing(_daypart) then
do;
    *If year, month and day are missing, but time is present*;
    if length(_daypart) >=5 then
do;
        if substr(_daypart, 1,5) eq "-----" then
do;
            _yr = "";
            _mn = "";
            _dy = "";
            _empdate = 1;
        end;
    end;
end;

if _empdate ne 1 then
do;
    ** Set Year **;
    *If year is not missing*;
    if substr(_daypart, 1,1) ne "-" then
do;
        if length(_daypart) >=4 then
do;
            _yr = substr(_daypart, 1, 4);
        end;
    else
do;
        _yr = "----";
    end;
end;
end;
end;

```

```

*If year is missing*;
else if substr(_daypart, 1,1) eq "-" then
do;
    _yr = "----";
end;

** Set Month **;
*If year is not missing, month starts from position
6.*;
if _yr not in ("----", "") then
do;
    if length(_daypart) >=7 then
    do;
        *If month is not missing*;
        if substr(_daypart, 6,1) ne "-"
then
            do;
                if substr(_daypart, 6, 2) =
'01' then
                    _mn= 'JAN';
                else if substr(_daypart, 6,
2) = '02' then
                    _mn= 'FEB';
                else if substr(_daypart, 6,
2) = '03' then
                    _mn= 'MAR';
                else if substr(_daypart, 6,
2) = '04' then
                    _mn= 'APR';
                else if substr(_daypart, 6,
2) = '05' then
                    _mn= 'MAY';
                else if substr(_daypart, 6,
2) = '06' then
                    _mn= 'JUN';
                else if substr(_daypart, 6,
2) = '07' then
                    _mn= 'JUL';
                else if substr(_daypart, 6,
2) = '08' then
                    _mn= 'AUG';
                else if substr(_daypart, 6,
2) = '09' then
                    _mn= 'SEP';
                else if substr(_daypart, 6,
2) = '10' then
                    _mn= 'OCT';
                else if substr(_daypart, 6,
2) = '11' then

```

```

2) = '12' then
else if substr(_daypart, 6,
                _mn= 'NOV';
end;
*If month is missing*;
else if substr(_daypart, 6,1) eq
"- " then
do;
    _mn = "----";
end;
else
do;
    _mn = "----";
end;
end;
*If year is missing, month starts from position
3.*;
if _yr in ("----", "") then
do;
    if length(_daypart) >=4 then
do;
    *If month is not missing*;
    if substr(_daypart, 3,1) ne "-"
then
do;
    if substr(_daypart, 3, 2) =
'01' then
        _mn= 'JAN';
    else if substr(_daypart, 3,
        _mn= 'FEB';
    else if substr(_daypart, 3,
        _mn= 'MAR';
    else if substr(_daypart, 3,
        _mn= 'APR';
    else if substr(_daypart, 3,
        _mn= 'MAY';
    else if substr(_daypart, 3,
        _mn= 'JUN';
    else if substr(_daypart, 3,
        _mn= 'JUL';
    else if substr(_daypart, 3,
2) = '02' then
2) = '03' then
2) = '04' then
2) = '05' then
2) = '06' then
2) = '07' then
2) = '08' then

```

```
else if substr(_daypart, 6,
```

```

2) = '09' then
2) = '10' then
2) = '11' then
2) = '12' then

"-" then

position 5. *;

starts from position 6. *;
"" then

                _mn= 'AUG';
else if substr(_daypart, 3,
                _mn= 'SEP';

else if substr(_daypart, 3,
                _mn= 'OCT';
else if substr(_daypart, 3,
                _mn= 'NOV';
else if substr(_daypart, 3,
                _mn= 'DEC';

end;
*If month is missing*;
else if substr(_daypart, 3,1) eq

do;
    _mn = "----";
end;

end;
else
do;
    _mn = "----";
end;
end;

end;

** Set Day **;
*If year and month are missing, day starts from

if _yr in ("-----", "") and _mn in ("----", "") then
do;
    if length(_daypart) >=6 then
do;
        _dy = substr(_daypart, 5, 2);
    end;
    else
do;
        _dy = "--";
    end;
end;

end;
*If year is missing and month not missing, day

else if _yr in ("-----", "") and _mn not in ("----",
"" then
do;
    if length(_daypart) >=7 then
do;
        _dy = substr(_daypart, 6, 2);

```

```

end;
else
do;
    _dy = "--";
end;
end;
*If year is not missing and month missing, day
starts from position 8. *;
else if _yr not in ("----", "") and _mn in ("---",
"") then
do;
    if length(_daypart) >=9 then
    do;
        _dy = substr(_daypart, 8, 2);
    end;
    else
    do;
        _dy = "--";
    end;
end;
*If year and month are not missing, day starts from
position 9. *;
else if _yr not in ("----", "") and _mn not in
("---", "") then
do;
    if length(_daypart) >=10 then
    do;
        _dy = substr(_daypart, 9, 2);
    end;
    else
    do;
        _dy = "--";
    end;
end;
end;
end;

** Construct output Character Date variable **;
&datec. = trim(left(_dy)) || trim(left(_mn)) ||
trim(left(_yr));

** If entire iso is null then datec = -- --- ---- (b) (6) **;
if &isodate. ne " " and &datec. eq " " then &datec. =
"-----";

** Construct output Numeric Date variable **;
if missing(_dy) or _dy="--" then _dy="";
if missing(_mn) or _mn="---" then _mn="";
if missing(_yr) or _yr="----" then _yr="";

** Clean up **;

```

```

        drop _yr _mn _dy _tpos _daypart _empdate;
    end;
%end;
%else
%do;
    put "ALERT_I: Parameters ISODATE and DATEC are mandatory. Please
provide the input ISO date and output Date variables";
%end;

** 08OCT2009 (b) (6) - Added logic below for Daten handling **;
** Check if input parameters including DATEN variable are provided **;

%if &isodate. ne and &datec. ne and &daten. ne %then
%do;
    length &daten. 8;
    &daten. = .;
    if not missing(&datec.) then
        &daten. = input(compress(&datec.), ??DATE9.);
    format &daten. DATE9.;
%end;

**08OCT2009 (b) (6) - end of updates **;

** Check if input parameters including Time variable are provided **;
%if &isodate. ne and &datec. ne and &timec. ne %then
%do;
    length _timepart $8
           _hrs      $2
           _mns      $2
           _scs      $2
           &timec.    $8;

    _timepart = "";
    _hrs      = "";
    _mns      = "";
    _scs      = "";
    &timec.    = "";

    ** If input ISO date is non-missing, then extract the time part **;
    if not missing(&isodate.) then
    do;
        _tpos = index(&isodate., "T");

        if _tpos > 0 then
            _timepart = substr(&isodate., _tpos+1);
        end;

        if not missing(_timepart) then
        do;

```

```

** Construct output Character Time variable **;
_hrs = trim(left(scan(_timepart, 1, ":")));
_mns = trim(left(scan(_timepart, 2, ":")));
_scs = trim(left(scan(_timepart, 3, ":")));

if missing(_hrs) or _hrs="-" then _hrs = "--";
else _hrs = put(input(_hrs, 2.), z2.); ** ADDED LEADING ZERO (b) (6) **;
if missing(_mns) or _mns="-" then _mns = "--";
if missing(_scs) or _scs="-" then _scs = "--";

*If hours and minutes are present and only seconds are
missing, do not pad dashes for seconds*;
if not missing(_hrs) and not missing(_mns) and _scs="--"
then
    _scs = "";

if _hrs="--" and _mns="--" and _scs="" then
    &timec. = "";
else if not missing(_hrs) and not missing(_mns) and
missing(_scs) then
    &timec. = trim(left(_hrs)) || ":" ||
trim(left(_mns));
else if not missing(_hrs) and not missing(_mns) and not
missing(_scs) then
    &timec. = trim(left(_hrs)) || ":" ||
trim(left(_mns)) || ":" || trim(left(_scs));
end;

** Clean up **;
drop _timepart _hrs _mns _scs _tpos;

%end;

%MEND ISO2SAS;

```