

# ***Section 7 — SAS Edits***



# Contents

<b>Section 7 — SAS Edits.....</b>	<b>7-1</b>
<b>Edit Messages.....</b>	<b>7-9</b>
<i>PROCESS A: Read Current Data File To Get Patient Record .....</i>	<i>7-10</i>
<i>PROCESS B: Create Derived Data Fields .....</i>	<i>7-13</i>
<i>PROCESS C: Data File Edits .....</i>	<i>7-17</i>
<i>S001: Census Date Not Valid.....</i>	<i>7-18</i>
<i>S002: No Header Record Found.....</i>	<i>7-18</i>
<i>S003: Incorrect Campus/Health Services Code.....</i>	<i>7-18</i>
<i>S004: Total Number of Records Not Equal To Total Patient Records.....</i>	<i>7-18</i>
<i>S005: Control Filename Incorrect .....</i>	<i>7-19</i>
<i>S006: Control File Content Incorrect.....</i>	<i>7-19</i>
<i>S051: Data Filename Incorrect.....</i>	<i>7-19</i>
<i>S052: Record Dates After Census Data .....</i>	<i>7-19</i>
<i>S053: See Previous Month Comparison Edits.....</i>	<i>7-19</i>
<i>S060: Blank/Invalid Unique Key.....</i>	<i>7-19</i>
<i>S061: Unique Key For Same Patient Already On List.....</i>	<i>7-20</i>
<i>S065: Blank Patient Identifier.....</i>	<i>7-20</i>
<i>S066: Invalid Patient Identifier.....</i>	<i>7-20</i>
<i>S080: Blank Medicare Number.....</i>	<i>7-20</i>
<i>S081: Invalid Medicare Number.....</i>	<i>7-21</i>
<i>S082: Medicare Code '0' and Age Is Greater Than Six Months.....</i>	<i>7-21</i>
<i>S083: Medicare Code '0' and Age Is Greater Than Twelve Months.....</i>	<i>7-21</i>
<i>S087: Blank Medicare Suffix.....</i>	<i>7-21</i>
<i>S088: Invalid Medicare Suffix.....</i>	<i>7-22</i>
<i>S090: Blank Sex Code.....</i>	<i>7-22</i>
<i>S091: Invalid Sex Code.....</i>	<i>7-22</i>
<i>S092: Indeterminate Sex, Age Greater Than Or Equal To Ninety Days.....</i>	<i>7-22</i>
<i>S093: Indeterminate Sex - Check.....</i>	<i>7-22</i>
<i>S095: Blank Date Of Birth.....</i>	<i>7-23</i>

<i>S096: Invalid Date Of Birth.....</i>	<i>7-23</i>
<i>S098: Age Greater Than 120 Years.....</i>	<i>7-23</i>
<i>S099: Registration Date Before Date Of Birth.....</i>	<i>7-23</i>
<i>S110: Blank Waiting Number.....</i>	<i>7-23</i>
<i>S111: Invalid Waiting Number.....</i>	<i>7-24</i>
<i>SLA assignment.....</i>	<i>7-24</i>
<i>S115: Area of Residence Error.....</i>	<i>7-24</i>
<i>S116: Invalid Locality (part 1).....</i>	<i>7-24</i>
<i>S116: Invalid Locality (part 2).....</i>	<i>7-24</i>
<i>S120: Blank Postcode.....</i>	<i>7-25</i>
<i>S121: Invalid Postcode.....</i>	<i>7-25</i>
<i>S122: Invalid Combination Postcode and Locality.....</i>	<i>7-25</i>
<i>End SLA assignment.....</i>	<i>7-26</i>
<i>S133: Blank Principal Prescribed Procedure.....</i>	<i>7-26</i>
<i>S134: Invalid Principal Prescribed Procedure.....</i>	<i>7-26</i>
<i>S135: Patient Already On Waiting List For Same PPP.....</i>	<i>7-26</i>
<i>S136: See Previous Month Comparison Edits.....</i>	<i>7-26</i>
<i>S146: Blank Surgical Specialty.....</i>	<i>7-27</i>
<i>S147: Invalid Surgical Specialty.....</i>	<i>7-27</i>
<i>S148: See Previous Month Comparison Edits.....</i>	<i>7-27</i>
<i>S160: Blank Clinical Urgency Category.....</i>	<i>7-27</i>
<i>S161: Invalid Clinical Urgency Category.....</i>	<i>7-27</i>
<i>S162: Clinical Urgency Reassignment Incomplete.....</i>	<i>7-27</i>
<i>S166: Blank Planned Length Of Stay.....</i>	<i>7-28</i>
<i>S167: Invalid Planned Length Of Stay.....</i>	<i>7-28</i>
<i>S168: Blank Registration Date.....</i>	<i>7-28</i>
<i>S169: Invalid Registration Date.....</i>	<i>7-28</i>
<i>S170: See Previous Month Comparison Edits.....</i>	<i>7-28</i>
<i>S171: See Previous Month Comparison Edits.....</i>	<i>7-28</i>
<i>S174: New Record, Old Registration Date.....</i>	<i>7-29</i>
<i>S192: Blank Source Of Referral.....</i>	<i>7-29</i>
<i>S193: Invalid Source Of Referral.....</i>	<i>7-29</i>
<i>S194: Blank Referring Hospital.....</i>	<i>7-29</i>

<i>S195: Invalid Referring Hospital.....</i>	<i>7-29</i>
<i>S196: Referring Hospital Conflict.....</i>	<i>7-30</i>
<i>S199: Invalid Date of Last Clinical Urgency Increase.....</i>	<i>7-30</i>
<i>S200: Clinical Urgency Increase Incomplete.....</i>	<i>7-30</i>
<i>S261: Prev Urgency Category Equals Clinical Urgency.....</i>	<i>7-30</i>
<i>S262: Invalid Previous Urgency Category.....</i>	<i>7-30</i>
<i>S263: Invalid Urgency Reassignment Date .....</i>	<i>7-31</i>
<i>S264: Blank Readiness Status.....</i>	<i>7-31</i>
<i>S265: Invalid Readiness Status.....</i>	<i>7-31</i>
<i>S266: Invalid Readiness Change Date.....</i>	<i>7-31</i>
<i>S267: Invalid Reason For NRFC Status .....</i>	<i>7-31</i>
<i>S268: Patient NRFC, No Reason For NRFC.....</i>	<i>7-31</i>
<i>S269: Reason for NRFC Patient Current RFC.....</i>	<i>7-32</i>
<i>S274: Blank Booking Number.....</i>	<i>7-32</i>
<i>S275 Invalid Booking Date.....</i>	<i>7-32</i>
<i>S276: Booking Date Before Registration Date.....</i>	<i>7-32</i>
<i>S277: Invalid Booking Number.....</i>	<i>7-32</i>
<i>S278: Incomplete Booking Details.....</i>	<i>7-33</i>
<i>S279: Invalid Reason For Rebooking.....</i>	<i>7-33</i>
<i>S280: See Previous Month Comparison Edits.....</i>	<i>7-33</i>
<i>S281: Booking Date Present/Booking Number Zero.....</i>	<i>7-33</i>
<i>S282: Scheduled Admission Date Present/Booking Number Zero.....</i>	<i>7-33</i>
<i>S283: Booking/Scheduled Admission Date and Booking Number Mismatch .....</i>	<i>7-33</i>
<i>S285: Invalid Scheduled Admission Date.....</i>	<i>7-34</i>
<i>S286: Scheduled Adm Date Before Booking Date.....</i>	<i>7-34</i>
<i>S287: Scheduled Admission Date Exceeded.....</i>	<i>7-34</i>
<i>S288: Booking Date But Booking Number Blank/Zero.....</i>	<i>7-34</i>
<i>S289: Booking Number But Booking Date Blank.....</i>	<i>7-34</i>
<i>S290: Invalid Removal Date.....</i>	<i>7-34</i>
<i>S291: Removal Date Before Registration Date.....</i>	<i>7-35</i>
<i>S292: Removed But No Reason.....</i>	<i>7-35</i>
<i>S293: Removed, No Removal Date.....</i>	<i>7-35</i>
<i>S294: See Previous Month Comparison Edits.....</i>	<i>7-35</i>

<i>S295: Removal Date Not Equal To Scheduled Admission Date.....</i>	<i>7-35</i>
<i>S296: Patient Admitted, Not Ready For Care.....</i>	<i>7-35</i>
<i>S298: Invalid Reason For Removal.....</i>	<i>7-36</i>
<i>S302: Invalid Insurance Declaration.....</i>	<i>7-36</i>
<i>S303: Blank Insurance Declaration.....</i>	<i>7-36</i>
<i>S306: Invalid Transfer Destination.....</i>	<i>7-37</i>
<i>S307: Transfer Destination Equals Campus/Health Service.....</i>	<i>7-37</i>
<i>S308: Transfer Destination Equals Campus Within This Health Service.....</i>	<i>7-37</i>
<i>S309: Blank Transfer Destination.....</i>	<i>7-38</i>
<i>S310: Transfer Destination Not Transferred.....</i>	<i>7-38</i>
<i>S311: See Previous Month Comparison Edits.....</i>	<i>7-38</i>
<i>S315: See Previous Month Comparison Edits.....</i>	<i>7-38</i>
<i>S318: Total NRFC Days Foll Last Clinical Urgency Increase – Not Numeric Or Blank.....</i>	<i>7-38</i>
<i>S319: Total NRFC Less Than Total Since Last Urgency Increase.....</i>	<i>7-38</i>
<i>S320: Total NRFC Days Not Numeric / Blank.....</i>	<i>7-39</i>
<i>S321: Total NRFC Days Following Last Urgency Reassignment Not Numeric/Blank.....</i>	<i>7-39</i>
<i>S322: Total NRFC Less Than Total Since Urgency Reassignment.....</i>	<i>7-39</i>
<i>S323: See Previous Month Comparison Edits.....</i>	<i>7-39</i>
<i>S324: See Previous Month Comparison Edits.....</i>	<i>7-39</i>
<i>S325: See Previous Month Comparison Edits.....</i>	<i>7-39</i>
<i>S326: Removed and Replaced with S336 S337 S338 1 JUNE 2001 .....</i>	<i>7-39</i>
<i>S327: Removed and Replaced with S336 S337 S338 1 JUNE 2001 .....</i>	<i>7-39</i>
<i>S328: Blank Hospital Initiated Postponement.....</i>	<i>7-39</i>
<i>S329: Invalid Combination Reason For Rebooking and Hospital Initiated Postponements.....</i>	<i>7-40</i>
<i>S330: See Previous Month Comparison Edits.....</i>	<i>7-40</i>
<i>S331: Booking Number Less Than Hospital Initiated Postponements.....</i>	<i>7-40</i>
<i>S332: See Previous Month Comparison Edits.....</i>	<i>7-40</i>
<i>S334: Hospital Initiated Postponement Not Zero.....</i>	<i>7-40</i>
<i>S311: Wait Equals Five Years Or More.....</i>	<i>7-43</i>
<i>S315: Clinical Urgency Category 1, Wait More Than 30 Days.....</i>	<i>7-43</i>
<i>S323: Total NRFC Days Not Equal To Zero.....</i>	<i>7-43</i>
<i>S324: Negative Waiting Time (No Urgency Reassignment) .....</i>	<i>7-44</i>
<i>S325: Negative Waiting Time (Urgency Reassignment) .....</i>	<i>7-44</i>

<i>S342: Tot NRFC Days Foll Last Urg Reassignment Without Clinical Urg Reassig Details...</i>	7-44
<i>S343: Tot NRFC Days Foll Last Urg Increase Without Clinical Urg Increase Details .....</i>	7-44
<i>S345: Date Of Last Clinical Urgency Increase Before 1 July 2001.....</i>	7-44
<i>S346: Urgency Reassignment Date Before Date Of Last Clinical Urgency Increase .....</i>	7-45
<i>S360: Date Of Procedure, But Patient Not Removed.....</i>	7-45
<i>S361: Date Of Procedure Not Valid Date.....</i>	7-45
<i>S362: Date Of Procedure Not Required.....</i>	7-45
<i>S363: Date Of Procedure Blank.....</i>	7-45
<i>S364: Date Of Procedure Before Removal Date.....</i>	7-46
<i>S365: Date Of Procedure After Removal Date, But Planned Same Day.....</i>	7-46
<i>S366: Date Of Procedure More Than 7 Days After Removal.....</i>	7-46
<i>S370: Blank/Invalid Intended/Actual Treatment Campus.....</i>	7-46
<i>COMPARISON EDITS .....</i>	7-47
<i>S136: Procedure Mismatch Between Months.....</i>	7-47
<i>S148: Surgical Specialty Mismatch Between Months.....</i>	7-47
<i>S170: Different Registration Date.....</i>	7-47
<i>S171: New Record, Reg Date Before This Month.....</i>	7-47
<i>S270: Readiness Status Changed But No Readiness Change Date.....</i>	7-47
<i>S280: Booking Number Mismatch Between Months.....</i>	7-48
<i>S294: Record Removed Last Month.....</i>	7-48
<i>S330: Hospital Initiated Postponement Decreased Between Months.....</i>	7-48
<i>S332: Booking Number Increased, Hospital Initiated Postponement unchanged.....</i>	7-48
<i>S335: Hospital Initiated Postponement Increased, Rebook Reason Not HIP.....</i>	7-48
<i>S336: Total Not Ready For Care Days – Fatal Mismatch.....</i>	7-49
<i>S337: Total NRFC Days Following Last Clinical Urgency Reassignment – Fatal Mismatch .</i>	7-49
<i>S338: Total NRFC Days Following Last Clinical Urgency Increase – Fatal Mismatch.....</i>	7-50
<i>S339: Total Not Ready For Care Days Mismatch – Warning.....</i>	7-51
<i>S340: Total Not Ready For Care Days Following Last Urgency Reassignment – Warning....</i>	7-52
<i>S341: Total Not Ready For Care Days Following Last Clinical Urgency Increase – Warning</i>	7-52
<i>S350: Clinical Urgency More Urgent But Date Last Clinical Urgency Increase Blank.....</i>	7-52
<i>S351: Clinical Urgency Less Urgent But Urgency Reassignment Date Blank.....</i>	7-53
<i>S352: Clinical Urgency More Urgent But Date Last Clinical Urgency Increase Unchanged .</i>	7-53
<i>S353: Clinical Urgency Less Urgent But Urgency Reassignment Date Unchanged .....</i>	7-53



# *Edit Messages*

Please note that the SAS code listed below is current as at 1 June 2002 and untested against live data for the 2002—2003 financial year. Therefore it is foreseeable that some alteration may be necessary when ESIS Fifth Edition goes live.

```
/******  
*   PRODUCT:   SAS  
*   VERSION:   8.02  
*   CREATOR:   External File Interface - Version 2  
*   DATE:      31Mar01  
*   DESC:      Generated SAS Datastep Code  
*   NAME:      ESIS_Submission_0203.sas  
*****/  
  
/*Variables given values in Run_mainload_&_reports.sas */  
  
%let campus      = 1121; /* campus/Health service code */  
%let e_ext       = 0107; /* YearMonth*/  
%let esisext     = 071; /* CurrentMonth*/  
%let esislast    = 061; /* PreviousMonth*/  
%let esisjune    = 061; /* June data for urgency date comparisons  
                        Added 1 July 2001 - DO NOT CHANGE */  
%let CDATE = 31/7/01; /* Default value*/  
%let MONTH = 7; /* Default value*/  
  
%include "f:\data\esis_data\ref_slas\formats_for_sla_allocation.sas";  
libname esis_v "f:\data\esis_data\esis_v\edata\";  
libname esis_iv "F:\data\esis_data\esis_iv\edata\";  
libname esis_ref "F:\data\esis_data\ref_tables\internal use only\";  
  
%macro setlib;  
    %if &esislast ne 062 %then  
        libname esislast "f:\data\esis_data\esis_v\edata\";  
    %else libname esislast "f:\data\esis_data\esis_iv\edata\";  
        libname esislast "F:\data\esis_data\esis_iv\edata\";  
%mend;  
%setlib;  
  
/*Read control file to get the header record - cdate, campus code and*/  
/*num of records */
```

```

data esis_v.con1;
  %let _EFIERR_ = 0;      /* clear ERROR detection macro variable */
  infile "f:\data\esis_data\esis_v\rawdata\&campus.\C&esisext.&campus..TXT"
delimiter = ',' MISSOVER DSD;
  format cdate date9.;
  format campus $4.;
  format n_rec best12.;
  informat cdate ddmmyy10.;
  informat campus $4.;
  informat n_rec best32.;
input      cdate
           campus $
           n_rec;

  If _ERROR_ then call symput('_EFIERR_',1);
run;

data esis_v.con2;
  set esis_v.con1;

  format e_ext $4. rcdte date9.;

  e_ext=&e_ext.;

  /*      This is required to check the correct census date      */

  if e_ext = 0207 then rcdte = '31Jul2002'd;
  if e_ext = 0208 then rcdte = '31Aug2002'd;
  if e_ext = 0209 then rcdte = '30SEP2002'd;
  if e_ext = 0210 then rcdte = '31OCT2002'd;
  if e_ext = 0211 then rcdte = '30NOV2002'd;
  if e_ext = 0212 then rcdte = '31DEC2002'd;
  if e_ext = 0301 then rcdte = '31JAN2003'd;
  if e_ext = 0302 then rcdte = '28FEB2003'd;
  if e_ext = 0303 then rcdte = '31MAR2003'd;
  if e_ext = 0304 then rcdte = '30APR2003'd;
  if e_ext = 0305 then rcdte = '31MAY2003'd;
  if e_ext = 0306 then rcdte = '30JUN2003'd;

  if e_ext = 0207 then call symput('CDATE','31/07/2002');
  if e_ext = 0208 then call symput('CDATE','31/08/2002');
  if e_ext = 0209 then call symput('CDATE','30/09/2002');
  if e_ext = 0210 then call symput('CDATE','31/10/2002');
  if e_ext = 0211 then call symput('CDATE','30/11/2002');
  if e_ext = 0212 then call symput('CDATE','31/12/2002');
  if e_ext = 0301 then call symput('CDATE','31/01/2003');
  if e_ext = 0302 then call symput('CDATE','28/02/2003');
  if e_ext = 0303 then call symput('CDATE','31/03/2003');
  if e_ext = 0304 then call symput('CDATE','30/04/2003');
  if e_ext = 0305 then call symput('CDATE','31/05/2003');
  if e_ext = 0306 then call symput('CDATE','30/06/2003');

run;

/*****PROCESS A: Read Current Data File To Get Patient Record *****/

data esis_v.d&esisext.;
  %let _EFIERR_ = 0;      /* clear ERROR detection macro variable */
  infile "f:\data\esis_data\esis_v\rawdata\&campus.\D&esisext.&campus..TXT"
delimiter = ',' MISSOVER DSD;

  format key $8.;          /* 1 */
  format urno $10.;       /* 2 */
  format mednum $11. ;    /* 3 */
  format medsuf $upcase3. ; /* 4 */
  format waitno $2. ;     /* 5 */
  format dob $8.;        /* 6 */

```

```

format sex $1. ; /* 7 */
format pstcode $4. ; /* 8 */
format locality $30. ; /* 9 */
format ppproc z3. ; /* 10 writes leading zeros to length of 3*/
format surgspec z2. ; /* 11 writes leading zeros to length of 2*/
format urgency $1. ; /* 12 */
format planstay $1. ; /* 13 */
format rg_dt $8. ; /* 14 */
format refsourc $1. ; /* 15 */
format refhosp $4. ; /* 16 */
format ug_dt $8. ; /* 17 */
format ugup_dt $8. ; /* 18 */
format old_urg $1. ; /* 19 */
format readstat $1. ; /* 20 */
format rd_dt $8. ; /* 21 */
format nrreason $1. ; /* 22 */
format bk_dt $8. ; /* 23 */
format bookno z2.0 ; /* 24 */
format rbreason $1. ; /* 25 */
format sch_dt $8. ; /* 26 */
format rm_dt $8. ; /* 27 */
format rmreason $1. ; /* 28 */
format insure $1. ; /* 29 */
format transdes $4. ; /* 30 */
format nr_total z4.0 ; /* 31 */
format nr_assig z4. ; /* 32 */
format nrtot_ui z4. ; /* 33 */
format hi_post z2.0 ; /* 34 */
format proc_dt $8. ; /*35 Date of Procedure */
format c_site $4. ; /*36 intended treatment campus*/

```

```

informat key $8. ;
informat urno $10. ;
informat mednum $11. ;
informat medsuf $3. ;
informat waitno $2. ;
informat dob $8. ;
informat sex $1. ;
informat pstcode $4. ;
informat locality $30. ;
informat ppproc 3.0 ;
informat surgspec 2.0 ;
informat urgency $1. ;
informat planstay $1. ;
informat rg_dt $8. ;
informat refsourc $1. ;
informat refhosp $4. ;
informat ug_dt $8. ;
informat ugup_dt $8. ;
informat old_urg $1. ;
informat readstat $1. ;
informat rd_dt $8. ;
informat nrreason $1. ;
informat bk_dt $8. ;
informat bookno 2.0 ;
informat rbreason $1. ;
informat sch_dt $8. ;
informat rm_dt $8. ;
informat rmreason $1. ;
informat insure $1. ;
informat transdes $4. ;
informat nr_total 4.0 ;
informat nr_assig 4. ;
informat nrtot_ui 4. ;
informat hi_post 2.0 ;
informat proc_dt $8. ;
informat c_site $4. ;

```

```

input
    key $
    urno $
    mednum $
    medsuf $
    waitno $
    dob/*birthdat*/ $
    sex $
    pstcode $
    locality $
    ppproc
    surgspec $
    urgency $
    planstay $
    rg_dt $ /*reg_date*/
    refsourc $
    refhosp $
    ug_dt $/*urg_date*/
    ugup_dt $ /*urgup_dt*/
    old_urg $
    readstat $
    rd_dt $ /*readdate*/
    nrreason $
    bk_dt $ /*bookdate*/
    bookno
    rbreason $
    sch_dt $ /*sch_date */
    rm_dt $ /*rem_date*/
    rmreason $
    insure $
    transdes $
    nr_total
    nr_assig
    nrtot_ui
    hi_post
    proc_dt $ /*dt_proc*/
    c_site;

    If _ERROR_ then          /* ERROR detection */
        call symput('_EFIERR_',1);

run;

/***** Merge in cdate date & campus & num records with the data file*****/

data esis_v.d&esisext.;
    if _n_=1 then set esis_v.con2;
    set esis_v.d&esisext.;
run;

/***** End Merge *****/

data esis_v.e&esisext.(drop =dob rg_dt ug_dt ugup_dt rd_dt bk_dt sch_dt rm_dt
proc_dt);

    set esis_v.d&esisext.;

```

\*\*\*\*\* PROCESS B: Create Derived Data Fields \*\*\*\*\*

```

epid = compbl(trimn(campus)||trim(left(urno))||trimn(waitno)) ;

format birthdat date9. reg_date date9. urg_date date9. urgup_dt date9.;
format readdate date9. bookdate date9. sch_date date9. rem_date date9. dt_proc
date9.;

if length(dob) = 8 then do;
  d = substr(dob,1,2);
  m = substr(dob,3,2);
  y = substr(dob,5,4);
  if substr(dob,1,4) = '0000' then do;
    est_dob = 'Y';
    birthdat = mdy(1,1,y);
  end;
  else do;
    est_dob = 'N';
    birthdat = mdy(m,d,y);
  end;
end;

if length(rg_dt) = 8 then do;
  d = substr(rg_dt,1,2);
  m = substr(rg_dt,3,2);
  y = substr(rg_dt,5,4);
  reg_date = mdy(m,d,y);
  if rg_dt ne '' and reg_date = . then regdt_er = 'Y';
end;

if length(ug_dt) = 8 then do;
  d = substr(ug_dt,1,2);
  m = substr(ug_dt,3,2);
  y = substr(ug_dt,5,4);
  urg_date = mdy(m,d,y);
  if ug_dt ne '' and urg_date = . then urg_er = 'Y';
end;

if length(ugup_dt) = 8 then do;
  d = substr(ugup_dt,1,2);
  m = substr(ugup_dt,3,2);
  y = substr(ugup_dt,5,4);
  urgup_dt = mdy(m,d,y);
  if ugup_dt ne '' and urgup_dt = . then urgup_er = 'Y';
end;

if length(rd_dt) = 8 then do;
  d = substr(rd_dt,1,2);
  m = substr(rd_dt,3,2);
  y = substr(rd_dt,5,4);
  readdate = mdy(m,d,y);
  if rd_dt ne '' and readdate = . then rddt_er = 'Y';
end;

if length(bk_dt) = 8 then do;
  d = substr(bk_dt,1,2);
  m = substr(bk_dt,3,2);
  y = substr(bk_dt,5,4);
  bookdate = mdy(m,d,y);
  if bk_dt ne '' and bookdate = . then bkdt_er = 'Y';
end;

if length(sch_dt) = 8 then do;
  d = substr(sch_dt,1,2);
  m = substr(sch_dt,3,2);
  y = substr(sch_dt,5,4);
  sch_date = mdy(m,d,y);
  if sch_dt ne '' and sch_date = . then schdt_er = 'Y';
end;

if length(rm_dt) = 8 then do;
  d = substr(rm_dt,1,2);
  m = substr(rm_dt,3,2);
  y = substr(rm_dt,5,4);
  rem_date = mdy(m,d,y);
  if rm_dt ne '' and rem_date = . then rmdt_er = 'Y';
end;

```

```

end;
if length(proc_dt) = 8 then do;
    d = substr(proc_dt,1,2);
    m = substr(proc_dt,3,2);
    y = substr(proc_dt,5,4);
    dt_proc = mdy(m,d,y);
    if proc_dt ne '' and dt_proc = . then proc_er = 'Y';
end;

if dt_proc > cdate then dt_proc = .; /* 9/8/2002 Decided in meeting (Tim, Jane,
Greg, Anna, Mark & Janelle) that any procedure dates > census dates would be
overwritten with a missing. This is because snapshot of esis taken at census
date and any dates after this will be assumptions/guesses. */
    else if proc_dt ne '' and dt_proc = . then proc_er = 'Y';
end;

run;

/***** To count for number of records in the data file *****/
data esis_v.e&esisext.;
    set esis_v.e&esisext.;
    count = 1;
run;

proc summary data=esis_v.e&esisext. nway missing;
    class campus;
    var count;
    output out=tcm (drop=_type_ _freq_) n(count)=d_n_rec;
run;

data keep4later;
    set tcm;
run;

data esis_v.e&esisext.;
    if _n_ = 1 then set tcm;
    set esis_v.e&esisext.;
run;

/* End Record Count */
proc sql;
    create table esis_v.e&esisext. as
    select distinct A.*
    from esis_v.e&esisext. a;

/* To check if a different waitno exists with the same */
/* Patient Identifier and the same ppproc */

proc summary data=esis_v.e&esisext. nway missing;
    where ppproc not in
(10,30,34,35,50,70,90,91,94,99,100,110,112,113,114,115,117,130,
133,160,170,180,190,213,214,218,223,224);
/*To exclude those ppprocs which may legitimately be recorded more than once*/
    class campus urno ppproc;
    var count;
    output out=tcml (drop=_type_ _freq_) n(count)=ppp_dup;
run;

proc sort data = esis_v.e&esisext.;
    by urno ppproc;

proc sort data = work.tcml out = work.tcml;
    by urno ppproc;

data esis_v.e&esisext.;
    merge esis_v.e&esisext.(in=ina) work.tcml (in=inb);
    by urno ppproc;
    if ina then output;
run;
/***** End ppp check *****/

```

```

/* To check if unique keys have been duplicated */
proc summary data=esis_v.e&esisext. nway missing;
  class campus key;
  var count;
  output out=key_dup (drop=_type_ _freq_) n(count)=key_dup;
run;

proc sort data = esis_v.e&esisext.;
  by key;
proc sort data = key_dup;
  by key;

data esis_v.e&esisext.;
  merge esis_v.e&esisext. (in = ina) key_dup (in=inb);
  by key;
  if ina then output;
run;
/** END DUPLICATE KEY CHECK **/

%macro previous;
%if &firstsub = 'N' %then %do; /*No previous month if first ever submission*/
%if &e_ext ne 0207 %then %do;
/***** READ PREVIOUS MONTH DATA FILE FOR MONTH TO MONTH COMPARISON *****/
data e_old(keep=key urno waitno sexold birthold procold regold urgold plansold
wstatold
      booknold hiposold nrtotold nrassold nruiold oldurgdt specold
rbrsnold urgupold rstatold rdateold dsetold);
  set esislst.e&esislst.&campus.(drop = birthold sexold procold regold
      specold urgold plansold wstatold hiposold oldurgdt
      nrtotold nrassold nruiold booknold rbrsnold urgupold rstatold
rdateold);

  format ppproc 4. bookno 2. surgspec 2. hi_post 2.;
  format nr_total 4. nr_assig 4. nrtot_ui 4.;
  format nruiold 4. booknold 2. specold 2. procold 3. nrtotold 4.;
  format nrassold 4. hiposold 2. rbrsnold $1. rstatold $1.;

  rename sex      = sexold;
  rename birthdat = birthold;
  rename ppproc   = procold;
  rename reg_date = regold;
  rename surgspec = specold;
  rename urgency  = urgold;
  rename planstay = plansold;
  rename waitstat = wstatold;
  rename bookno   = booknold;
  rename hi_post  = hiposold;
  rename urg_date = oldurgdt;
  rename rbreason = rbrsnold;
  rename urgup_dt = urgupold;
  rename readstat = rstatold;
  rename readdate = rdateold;
  if nr_total = . then nr_total = 0;
  if nr_assig = . then nr_assig = 0;
  if nrtot_ui = . then nrtot_ui = 0;
  rename nr_total = nrtotold;
  rename nr_assig = nrassold;
  rename nrtot_ui = nruiold;
  dsetold = 'old';
  if rmreason ne 'E';
run;
%end;
%else %do;

```

```

data e_old(keep=key urno waitno sexold birthold procold regold urgold plansold
wstatold
            booknold hiposold nrtotold nrassold nruiold oldurgdt specold
rbrsnold urgupold rstatold rdateold dsetold);
  set esislast.e&esislast.&campus.(drop = birthold sexold procold regold
specold urgold plansold wstatold hiposold oldurgdt
nrtotold nrassold nruiold booknold);

format ppproc 4. bookno 2. surgspec 2. hi_post 2.;
format nr_total 4. nr_assig 4. nrtot_ui 4.;
format nruiold 4. booknold 2. specold 2. procold 3. nrtotold 4.;
format nrassold 4. hiposold 2. rbrsnold $1. rstatold $1.;

rename sex      = sexold;
rename birthdat = birthold;
rename ppproc   = procold;
rename reg_date = regold;
rename surgspec = specold;
rename urgency  = urgold;
rename planstay = plansold;
rename waitstat = wstatold;
rename bookno   = booknold;
rename hi_post  = hiposold;
rename urg_date = oldurgdt;
rename rbreason = rbrsnold;
rename urgup_dt = urgupold;
rename pat_stat = rstatold;
rename sr_date  = rdateold;
if nr_total = . then nr_total = 0;
if nr_assig = . then nr_assig = 0;
if nrtot_ui = . then nrtot_ui = 0;
rename nr_total = nrtotold;
rename nr_assig = nrassold;
rename nrtot_ui = nruiold;
dsetold = 'old';
if rmreason ne 'E';
run;
%end;
%end;
%if &firstsub. = 'Y' %then %do;
  data e_old;
    format key $10. urno $10. waitno $1. birthold 9. sexold $1.;
    format ppproc 4. bookno 2. surgspec 2. hi_post 2.;
    format nr_total 4. nr_assig 4. nrtot_ui 4.;
    format nruiold 4. booknold 2. specold 2. procold 3. nrtotold 4.;
    format nrassold 4. hiposold 2. rbrsnold $1. rstatold $1.;
  run;
%end;
%mend;

%previous;
proc sort data=e_old;
  by key urno waitno sexold birthold;
run;

/***** END E_OLD *****/

data e_new;
  set esis_v.e&esisext.;
  dset = 'new';
run;

proc sort data=e_new;
  by key urno waitno sex birthdat;
run;

data e_comb;
  merge e_old e_new;
  by key;
run;

```

```

/** READ JUNE DATA FILE FOR URGENCY REASSIGNMENT DATE COMPARISONS
    Added 1 July 2001
    ***/

data e_june;
  set esis_ref.final_nodups_june01; /*list of unduplicated june01 keys, campus,
urno's and urgdates and nr_assig */
  if &campus. eq '5510' then do;
    if campcode in ('1170','2111','2112','1220');
  end;
  else do;
    if campcode = &campus.;
  end;
  keep key urno Junurgdt Junnrass;
  if nr_assig = . then nr_assig = 0;
run;

proc sort data=e_june;
  by key urno ;
run;

proc sort data=e_comb;
  by key urno waitno;
run;

proc sql;
  create table e_comp as
  select a.*, b.junurgdt, b.junnrass
  from e_comb a left join e_june b
  on a.key = b.key;

data e_comp (drop = d_camp d_site);
  set e_comp;
  if &campus. in ('5510','5520') then do;
    d_camp = campus;
    d_site = c_site;
    campus = d_site;
    c_site = d_camp;
  end;
run;

/*****
/***** PROCESS C: Data File Edits *****/
/*****

data esis_v.z&esisext. esis_v.r&esisext.&campus. esis_v.x&esisext.&campus.;

  set e_comp ;
  format type type1 type2 8.0;
  type1 = 0; type2 = 0; type = 0; ok = 0;
  format row_number 5.0;
  row_number = _n_;
  array mess{20} $80. ;          /* set up array for error mess
    */

```

```

/*****
/* S053: Record Not Removed Last Month Missing From File */
/*****

  if (wstatold = 'Y') AND (sex = ' ') AND (birthdat = .) then do;
    type1 = type1 + 1;
    type = type + 1;
    mess{type} = 'Rejection: S053 Record not removed last month missing from
file';
  end;

if dset eq 'new' then do; /* CHECKS AGAINST NEW DATA ONLY */

/*****
/* S001: Census Date Not Valid */
/*****

  if (cdate ne rdate) and dset ne 'old' then do;
    type1 = type1 + 1;
    type=type + 1;
    mess{type} = 'Rejection: S001 Census Date Not Valid';
  end;

/*****
/* S002: No Header Record Found */
/*****

  if ((cdate = .) or (campus = '') or (n_rec = .)) and dset ne 'old' then do;
    type1 = type1 + 1;
    type=type + 1;
    mess{type} = 'Rejection: S002 No Header Record Found';
  end;

/*****
/* S003: Incorrect Campus/Health Services Code */
/*****

  if ((campus = '') or
      (campus not in ('1010','1590','1031','2010','2050','1020',
                     '1050','2111','2220','1121','1210','1170',
                     '2440','1280','1191','1331','1240','1230',
                     '1360','1450','1390','1150','1580','1180',
                     '1170','1220','2111','2112','1460')))
    and dset ne 'old' then do;
    type1 = type1 + 1;
    type=type + 1;
    mess{type} = 'Rejection: S003 Incorrect Campus/Health Service Code';
  end;

/*****
/* S004: Total Number of Records Not Equal To Total Patient Records */
/*****

  if n_rec ne d_n_rec then do;
    type1 = type1 + 1;
    type=type + 1;
    mess{type} = 'Rejection: S004 Total Number Of Records Not Equal To Total
Patient Records';
  end;

```

```

/*****/
/* S005: Control Filename Incorrect */
/* Submission officer to manually check filename and inform the hospital of this error */
/*****/

/*****/
/* S006: Control File Content Incorrect */
/* As we only read/input the 1st 3 fields from the control file */
/* and ignore the others, it will not become a data problem */
/*****/

/*****/
/* S051: Data Filename Incorrect */
/* Submission officer to manually check filename and inform the hospital of this */
/*****/

/*****/
/* S052: Record Dates After Census Data - MODIFIED 1 JULY 2002 */
/*****/
    if ((cdate < bookdate) or (cdate < readdate) or (cdate < urg_date)
        or (cdate < reg_date) or (cdate < rem_date) or (cdate < urgup_dt) or
            (cdate < dt_proc)) then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S052 Record Dates After Census Date';
    end;

/*****/
/* S053: See Previous Month Comparison Edits */
/*****/

/*****/
/* S060: Blank/Invalid Unique Key */
/*****/
tot=0;

    do a = 1 to length(key);
        tot =
indexc(substr(key,a,1), '0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz
tuvxyz') + tot;
    end;

    if (tot/length(key) ne 1) or ((length(key) > 8) or
        (length(key) < 8) or key = '') then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S060 Invalid/Blank Unique Key';
    end;

```

```

/*****
/* S061: Unique Key For Same Patient Already On List */
/*****
    if key_dup > 1 then do;
        type1 = type1 + 1;
        type = type + 1;
        mess(type) = 'Rejection: S061 Unique Key for same patient already on list';
    end;

/*****
/* S065: Blank Patient Identifier */
/*****
    if urno = '' then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S065 Blank Patient Identifier';
    end;

/*****
/* S066: Invalid Patient Identifier */
/* This edit not checking for leading zero's b/c incl in format */
/*****
    tot=0;

    do a = 1 to length(urno);
        tot =
indexc(substr(urno,a,1), '0123456789ABCDEFGHIJKLMNopqrstuvwxyz') + tot;
    end;

    if (tot/length(urno) ne 1) or ((length(urno) > 10) or
        (length(urno) < 10)) then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S066 Invalid Patient Identifier';
    end;

/*****
/* S080: Blank Medicare Number */
/*****
    if (rem_date ne .) and (rmreason eq 'W') and
        ((medsuf = '') or (medsuf not in ('C-U', 'N-E', 'P-N', 'BAB'))) and (mednum =
        '') then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S080 Blank Medicare Number';
    end;

```

```

/*****/
/* S081: Invalid Medicare Number */
/*****/

tot = 0;

do a = 1 to 11;
  tot = indexc(substr(mednum,a,1),' 0123456789') + tot;
end;

if (rem_date ne . and rmreason eq 'W') then do;
  checkdg = MOD((substr(mednum,1,1)*1) + (substr(mednum,2,1)*3) +
    (substr(mednum,3,1)*7) + (substr(mednum,4,1)*9) +
    (substr(mednum,5,1)*1) +
    (substr(mednum,6,1)*3) + (substr(mednum,7,1)*7) +
    (substr(mednum,8,1)*9),10);
  if tot ne 11 or (substr(mednum,9,1) ne checkdg) then do;
    type1 = type1 + 1;
    type=type + 1;
    mess{type} = 'Rejection: S081 Invalid Medicare Number';
  end;
end;

/*****/
/* S082: Medicare Code '0' and Age Is Greater Than Six Months */
/*****/

if (rem_date ne .) and (rem_date >= '01Aug2000'd) and (substr(mednum,11,1) =
'0')
  and (intck('month', birthdat, cdate) > 6) then do;
  type2 = type2 + 1;
  type=type + 1;
  mess{type} = 'Warning: S082 Invalid Combination Medicare Code & DOB';
end;

/*****/
/* S083: Medicare Code '0' and Age Is Greater Than Twelve Months */
/*****/

if (rem_date ne .) and (rem_date >= '01Aug2000'd) and
  intck('day', birthdat, cdate) >= 365 and
  substr(mednum,11,1) = '0' then do;
  type2 = type2 + 1;
  type=type + 1;
  mess{type} = 'Warning: S083 Invalid Combination Medicare Code & DOB';
end;

/*****/
/* S087: Blank Medicare Suffix */
/*****/

if reg_date = > '01Jul2000'd and (rem_date ne .) and
  (rmreason eq 'W') and (medsuf = '') then do;
  type1 = type1 + 1;
  type=type + 1;
  mess{type} = 'Rejection: S087 Blank Medicare Suffix';
end;

```

```

/*****
/* S088: Invalid Medicare Suffix
*/
/*****
if reg_date = > '01Jul2000'd and (rem_date ne .) and rmreason eq 'W'
and medsuf ne ''
and (substr(medsuf,1,1) not in
('A','B','C','D','E','F','G','H','I','J',
'K','L','M','N','O','P','Q','R','S','T',
'U','V','W','X','Y','Z'))
or (substr(medsuf,2,1) not in
('A','B','C','D','E','F','G','H','I','J',
'K','L','M','N','O','P','Q','R','S','T',
'U','V','W','X','Y','Z','-',' ',''))
or (substr(medsuf,3,1) not in
('A','B','C','D','E','F','G','H','I','J',
'K','L','M','N','O','P','Q','R','S','T',
'U','V','W','X','Y','Z',' ',' ','-')) then do;
  typel = typel + 1;
  type=type + 1;
  mess{type} = 'Rejection: S088 Invalid Medicare Suffix';
end;

/*****
/* S090: Blank Sex Code
*/
/*****
if sex = '' then do;
  typel = typel + 1;
  type=type + 1;
  mess{type} = 'Rejection: S090 Blank Sex Code';
end;

/*****
/* S091: Invalid Sex Code
*/
/*****
if sex not in ('1','2','3') then do;
  typel = typel + 1;
  type=type + 1;
  mess{type} = 'Rejection: S091 Invalid Sex Code';
end;

/*****
/* S092: Indeterminate Sex, Age Greater Than Or Equal To Ninety Days
*/
/*****
if sex = '3' and intck('day',birthdat,reg_date) >= 90 then do;
  typel = typel + 1;
  type=type + 1;
  mess{type} = 'Rejection: S092 Indeterminate Sex, Age >= 90 Days';
end;

/*****
/* S093: Indeterminate Sex - Check
*/
/*****
if sex = '3' and intck('day',birthdat,reg_date) < 90 then do;
  type2 = type2 + 1;
  type=type + 1;
  mess{type} = 'Warning: S093 Indeterminate Sex - Check';
end;

```

```

/*****/
/* S095: Blank Date Of Birth */
/*****/
    if birthdat = . then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S095 Blank Date of Birth';
    end;

/*****/
/* S096: Invalid Date Of Birth */
/*****/
    dobcc = substr(put(birthdat, ddmmyy10.),7,2);
    if birthdat ne . and dobcc not in ('18','19','20') then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S096 Invalid Date of Birth';
    end;

/*****/
/* S098: Age Greater Than 120 Years */
/*****/
    if intck('year',birthdat,reg_date) >= 120 then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S098 Age Greater Than 120 Years';
    end;

/*****/
/* S099: Registration Date Before Date Of Birth */
/*****/
    if reg_date ne . and reg_date < birthdat then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S099 Registration Date Before Date of Birth';
    end;

/*****/
/* S110: Blank Waiting Number */
/*****/
    if waitno = '' then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S110 Blank Waiting Number';
    end;

```

```

/*****
/* S111: Invalid Waiting Number */
/*****
  if waitno ne '' and
    (substr(waitno,1,1) not in ('0','1','2','3','4','5','6','7','8','9') or
     substr(waitno,2,1) not in ('0','1','2','3','4','5','6','7','8','9',' '))
  then do;
    type1 = type1 + 1;
    type=type + 1;
    mess{type} = 'Rejection: S111 Invalid Waiting Number';
  end;

  if waitno ne '' and (waitno = '00' or waitno = '0') then do;
    type1 = type1 + 1;
    type=type + 1;
    mess{type} = 'Rejection: S111 Invalid Waiting Number';
  end;

/***** SLA assignment *****/

format
  sla_1 sla_2 sla_3 sla_4 sla_5 $5.
  pw_1 pw_2 pw_3 pw_4 pw_5 8.
  psloc_ok loc_ok pst_ok $1.
  pst_loc $34.;

/*****
/* S115: Area of Residence Error */
/*****
  if (((locality = '') and (pstcode ne 1000 and
    pstcode ne 8888 and pstcode ne 9988))or ((locality ne '') and
    (pstcode = 1000 or pstcode = 9988))) then do;
    type1 = type1 + 1;
    type=type + 1;
    mess{type} = 'Rejection: S115 Area of Residence Error';
  end;

/*****
/* S116: Invalid Locality (part 1) */
/*****
  if (locality) ne '' and pstcode=: '3' then do;
    h_label=put(locality,loc00fma.);
    loc_ok = scan(h_label,1);
    if loc_ok ne '1' then do;
      type1 = type1 + 1;
      type=type + 1;
      mess{type} = 'Rejection: S116 Invalid Locality';
    end;
  end;

/*****
/* S116: Invalid Locality (part 2) */
/*****
  if reg_date = > '01Jul2000'd and verify(uppercase(locality),
    "1234567890QWERTYUIOPASDFGHJKLZXCVBNM'.- ") ne 0 then do;
    type1 = type1 + 1;
    type=type + 1;
    mess{type} = 'Rejection: S116 Invalid Locality';
  end;

```

```

/*****/
/* S117: Blank Locality (part 2) Deleted 1 July 2002 */
/*****/

/*****/
/* S120: Blank Postcode */
/*****/
    if pstcode = '' then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S120 Blank Postcode';
    end;

/*****/
/* S121: Invalid Postcode */
/*****/
    p_label=put(pstcode,pc00fma.);
    pst_ok = scan(p_label,1);
    if reg_date = > '01Jul2000'd and pst_ok ne '1' then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S121 Invalid Postcode';
    end;

    if pst_ok = '1' AND loc_ok = '1' then do;
        pst_loc=trim(pstcode)||trim(locality);
        g_label=put(pst_loc,sla00fma.);
        psloc_ok= scan(g_label,1,',');
        sla_1= scan(g_label,2,','); pw_1= scan(g_label,3,',');
        sla_2= scan(g_label,4,','); pw_2= scan(g_label,5,',');
        sla_3= scan(g_label,6,','); pw_3= scan(g_label,6,',');
        sla_4= scan(g_label,8,','); pw_4= scan(g_label,9,',');
        sla_5= scan(g_label,10,','); pw_5= scan(g_label,11);
    end;

/*****/
/* S122: Invalid Combination Postcode and Locality */
/*****/
    if reg_date = > '01Jul2000'd and psloc_ok ne '1' and pstcode='3' then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S122 Invalid Combination Postcode And Locality';
    end;

/***** SLA ASSIGNMENT WILL GO HERE *****/

    if psloc_ok = '1' then do;
        t1 = pw_1;
        t2 = t1 + pw_2;
        t3 = t2 + pw_3;
        t4 = t3 + pw_4;
        t5 = t4 + pw_5;

        pw = 100*ranuni(4);

        if 0 <= pw <= t1 then sla00 = sla_1;
        else if t1 < pw <= t2 then sla00 = sla_2;
        else if t2 < pw <= t3 then sla00 = sla_3;
        else if t3 < pw <= t4 then sla00 = sla_4;

```

```

else if t4 < pw <= 100 then sla00 = sla_5;
else sla00='error';

end;

/***** end SLA assignment *****/

/*****
/* S133: Blank Principal Prescribed Procedure */
/*****
if ppproc = '' then do;
  typel = typel + 1;
  type=type + 1;
  mess{type} = 'Rejection: S133 Blank Principal Prescribed Procedure';
end;

/*****
/* S134: Invalid Principal Prescribed Procedure */
/*****
if reg_date ne . and reg_date < '01Jul2000'd and ppproc ne '' then do;
  if ppproc ne '' and ppproc not in
    ('01', '02', '03', '04', '05', '06', '07', '08', '10', '11',
     '12', '13', '14', '15', '16', '17', '18', '19', '20', '21',
     '22', '30', '32', '33', '34', '35', '36', '37', '38', '39',
     '40', '41', '42', '50', '52', '54', '55', '56', '57', '58',
     '59', '60', '61', '62', '70', '71', '72', '73', '74', '75',
     '77', '79', '81', '82', '83', '84', '85', '86', '90', '91',
     '92', '93', '94', '95', '96', '97', '98', '99',
     '100', '110', '111', '112', '113', '114', '115', '116', '117',
     '118', '119', '120', '121', '130', '131', '132', '133', '134',
     '135', '136', '137', '138', '140', '141', '142', '143', '150',
     '160', '161', '162', '163', '164', '165', '166', '167', '168',
     '169', '170', '171', '172', '173', '174', '175', '176', '180',
     '181', '182', '183', '184', '185', '186', '187', '188', '190',
     '191', '192', '193', '194', '195', '196', '197', '198', '199',
     '202', '203', '204', '205', '206', '207', '208', '209', '210',
     '211', '212', '213', '214', '215', '216', '217', '218', '219',
     '220', '221', '222', '223', '224', '225', '226') then do;
    typel = typel + 1;
    type=type + 1;
    mess{type} = 'Rejection: S134 Invalid Principal Prescribed Procedure';
  end;
end;

/*****
/* S135: Patient Already On Waiting List For Same PPP */
/*****
if ppp_dup > 1 then do;
  typel = typel + 1;
  type=type + 1;
  mess{type} = 'Rejection: S135 Patient Already On Waiting List For Same
PPP';
end;

/*****
/* S136: See Previous Month Comparison Edits */
/*****

```

```

/*****/
/* S146: Blank Surgical Specialty */
/*****/
    if surgspec = '' then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S146 Blank Surgical Specialty';
    end;

/*****/
/* S147: Invalid Surgical Specialty */
/*****/
    if surgspec not in ('01','02','03','04','05','06','07','08',
        '09','10','11') then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S147 Invalid Surgical Specialty';
    end;

/*****/
/* S148: See Previous Month Comparison Edits */
/*****/

/*****/
/* S160: Blank Clinical Urgency Category */
/*****/
    if urgency = '' then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S160 Blank Clinical Urgency Category';
    end;

/*****/
/* S161: Invalid Clinical Urgency Category */
/*****/
    if urgency ne '' and urgency not in ('1','2','3') then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S161 Invalid Clinical Urgency Category';
    end;

/*****/
/* S162: Clinical Urgency Reassignment Incomplete */
/*****/
    if (old_urg ne '' and urg_date = .) or
        (old_urg = '' and urg_date ne .) then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S162 Clinical Urgency Reassignment Incomplete';
    end;

```

```

/*****
/* S166: Blank Planned Length Of Stay */
/*****
    if planstay = '' then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S166 Blank Planned length of Stay';
    end;

/*****
/* S167: Invalid Planned Length Of Stay */
/*****
    if planstay ne '' and planstay not in ('1','2') then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S167 Invalid Planned length of Stay';
    end;

/*****
/* S168: Blank Registration Date */
/*****
    if reg_date = . then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S168 Blank Registration Date';
    end;

/*****
/* S169: Invalid Registration Date */
/*****
    regcc = substr(put(reg_date, ddmmyy10.),7,2);
    if (reg_date ne . and regcc not in ('19','20')) or (regdt_er = 'Y') then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S169 Invalid Registration Date';
    end;

/*****
/* S170: See Previous Month Comparison Edits */
/*****

/*****
/* S171: See Previous Month Comparison Edits */
/*****

/*****
/* S172: Reg Date = Urgency Reassignment Date or Date OF Last Clinical Urgency Increase - DELETED
   1 JULY 2002 */
/*****

/*****
/* S173: Date of Reg = Status Reassignment Date - DELETED 1 JULY 2002 */
/*****

```

```

/*****/
/* S174: New Record, Old Registration Date */
/*****/
if dset = 'new' and reg_date < intck('year',cdate, 1) and
  refsourc ne '2' then do;
  type1 = type1 + 1;
  type = type + 1;
  mess(type) = 'Rejection: S174 New Record, Old registration date';
end;

/*****/
/* S192: Blank Source Of Referral */
/*****/
if reg_date = > '01Jul2000'd and refsourc = '' then do;
  type1 = type1 + 1;
  type=type + 1;
  mess{type} = 'Rejection: S192 Blank Source of Referral';
end;

/*****/
/* S193: Invalid Source Of Referral */
/*****/
if reg_date = > '01Jul2000'd and (refsourc) ne '' and (refsourc) not in
('1','2','3','4','5') then do;
  type1 = type1 + 1;
  type=type + 1;
  mess{type} = 'Rejection: S193 Invalid Source of Referral';
end;

/*****/
/* S194: Blank Referring Hospital */
/*****/
if reg_date = > '01Jul2000'd and refsourc = '2' and refhosp = '' then do;
  type1 = type1 + 1;
  type=type + 1;
  mess{type} = 'Rejection: S194 Blank Referring Hospital';
end;

/*****/
/* S195: Invalid Referring Hospital */
/*****/
if reg_date = > '01Jul1999'd and (refsourc = '2' and refhosp ne '')
and refhosp not in
('1010','1590','1031','2010','2050','1020','1050','2110','2111',
'2220','1121','1210','1170','2440','1280','1191','1331',
'1240','1230','1360','1450','1390','1150','1580','1180','1550','1460')
then do;
  type1 = type1 + 1;
  type=type + 1;
  mess{type} = 'Rejection: S195 Invalid Referring Hospital';
end;

```

```

/*****
/* S196: Referring Hospital Conflict */
/*****
  if reg_date = > '01Jul2000'd and refsourc not in ('2') and refhosp ne '' then
do;
  type1 = type1 + 1;
  type=type + 1;
  mess{type} = 'Rejection: S196 Referring Hospital Conflict';
end;

/*****
/* S199: Invalid Date of Last Clinical Urgency Increase */
/*   ADDED 1 JULY 2001 */
/*****
  urgcc = substr(put(urgup_dt, ddmmyy10.),7,2);
  if (urgup_dt ne . and urgcc not in ('19','20')) or (urgup_er eq 'Y') then do;
  type1 = type1 + 1;
  type=type + 1;
  mess{type} = 'Rejection: S199 Invalid Date of Last Clinical Urgency
Increase';
end;

/*****
/* S200: Clinical Urgency Increase Incomplete */
/*****
  if (urgup_dt ne '' and old_urg = '' )or (urgup_dt = '' and old_urg > urgency
and old_urg ne . and urg_date >= '01Jul2001'd)
  then do;
  type1 = type1 + 1;
  type = type + 1;
  mess(type) = 'Rejection: S200 Clinical urgency increase incomplete';
end;

/*****
/* S261: Prev Urgency Category Equals Clinical Urgency */
/*****
  if old_urg ne '' and urgency ne '' and old_urg = urgency then do;
  type1 = type1 + 1;
  type=type + 1;
  mess{type} = 'Rejection: S261 Prev Urgency Category = Clinical Urgency';
end;

/*****
/* S262: Invalid Previous Urgency Category */
/*****
  if urg_date ne . and old_urg ne '' and
old_urg not in ('1','2','3') then do;
  type1 = type1 + 1;
  type=type + 1;
  mess{type} = 'Rejection: S262 Invalid Previous Urgency Category';
end;

```

```

/*****/
/* S263: Invalid Urgency Reassignment Date */
/*****/
urgcc = substr(put(urg_date, ddmmyy10.),7,2);
if (urg_date ne . and urgcc not in ('19','20')) or (urg_er eq 'Y') then do;
    type1 = type1 + 1;
    type=type + 1;
    mess{type} = 'Rejection: S263 Invalid Urgency Reassignment Date';
end;

/*****/
/* S264: Blank Readiness Status */
/*****/
if readstat = ' ' then do;
    type1 = type1 + 1;
    type=type + 1;
    mess{type} = 'Rejection: S264 Blank Readiness Status';
end;

/*****/
/* S265: Invalid Readiness Status */
/*****/
if readstat ne ' ' and readstat not in ('N','R') then do;
    type1 = type1 + 1;
    type=type + 1;
    mess{type} = 'Rejection: S265 Invalid Readiness Status';
end;

/*****/
/* S266: Invalid Readiness Change Date */
/*****/
srcc = substr(put(readdate, ddmmyy10.),7,2);
if (readdate ne . and srcc not in ('19','20')) or (rddt_er eq 'Y') then do;
    type1 = type1 + 1;
    type=type + 1;
    mess{type} = 'Rejection: S266 Invalid Readiness Change Date';
end;

/*****/
/* S267: Invalid Reason For NRFC Status */
/*****/
if readstat = 'N' and nrreason ne ' ' and nrreason not in ('C','D','S') then
do;
    type1 = type1 + 1;
    type=type + 1;
    mess{type} = 'Rejection: S267 Invalid Reason For NRFC status';
end;

/*****/
/* S268: Patient NRFC, No Reason For NRFC */
/*****/
if readstat = 'N' and nrreason = ' ' then do;
    type1 = type1 + 1;
    type=type + 1;
    mess{type} = 'Rejection: S268 Patient NRFC, No Reason For NRFC';
end;

```

```

/*****
/* S269: Reason for NRFC Patient Current RFC                                     */
/*****
  if readstat = 'R' and nrreason ne '' then do;
    typel = typel + 1;
    type=type + 1;
    mess{type} = 'Rejection: S269 Reason For NRFC Patient Current RFC';
  end;

/*****
/* S274: Blank Booking Number                                               */
/*****
if bookno = '' then do;
  typel = typel + 1;
  type=type + 1;
  mess{type} = 'Rejection: S2740 Blank Booking Number Code';
end;

/*****
/* S275 Invalid Booking Date                                               */
/*****
  bookcc = substr(put(bookdate, ddmmyy10.),7,2);
  if (bookdate ne . and bookcc not in ('19','20')) or (bkdt_er eq 'Y') then do;
    typel = typel + 1;
    type=type + 1;
    mess{type} = 'Rejection: S275 Invalid Booking Date';
  end;

/*****
/* S276: Booking Date Before Registration Date                             */
/*****
  if reg_date = > '01Jul1999'd and bookdate ne . and bookdate < reg_date then
do;
  typel = typel + 1;
  type=type + 1;
  mess{type} = 'Rejection: S276 Booking Date Before Registration Date';
end;

/*****
/* S277: Invalid Booking Number                                             */
/*****
  if bookno ne . and bookno not in
    ('00','01','02','03','04','05','06','07','08','09','10',
     '11','12','13','14','15','16','17','18','19','20') then do;
    typel = typel + 1;
    type=type + 1;
    mess{type} = 'Rejection: S277 Invalid Booking Number';
  end;

```

```

/*****/
/* S278: Incomplete Booking Details */
/*****/
    if sch_date ne '' then do;
        if (bookdate eq '') or (bookno eq '') then do;
            type1 = type1 + 1;
            type=type + 1;
            mess{type} = 'Rejection: S278 Incomplete Booking Details';
        end;
    end;

/*****/
/* S279: Invalid Reason For Rebooking */
/*****/
    if bookno > 1 and bookno ne '' and
        rbreason not in ('A','X','C','D','H','P','F') then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S279 Invalid Reason For Rebooking';
    end;

    if bookno = 1 and sch_date eq '' and
        rbreason not in ('A','X','C','D','H','P','F') then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S279 Invalid Reason For Rebooking';
    end;

/*****/
/* S280: See Previous Month Comparison Edits */
/*****/

/*****/
/* S281: Booking Date Present/Booking Number Zero */
/*      DELETED 1 JULY 2001 */
/*****/

/*****/
/* S282: Scheduled Admission Date Present/Booking Number Zero */
/*      DELETED 1 JULY 2001 */
/*****/

/*****/
/* S283: Booking/Scheduled Admission Date and Booking Number Mismatch */
/*****/
    if ((sch_date ne . or bookdate ne .) and bookno in ('00','0')) or
        (bookno gt 0 and bookdate eq .) then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S283 Booking/Scheduled Admission Date and Booking
Number mismatch';
    end;

```

```

/*****/
/* S285: Invalid Scheduled Admission Date */
/*****/
schcc = substr(put(sch_date, ddmmyy10.),7,2);
if (schcc not in ('19','20') and sch_date ne .) or (schdt_er eq 'Y') then do;
  typel = typel + 1;
  type=type + 1;
  mess{type} = 'Rejection: S285 Invalid Scheduled Admission Date';
end;

/*****/
/* S286: Scheduled Adm Date Before Booking Date */
/*****/
if sch_date < bookdate and sch_date ne . then do;
  typel = typel + 1;
  type=type + 1;
  mess{type} = 'Rejection: S286 Scheduled Adm Date Before Booking Date';
end;

/*****/
/* S287: Scheduled Admission Date Exceeded */
/*****/
if rmreason not in ('E','X','W') and sch_date ne .
and sch_date <= rcddate and rem_date = . then do;
  typel = typel + 1;
  type=type + 1;
  mess{type} = 'Rejection: S287 Scheduled Admission Date Exceeded';
end;

/*****/
/* S288: Booking Date But Booking Number Blank/Zero */
/*****/
if bookdate ne '' and bookno in (.,0) then do;
  typel= typel + 1;
  type = type + 1;
  mess(type) = 'Rejection: S288 Booking date but booking number blank/zero';
end;

/*****/
/* S289: Booking Number But Booking Date Blank */
/*****/
if bookno not in (.,0) and bookdate eq '' then do;
  typel = typel + 1;
  type = type + 1;
  mess(type) = 'Rejection: S289 Booking number but booking date blank';
end;

/*****/
/* S290: Invalid Removal Date */
/*****/
remcc = substr(put(rem_date, ddmmyy10.),7,2);
if (remcc not in ('19','20') and rem_date ne .) or (rmdt_er eq 'Y') then do;
  typel = typel + 1;
  type=type + 1;
  mess{type} = 'Rejection: S290 Invalid Removal Date';
end;

```

```

/*****
/* S291: Removal Date Before Registration Date */
/*****
    if rem_date ne . and rem_date < reg_date then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S291 Removal Date Prior To Reg Date';
    end;

/*****
/* S292: Removed But No Reason */
/*****
    if rem_date ne . and rmreason = '' then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S292 Removed But No Reason';
    end;

/*****
/* S293: Removed, No Removal Date */
/*****
    if rmreason in ('X','W','M','B','I','U','R','Z','Q','F','O','N','T')
        and rem_date = . then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S293 Removed, No Removal Date';
    end;

/*****
/* S294: See Previous Month Comparison Edits */
/*****

/*****
/* S295: Removal Date Not Equal To Scheduled Admission Date */
/*****
    if rmreason in ('X','W') and rem_date ne .
        and rem_date ne sch_date then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S295 Removal Date Not = Scheduled Admission Date';
    end;

/*****
/* S296: Patient Admitted, Not Ready For Care */
/*****
    if rmreason in ('X','W') and rem_date ne . and readstat = 'N' then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S296 Patient Admitted, Not Ready For Care';
    end;

```

```

/*****
/* S298: Invalid Reason For Removal */
/*****
S298 = 'N';

if rmreason not in ('X','W','M','B','I','U','R','Z','Q','F','O',
                  'N','T','') then do;
    S298 = 'Y';
end;

if (rmreason not in ('X','W') and S298 eq 'N'
    and rem_date > '01Jul2001'd) then do;
    if rem_date ne . and rmreason ne '' and
        rmreason not in
            ('X','W','M','B','I','U','R','Z','Q','F','O','N','T') then do;
        S298 = 'Y';
    end;
end;

if S298 = 'Y' then do;
    typel = typel + 1;
    type=type + 1;
    mess{type} = 'Rejection: S298 Invalid Reason For Removal';
end;

/*****
/* S302: Invalid Insurance Declaration */
/*****
if insure ne '' and insure not in
    ('M','P','V','W','T','A','S','C','O','X') then do;
    typel = typel + 1;
    type=type + 1;
    mess{type} = 'Rejection: S302 Invalid Insurance Declaration';
end;

/*****
/* S303: Blank Insurance Declaration */
/*****
if rmreason eq 'W' and insure = '' then do;
    typel = typel + 1;
    type=type + 1;
    mess{type} = 'Rejection: S303 Blank Insurance Declaration';
end;

```

```

/*****/
/* S306: Invalid Transfer Destination */
/*****/
    if (rmreason = 'T' and transdes not in
('1010', '1590', '1031', '1032', '2010', '2050', '1020',
    '1050', '2111', '2220', '1121', '1210', '1170',
    '2440', '1280', '1191', '1331', '1240', '1230',
    '1360', '1450', '1390', '1150', '1580', '1180',
    '1170', '1220', '2111', '2112', '1460') or
    (rmreason = 'N' and transdes not in
('1040', '1060', '1071', '1072', '1090', '1101',
'1102', '1122', '1123', '1130', '1140', '1160', '1192', '1250', '1260', '1270', '1290',
'1320', '1331', '1332', '1340', '1351', '1352', '1370', '1440', '1480', '1491', '1492',
'1500', '1550', '1560', '1571', '1572', '1660', '1670', '1700', '1770', '1780', '1820',
'1881', '1882', '1883', '1900', '2020', '2030', '2040', '2060', '2070', '2080', '2090',
'2100', '2120', '2130', '2140', '2150', '2160', '2170', '2180', '2190', '2210', '2230',
'2240', '2251', '2252', '2260', '2320', '2350', '2360', '2371', '2372', '2373', '2374',
'2390', '2550', '2580', '2590', '2660', '2670', '2970', '3010', '3020', '3030', '3040',
'3050', '3060', '3080', '3110', '3120', '3130', '3140', '3180', '3190', '3220', '3240',
'3330', '3350', '3370', '3460', '3590', '4010', '4020', '4050', '4080', '4120', '4130',
    '4161', '4162', '4240', '4460', '5010', '5050', '5110', '5120'))))
        and dset ne 'old' then do;
            type1 = type1 + 1;
            type = type + 1;
            mess{type} = 'Rejection: S306 Invalid Transfer Destination';
        end;

/*****/
/* S307: Transfer Destination Equals Campus/Health Service */
/*****/
    if transdes = campus then do;
        type1 = type1 + 1;
        type = type + 1;
        mess(type) = 'Rejection: S307 Transfer destination = campus/health
service';
    end;

/*****/
/* S308: Transfer Destination Equals Campus Within This Health Service */
/*****/
    if campus in ('1170', '1220', '2111', '2112') and transdes in
('1170', '1220', '2111', '2112') then do;
        type1 = type1 + 1;
        type = type + 1;
        mess(type) = 'Rejection: S308 Transfer destination = campus within this
health service';
    end;

```

```

/*****
/* S309: Blank Transfer Destination */
/*   MODIFIED 1 JULY 2001 */
/*   * reg_date comparison date from 2000 to 1999 */
/*   * include 'N' in rmreason code */
/*****
    if reg_date = > '01Jul1999'd and rmreason in ('T','N','X')
        and transdes = '' then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S309 Blank Transfer Destination';
    end;

/*****
/* S310: Transfer Destination Not Transferred */
/*   MODIFIED 1 JULY 2001 */
/*   * include 'N' in rmreason code */
/*****
    if reg_date = > '01Jul2001'd and transdes ne ''
        and rmreason not in ('T','N','X') then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S310 Transfer Destination Not Transferred';
    end;

/*****
/* S311: See Previous Month Comparison Edits */
/* S315: See Previous Month Comparison Edits */
/*****

/*****
/* S318: Total NRFC Days Foll Last Clinical Urgency Increase – Not Numeric Or Blank */
/*   ADDED 1 JULY 2001 */
/*****
    if not verify(nrtot_ui, '0123456789.') then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S318 Total NRFC Days /Date of Last Clinical
Urgency Increase Not Numeric/Blank';
    end;

/*****
/* S319: Total NRFC Less Than Total Since Last Urgency Increase */
/*****
    if nrtot_ui > nr_total then do;
        type1 = type1 + 1;
        type = type + 1;
        mess{type} = 'Rejection: S319 Total NRFC less than total since last urgency
increase';
    end;

```

```

/*****/
/* S320: Total NRFC Days Not Numeric / Blank */
/*****/
    if not verify(nr_total, '0123456789.') then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S320 Total NRFC Days Not Numeric Or Blank';
    end;

/*****/
/* S321: Total NRFC Days Following Last Urgency Reassignment Not Numeric/Blank */
/*****/
    if not verify(nr_assig, '0123456789.') then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S321 Tot NRFC Days Following Last Urg Reass
Not Numeric Or Blank';
    end;

/*****/
/* S322: Total NRFC Less Than Total Since Urgency Reassignment */
/*****/
    if nr_assig > nr_total then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S322 Total NRFC Less Than Total Since Urg
Reassign';
    end;

/*****/
/* S323: See Previous Month Comparison Edits */
/* S324: See Previous Month Comparison Edits */
/* S325: See Previous Month Comparison Edits */
/* S326: Removed and Replaced with S336 S337 S338 1 JUNE 2001 (JB) */
/* S327: Removed and Replaced with S336 S337 S338 1 JUNE 2001 (JB) */
/*****/

/*****/
/* S328: Blank Hospital Initiated Postponement */
/*****/
    if hi_post = '' then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S328 Blank Hospital Initiated Postponement';
    end;

```

```

/*****/
/* S329: Invalid Combination Reason For Rebooking and Hospital Initiated Postponements */
/*****/
    if bookdate >= '01Jul2000'd and rbreason in ('H','D') and bookdate ne .
        and (hi_post = '' or hi_post = '0') then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S329 Invalid comb Reason for Rebooking & Hosp
Initiated Post';
        end;

/*****/
/* S330: See Previous Month Comparison Edits */
/*****/

/*****/
/* S331: Booking Number Less Than Hospital Initiated Postponements */
/*****/
    if bookno ne . and bookno < hi_post then do;
        type1 = type1 + 1;
        type=type + 1;
        mess{type} = 'Rejection: S331 Booking Number < Hospital Initiated
Postponement';
        end;

/*****/
/* S332: See Previous Month Comparison Edits */
/*****/

/*****/
/* S333: Invalid Hospital Initiated Postponement - DELETED 1 JULY 2002 */
/*****/

/*****/
/* S334: Hospital Initiated Postponement Not Zero, */
/*      Rebook Reason not Hosp Initiated DELETED 1 JULY 2002 */
/*****/

format waitdays 4. tot_wait 4.;
if readstat = 'N' then w_stat = 'N';

/*REMOVED FROM WAITING LIST*/
if rmreason ne '' and rem_date ne . then do;
    if rmreason in ('W','X') AND (rem_date ne .)
        then w_stat = 'A';                               /* if they were admitted */
    else if rmreason in ('R','Z','Q','F','O','B','I','U','M')
        then w_stat = 'C';                               /* if they were
cancelled/removed */
    else if rmreason in ('E')
        then w_stat = 'E';                               /* if they were data error */
    else if rmreason in ('N','T')
        then w_stat = 'T';                               /* if they were transferred */

```

```

/* do not have urgdate or urg increase date*/
if (urg_date=. and urgup_dt=.) then do;
    waitdays = intck('day', reg_date, rem_date );
    waityrs = intck('year', reg_date, rem_date );
end;

/*urgency/increase date changed before 1 July 2001 */
if urg_date <= '30JUN2001'd and urgup_dt = . and urg_date ne . then do;
    waitdays = intck('day', urg_date, rem_date);
    waityrs = intck('year', urg_date, rem_date);
end;

/*Urgency increase after 1 July 2001*/
if urgup_dt ne . then do;
    waitdays = intck('day',urgup_dt, rem_date);
    waityrs = intck('year',urgup_dt, rem_date);
end;

/*Urgency decrease after 1 July 2001 and June data contains an
urgency date (Uses June urgency date to calculate waitdays
and waityrs)*/
if urg_date >= '01JUL2001'd and ((urgup_dt eq .) and
(junurgdt ne .)) then do;
    waitdays = intck('day',junurgdt, rem_date);
    waityrs = intck('year',junurgdt, rem_date);
end; /*Urgency decrease after 1 July 2001 and June data does not
contain an urgency data (Uses registration date to calculate
waitdays and waityrs)*/
else if urg_date >= '01JUL2001'd and ((urgup_dt eq .) and
(junurgdt = .)) then do;
    waitdays = intck('day',reg_date, rem_date);
    waityrs = intck('year',reg_date, rem_date);
end;
end; /*end removed from waiting list*/

/*STILL ON WAITING LIST*/
if (rmreason = ' ' and rem_date = .) then do;
w_stat = 'W';
if readstat = 'N' then w_stat = 'N';

/* do not have urgdate/urgency increase date */
if (urg_date=. and urgup_dt=.) then do;
    waitdays = intck('day', reg_date, cdate);
    waityrs = intck('year', reg_date, cdate);
end;

/*Urgency/increase change before 1 July 2001*/
if urg_date <= '30JUN2001'd and urgup_dt = . and urg_date ne . then do;
    waitdays = intck('day', urg_date, cdate);
    waityrs = intck('year',urg_date, cdate);
end;

/*Urgency increase after 1 July 2001*/
if urgup_dt ne . then do;
    waitdays = intck('day',urgup_dt, cdate);
    waityrs = intck('year',urgup_dt, cdate);
end;

/*Urgency decrease after 1 July 2001*/
if urg_date >= '01JUL2001'd and ((urgup_dt = .) and
(junurgdt ne .)) then do;
    waitdays = intck('day', junurgdt, cdate);
    waityrs = intck('year', junurgdt, cdate);
end;
else if urg_date >= '01JUL2001'd and ((urgup_dt = .) and
(junurgdt = .)) then do;
    waitdays = intck('day',reg_date, cdate);
    waityrs = intck('year',reg_date, cdate);
end;

```

```

end; /*end still on waiting list*/

/** Calculate Total Waiting Time for those who do not have urgdate/urg increase
date ***/

if urg_date = . and urgup_dt = . and reg_date ne . then do;
  if w_stat in ('N','W') then tot_wait = (waitdays - nr_total) + 1;
  if w_stat in ('C','E','T','A') then tot_wait = (waitdays - nr_total);
end;

/** Calculate Total Waiting Time for those who have urgdate before 1 July 2001
***/

if urg_date <= '30JUN2001'd and urgup_dt = . and reg_date ne . and urg_date
ne . then do;
  if &firstsub. eq 'N' then do;
    if w_stat in ('N','W') then tot_wait = (waitdays - nrassold) + 1;
    if w_stat in ('C','E','T','A') Then tot_wait = (waitdays - nrassold);
  end;
  else if &firstsub. eq 'Y' then do;
    if w_stat in ('N','W') then tot_wait = (waitdays - nr_assig) + 1;
    if w_stat in ('C','E','T','A') Then tot_wait = (waitdays - nr_assig);
  end;
end;

/** Calculate Total Waiting Time for those who have urgency increase after 1
July 2001 ***/

if urgup_dt ne . and reg_date ne . then do;
  if w_stat in ('N','W') then tot_wait = (waitdays - nrtot_ui) + 1;
  if w_stat in ('C','E','T','A') then tot_wait = (waitdays - nrtot_ui);
end;

/** Calculate Total Waiting Time for those who have urgency decrease after 1
July 2001 ***/

if urg_date >= '01JUL2001'd and ((urgup_dt = .))
and reg_date ne . and reg_date < '01JUL2001'd then do;
  if w_stat in ('N','W') then tot_wait = (waitdays - nr_assig) + 1;
  if w_stat in ('C','E','T','A') then tot_wait = (waitdays - nr_assig);
end;

/** Calculate Total Waiting Time for those who have urgency decrease after 1
July 2001
and registration date is after 01 July 2001 ***/

if nr_total eq . then nr_total = 0;

if urg_date >= '01JUL2001'd and ((urgup_dt = .) and urg_date ne .)
and reg_date >= '01JUL2001'd then do;
  if w_stat in ('N','W') then tot_wait = (waitdays - nr_total) + 1;
  if w_stat in ('C','E','T','A') then tot_wait = (waitdays - nr_total);
end;

/***** END TOTAL WAITING TIME CALCULATIONS *****/

OVERADMT = 'N'; OVERWAIT = 'N'; WAITSTAT = 'N';

/* Patients remaining on the WAITING LIST have waited more than the desirable
time */

if w_stat in ('W') then do;
  if urgency = 1 then do;
    if tot_wait > 30 then OVERWAIT = 'Y';
  end;
  else if urgency = 2 then do;
    if tot_wait > 90 then OVERWAIT = 'Y';
  end;
end;

```

```

/* Patients who have waited more than the desirable time PRIOR TO ADMISSION */

if w_stat = 'A' and rem_date ne . and rmreason in ('X','W') then do;
  if urgency = 1 then do;
    if tot_wait > 30 then OVERADMT = 'Y';
  end;
  else if urgency = 2 then do;
    if tot_wait > 90 then OVERADMT = 'Y';
  end;
end;

/***** Assign waitstat *****/

if w_stat in ('N') AND
  rmreason in ('X','W','M','B','I','U','R','Z','Q','F','O','N','T')
then waitstat = 'N';
else if w_stat in ('W','N') then waitstat = 'Y';
else if w_stat in ('C','A') then waitstat = 'N';
else waitstat = 'E';

/***** Extra edits *****/

/*****
/* S311: Wait Equals Five Years Or More */
/*****

if waityrs >= 5 then do;
  type2 = type2 + 1;
  type=type + 1;
  mess{type} = 'Warning: S311 Wait Equals Five Years or more';
end;

/*****
/* S315: Clinical Urgency Category 1, Wait More Than 30 Days */
/*****

if urgency = '1' and (overadmt = 'Y' or overwait = 'Y') then do;
  type2 = type2 + 1;
  type=type + 1;
  mess{type} = 'Warning: S315 Clinical Urg Cat 1, Waiting > 30 Days';
end;

/*****
/* S316: Clinical Urgency Category 2, Waiting > 90 Days - DELETED 1 JULY 2002 */
/*****

/*****
/* S323: Total NRFC Days Not Equal To Zero */
/*****

if reg_date = readdate and readstat = 'R'
  and nr_total ne 0 then do;
  type1 = type1 + 1;
  type = type + 1;
  mess{type} = 'Rejection: S323 Total NRFC Days not Equal to Zero';
end;

```

```

/*****
/* S324: Negative Waiting Time (No Urgency Reassignment) */
/*****
    if urg_date = . and tot_wait < 0 then do;
        type1 = type1 + 1;
        type = type + 1;
        mess{type} = 'Rejection: S324 Negative Waiting Time (No Urgency
Reassignment)';
    end;

/*****
/* S325: Negative Waiting Time (Urgency Reassignment) */
/*****
    if urg_date ne . and tot_wait < 0 then do;
        type1 = type1 + 1;
        type = type + 1;
        mess{type} = 'Rejection: S325 Negative Waiting Time (Urgency
Reassignment)';
    end;

/*****
/* S342: Tot NRFC Days Foll Last Urg Reassignment Without Clinical Urg Reassignment Details */
/*****
    if nr_assig > 0 and (old_urg = '' or urg_date = .) then do;
        type1 = type1 + 1;
        type = type + 1;
        mess(type) = 'Rejection: S342 NRFC days fol. Urg. reassign, no urg reassign
details';
    end;

/*****
/* S343: Tot NRFC Days Foll Last Urg Increase Without Clinical Urg Increase Details */
/*****
    if nrtot_ui > 0 and (old_urg = '' or urgup_dt = .) then do;
        type1 = type1 + 1;
        type = type + 1;
        mess(type) = 'Rejection: S343 NRFC days fol. Urg. Incr., no urg incr.
Details';
    end;

/*****
/* S345: Date Of Last Clinical Urgency Increase Before 1 July 2001 */
/* ADDED 1 JULY 2001 */
/*****
    if urgup_dt ne . and urgup_dt <= '30JUN2001'd then do;
        type1 = type1 + 1;
        type = type + 1;
        mess {type} = 'Rejection: S345 Date of Last Clinical Urgency Increase < 1
July 2001';
    end;

```

```

/*****/
/* S346: Urgency Reassignment Date Before Date Of Last Clinical Urgency Increase */
/* ADDED 1 JULY 2001 */
/*****/
    if (urg_date < urgup_dt) or (urg_date eq . and urgup_dt ne .) then do;
        type1 = type1 + 1;
        type = type + 1;
        mess {type} = 'Rejection: S346 Urgency reass date < date of last clinical
urgency increase';
    end;

    if type1 = . then type1 = 0;
    if type2 = . then type2 = 0;
    if type = . then type = 0;

    x = intnx('month', cdate ,0); /** FIRST DAY OF CENSUS MONTH ***/
/*****/
/* S360: Date Of Procedure, But Patient Not Removed */
/*****/
    if dt_proc ne '' and rmreason eq '' then do;
        type1 = type1 + 1;
        type = type + 1;
        mess(type) = 'Rejection: S360 Date of Procedure, but patient not removed';
    end;

/*****/
/* S361: Date Of Procedure Not Valid Date */
/*****/
    dproc = substr(put(dt_proc,ddmmyy10.),7,2);
    if (rmreason eq 'W' and dproc not eq ('20') and dt_proc ne .) or (proc_er eq
'Y') then do;
        type1 = type1 + 1;
        type = type + 1;
        mess(type) = 'Rejection: S361 Date of Procedure not valid date';
    end;

/*****/
/* S362: Date Of Procedure Not Required */
/*****/
    if dt_proc ne '' and reg_date >= '01Jul2002'd and rmreason not eq 'W' then
do;
        type1 = type1 + 1;
        type = type + 1;
        mess(type) = 'Rejection: S362 Date of Procedure not required';
    end;

/*****/
/* S363: Date Of Procedure Blank */
/*****/
    if reg_date >= '01jul2002'd and rmreason eq 'W' and dt_proc eq '' then do;
        type2 = type2 + 1;
        type = type + 1;
        mess(type) = 'Warning: S363 Date of Procedure Blank';
    end;

```

```

/*****/
/* S364: Date Of Procedure Before Removal Date */
/*****/
if reg_date >= '01JUL2002'd and dt_proc ne . and dt_proc < rem_date then do;
    type1 = type1 + 1;
    type = type + 1;
    mess(type) = 'Rejection: S364 Date of procedure < removal date';
end;

/*****/
/* S365: Date Of Procedure After Removal Date, But Planned Same Day */
/*****/
if reg_date >= '01jul2002'd and dt_proc > rem_date and planstay eq '1' then
do;
    type2 = type2 + 1;
    type = type + 1;
    mess(type) = 'Warning: S365 Date of procedure > removal date, but intended
same day';
end;

/*****/
/* S366: Date Of Procedure More Than 7 Days After Removal */
/*****/
if reg_date >= '01jul2002'd and rmreason eq 'W' and
(intck('day', rem_date, dt_proc) > 7) then do;
    type2= type2 + 1;
    type = type + 1;
    mess(type) = 'Warning: S366 Date of procedure > 7 days after removal';
end;

/*****/
/* S370: Blank/Invalid Intended/Actual Treatment Campus */
/*****/
if ((c_site = ' ') or (c_site not in
('1010','1590','1031','1032','2010','2050','1020','1050','2111','2220','1121','
1210',
'1170','2440','1280','1191','1331','1240','1230','1360','1450','1390','1150','1
580',
'1180','1170','1220','2111','2112'))) and dset ne 'old' then do;
    type1 = type1 + 1;
    type=type + 1;
    mess{type} = 'Rejection: S370 Blank/invalid intended/actual treatment
campus';
end;
end;

if &firstsub = 'N' then do;/*don't check 1st time campus submissions*/

/* END CHECKS AGAINST NEW DATA ONLY */

```

```

/***** COMPARISON EDITS *****/

/*****/

/* S136: Procedure Mismatch Between Months */
/*****/
    if (sex = sexold) and (birthdat = birthold) and (pproc ne procold) then do;
        type2 = type2 + 1;
        type = type + 1;
        mess{type} = 'Warning: S136 Procedure Mismatch Between Months';
    end;

/*****/

/* S148: Surgical Specialty Mismatch Between Months */
/*****/
    if (sex = sexold) and (birthdat = birthold) and (surgspec ne specold) then
do;
        type2 = type2 + 1;
        type = type + 1;
        mess{type} = 'Warning: S148 Surgical Specialty Mismatch Between Months';
    end;

/*****/

/* S170: Different Registration Date */
/*****/
    if (sex = sexold) and (birthdat = birthold) and (reg_date ne regold) then do;
        type1 = type1 + 1;
        type = type + 1;
        mess{type} = 'Rejection: S170 Different Registration Date';
    end;

/*****/

/* S171: New Record, Reg Date Before This Month */
/*****/
    if reg_date ne . and (sexold = ' ') and (birthold = .) then do;
        if (reg_date < x) then do;          /*** x is first day of census month
    ***/
            type2 = type2 + 1;
            type = type + 1;
            mess{type} = 'Warning: S171 New record, Reg Date Prior To This Month';
        end;
    end;

/*****/

/* S270: Readiness Status Changed But No Readiness Change Date */
/*****/
    if dset = 'new' and (birthold ne . and sexold ne ' ') then do;
        if readstat ne rstatold and (readdate eq . or (readdate = rdateold)) then do;
            type1 = type1 + 1;
            type = type + 1;
            mess(type) = 'Rejection: S270 Readiness status changed but no Readiness
                change date';
        end;
    end;
end;

```

```

/*****
/* S280: Booking Number Mismatch Between Months */
/*****
  if (wstatold = 'Y') AND (sex = sexold) AND (birthdat = birthold)
    AND (bookno < booknold) then do;
    typel = typel + 1;
    type = type + 1;
    mess{type} = 'Rejection: S280 Booking Number mismatch between months';
  end;
/*****

/* S294: Record Removed Last Month */
/*****
  s294 = 'N';
  if (wstatold = 'N') AND (sex = sexold) AND (birthdat = birthold) then do;
    typel = typel + 1;
    type = type + 1;
    mess{type} = 'Rejection: S294 Record Removed Last Month';
    s294 = 'Y';
  end;

/*****

/* S330: Hospital Initiated Postponement Decreased Between Months */
/*****
  if (wstatold = 'Y') AND (sex = sexold) AND (birthdat = birthold)
    AND (hi_post < hiposold) then do;
    typel = typel + 1;
    type = type + 1;
    mess{type} = 'Rejection: S330 Hospital Initiated Postponement Mismatch
Between Months';
  end;

/*****

/* S332: Booking Number Increased, Hospital Initiated Postponement Unchanged */
/*****
  if (wstatold = 'Y') AND (sex = sexold) AND (birthdat = birthold)
    AND rbreason in ('H', 'D') AND rbrsnold not in ('H','D') AND bookno >
booknold AND hi_post = hiposold then do;
    typel= typel + 1;
    type = type + 1;
    mess{type} = 'Rejection: S332 Booking Number Increased, Hosp Initiated Post
Unchanged';
  end;

/*****

/* S335: Hospital Initiated Postponement Increased, Rebook Reason Not HIP */
/* This is from July 2000 data only DELETED 1 JULY 2002 */
/*****

```

```

/*****/
/* S336: Total Not Ready For Care Days – Fatal Mismatch */
/*   ADDED 1 JULY 2001 */
/*****/

rejs336 = 'N';

if readstat = 'N' and ((rem_date ne .) or
  (rmreason in ('X','W','M','B','I','U','R','Z','Q','F','O','N','T')))
then do;
  /*1.1*/ if readdate < x and (nr_total < (nrtotold +
intck('day',x,rem_date)))
    then rejs336 = 'Y';
  /*1.2*/ if readdate >= x and (nr_total < (nrtotold +
intck('day',readdate,rem_date)))
    then rejs336 = 'Y';
end;

if readstat = 'N' and rem_date eq . and rmreason eq ' ' and readdate < x and
readdate ne . then do;
  /*2.1*/ if month(cdate) in (7,8,10,12,1,3,5) then do;
    if nr_total ne (nrtotold + 31) then rejs336 = 'Y';
  end;
  else if month(cdate) eq 2 then do;
    if nr_total ne (nrtotold + 28) then rejs336 = 'Y';
  end;
  else if month(cdate) in (9,11,4,6) then do;
    if nr_total ne (nrtotold + 30) then rejs336 = 'Y';
  end;
end;

if readstat = 'N' and rem_date eq . and rmreason eq ' ' and readdate >= x and
readdate ne . then do;
  /*2.2*/ if nr_total < (nrtotold + intck('day',readdate,cdate) + 1)
    then rejs336 = 'Y';
end;

if rejs336 eq 'Y' then do;
  typel = typel + 1;
  type = type + 1;
  mess {type} = 'Rejection: S336 Total NRFC days - fatal mismatch';
end;

/*****/
/* S337: Total NRFC Days Following Last Clinical Urgency Reassignment – Fatal Mismatch */
/*   ADDED 1 JULY 2001 */
/*****/

if cdate > '31JUL2001'd then do;
  rejs337 = 'N';

  if readdate ne . and urg_date ne . then do;

    /***** S337 REMOVED FROM WAITING LIST *****/

    if readstat = 'N' and ((rem_date ne .) or (rmreason in
('X','W','M','B','I',
  'U','R','Z','Q','F','O','N','T')) then do;
      /*1.1*/ if urg_date >= x and readdate >= x and urg_date < readdate
        and (nr_assig < (/*nrassold + */intck('day',readdate,rem_date)))
        then rejs337 = 'Y';
      /*1.2*/ if urg_date >= x and readdate >= x and urg_date > readdate
        and (nr_assig < (/*nrassold + */intck('day',urg_date,rem_date)))
        then rejs337 = 'Y';
      /*1.3*/ if urg_date < x and readdate < x and
        (nr_assig < (/*nrassold + */intck('day',x,rem_date)))

```

```

then rejs337 = 'Y';
/*1.4*/ if urg_date >= x and readdate < x and
  (nr_assig < (/nrassold + */intck('day',urg_date,rem_date)))
then rejs337 = 'Y';
/*1.5*/ if urg_date < x and readdate >= x and
  (nr_assig < (/nrassold + */intck('day',readdate,rem_date)))
then rejs337 = 'Y';
end;

/***** S337 REMAIN ON WAITING LIST *****/

if readstat = 'N' and (rem_date eq . and rmreason eq '') then do;
/*2.1*/ if urg_date >= x and readdate >= x and urg_date < readdate
  and (nr_assig < (/nrassold + */intck('day',readdate,cdate)+1))
then rejs337 = 'Y';
/*2.2*/ if urg_date >= x and readdate >= x and urg_date > readdate
  and (nr_assig < (/nrassold + */intck('day',urg_date,cdate)+1))
then rejs337 = 'Y';
/*2.3*/ if urg_date < x and readdate < x and
  (nr_assig < (/nrassold + */intck('day',x,cdate)+1))
then rejs337 = 'Y';
/*2.4*/ if urg_date >= x and readdate < x and
  (nr_assig < (/nrassold + */intck('day',urg_date,cdate)+1))
then rejs337 = 'Y';
/*2.5*/ if urg_date < x and readdate >= x and
  (nr_assig < (/nrassold + */intck('day',readdate,cdate)+1))
then rejs337 = 'Y';
end;
end;
if rejs337 = 'Y' then do;
  typel = typel + 1;
  type = type + 1;
  mess {type} = 'Rejection: S337 Total NRFC days following last urg. reass
- fatal mismatch';
end;
end;

/*****
/* S338: Total NRFC Days Following Last Clinical Urgency Increase – Fatal Mismatch */
/* ADDED 1 JULY 2001 */
*****/

/***** S338 REMOVED FROM WAITING LIST *****/
if cdate > '31JUL2001'd then do;
  rejs338 = 'N';
if urgup_dt ne . then do;

if readstat = 'N' and ((rem_date ne .) or
  (rmreason in ('X','W','M','B','I','U','R','Z','Q','F','O','N','T')))
then do;
/*1.1*/ if urgup_dt >= x and readdate >= x and urgup_dt < readdate and
  (nrtot_ui < (nruiold + intck('day',readdate,rem_date)))
then rejs338 = 'Y';
/*1.2*/ if urgup_dt >= x and readdate >= x and urgup_dt > readdate and
  (nrtot_ui < (nruiold + intck('day',urgup_dt,rem_date)))
then rejs338 = 'Y';
/*1.3*/ if urgup_dt < x and readdate < x and
  (nrtot_ui < (nruiold + intck('day',x,rem_date)))
then rejs338 = 'Y';
/*1.4*/ if urgup_dt >= x and readdate < x and
  (nrtot_ui < (nruiold + intck('day',urgup_dt,rem_date)))
then rejs338 = 'Y';
/*1.5*/ if urgup_dt < x and readdate >= x and
  (nrtot_ui < (nruiold + intck('day',readdate,rem_date)))
then rejs338 = 'Y';
end;

```

```

/***** S338 REMAIN ON WAITING LIST *****/

if readstat = 'N' and (rem_date eq . or rmreason eq '') then do;
/*2.1*/ if readdate >= x and urgup_dt >= x and urgup_dt < readdate and
(nrtot_ui < (nruiold + intck('day',readdate,cdate)+1))
then rejs338 = 'Y';
/*2.2*/ if readdate >= x and urgup_dt >= x and urgup_dt > readdate and
(nrtot_ui < (nruiold + intck('day',urgup_dt,cdate)+1))
then rejs338 = 'Y';
/*2.3*/ if readdate < x and urgup_dt < x and
(nrtot_ui < (nruiold + intck('day',x,cdate)+1))
then rejs338 = 'Y';
/*2.4*/ if readdate < x and urgup_dt >= x and
(nrtot_ui < (nruiold + intck('day',urgup_dt,cdate)+1))
then rejs338 = 'Y';
/*2.5*/ if readdate >= x and urgup_dt < x and
(nrtot_ui < (nruiold + intck('day',readdate,cdate)+1))
then rejs338 = 'Y';
end;

if rejs338 = 'Y' then do;
type1 = type1 + 1;
type = type + 1;
mess {type} = 'Rejection: S338 Total NRFC days following last urg. reass
- fatal mismatch';
end;
end;
end;

/*****
/* S339: Total Not Ready For Care Days Mismatch - Warning */
/* ADDED 7 MARCH 2002 */
/*****

rejs339 = 'N';

if readstat = 'R' and readdate > reg_date and nr_total = 0
then rejs339 = 'Y';

if rejs339 = 'Y' then do;
type2 = type2 + 1;
type = type + 1;
mess {type} = 'Warning: S339 Total NRFC days mismatch - Warning';
end;

```

```

/*****
/* S340: Total Not Ready For Care Days Following Last Urgency Reassignment - Warning Mismatch */
/*   ADDED 7 MARCH 2002 */
*****/

if cdate > '31JUL2001'd then do;
  rejs340 = 'N';

  if readdate ne . and urg_date ne . then do;
    if readstat = 'R' and readdate > urg_date and readdate > reg_date and
      urg_date > reg_date and nr_assig = 0 then rejs340 = 'Y';
    end;
  end;

  if rejs340 = 'Y' then do;
    type2 = type2 + 1;
    type = type + 1;
    mess {type} = 'Warning: S340 Total NRFC days following last urgency
      reassignment - Warning Mismatch';
  end;

/*****
/* S341: Total Not Ready For Care Days Following Last Clinical Urgency Increase - Warning Mismatch*
/*   ADDED 7 MARCH 2002 */
*****/

if cdate > '31JUL2001'd then do;
  rejs341 = 'N';
if urgup_dt ne . then do;
  if readstat = 'R' and readdate > urgup_dt and readdate > reg_date and
    urgup_dt > reg_date and nrtot_ui = 0 then rejs341 = 'Y';
  end;
end;

  if rejs341 = 'Y' then do;
    type2 = type2 + 1;
    type = type + 1;
    mess {type} = 'Warning: S341 Total NRFC days following last clinical
urgency increase - Warning Mismatch';
  end;

/*****
/* S350: Clinical Urgency More Urgent But Date Last Clinical Urgency Increase Blank */
*****/

if reg_date >= '01JUL2002'd and urgency > old_urg and urgup_dt eq '' then do;
  type1 = type1 + 1;
  type = type + 1;
  mess(type) = 'Rejection: S350 Clinical urg. Increased by urg inc. dt is
blank';
end;

```

```

/*****/
/* S351: Clinical Urgency Less Urgent But Urgency Reassignment Date Blank */
/*****/
    if urgency < old_urg and urg_date eq '' then do;
        type1 = type1 + 1;
        type = type + 1;
        mess(type) = 'Rejection: S351 Clinical urg. Decreased, urg reass. Date
blank';
    end;

/*****/
/* S352: Clinical Urgency More Urgent But Date Last Clinical Urgency Increase Unchanged */
/*****/
    if reg_date >= '01JUL2002'd and urgency > urgold and urgup_dt = urgupold
    then do;
        type1 = type1 + 1;
        type = type + 1;
        mess(type) = 'Rejection: S352 Clin. Urg increased, urg incr. Dt same';
    end;

/*****/
/* S353: Clinical Urgency Less Urgent But Urgency Reassignment Date Unchanged */
/*****/
    if dset = 'new' then do;
        if urgency < urgold and urg_date = oldurgdt then do;
            type1 = type1 + 1;
            type = type + 1;
            mess(type) = 'Rejection: S353 Clin. Urg decreased, urg reass. dt same';
        end;
    end;

end; /* end for &firstsub = 'N' to do comparison checks only for hospitals who
    have submitted previous months data */

if type2 >= 1 then error_flag = 'W';
if type1 >= 1 then error_flag = 'R';

if (type > 0) and rmreason ne 'E' then output esis_v.r&esisext.&campus.;
if dset eq 'new' and rmreason ne 'E' then output esis_v.z&esisext.;
if rmreason eq 'E' then output esis_v.x&esisext.&campus.;

run;

```