

Analysis Method Notice

Chronic Conditions (Basket of 8) Emergency Hospital Multiple Admissions

This notice describes an Analysis Method that has been developed for use in the production of published national outcome indicators, performance measures and/or currencies

The Analysis Method has been reviewed by the Analysis Methodologies Group and its output submitted to the Welsh Information Standards Board (WISB) for potential accreditation.

*It should be noted that, where the data flow on which the analysis is being undertaken has not been reviewed by WISB (see 'Status of WISB Data Standards Assurance' below), accreditation of the analysis method **cannot** be interpreted as an approval of the underlying flow or of the quality of the data used.*

. For further details about the group, including Terms of Reference and membership, please visit the following website:

<http://howis.wales.nhs.uk/sites3/page.cfm?orgid=742&pid=56696>

WISB Reference: IGRN Ref. 2007 / 026

Please address enquiries about this Analysis Method the NHS Wales Informatics Service Data Standards Team.

E-mail: data.standards@wales.nhs.uk / Tel: 029 2050 2539

WISB Appraisal Assessment	Refinement Required WISB suggests that modifications are made to the Analysis Method as outlined in the appraisal Outcome.
Status of Data Standards Assurance	WISB Reviewed The data used in this Analysis Method are based on data item standards that have been through the Information Standards Assurance Process.
WISB Outcome(s)	Outcome

Indicator

The number of emergency hospital (provider spell) multiple admissions (within a year) for basket of 8 chronic conditions.

Target:

Reduction in the number of multiple admissions for the basket of 8 chronic conditions.

Rationale / Context

Emergency admissions are acknowledged as being directly influenced by the availability and quality of a comprehensive package of services for Integrated Chronic Conditions Management in the community. This target will be achieved by improving chronic conditions management services in the community, thus stabilising acute capacity.

Communities are therefore, required to work together, with all key partners, to ensure that people have access to appropriate and evidence based services in their own homes or local communities and that the transition between primary, secondary, tertiary and social care are managed effectively to ensure that the patient pathway is smooth and efficient.

Achievement of this target primarily lies in the joint planning and proactive management of chronic conditions to improve support and services to the patient.

This measure forms part of the Welsh Government 2013/14 Tier 1 Measures.

Data Source

Admitted Patient Care data set (APC ds) / Patient Episode Database for Wales (PEDW)

Definitions:

Definitional Guidance:

The chronic conditions included in this indicator are those identified by the ICD-10 4th Edition codes listed under 'additional information'. The list of ICD-10 (4th Edition) codes correspond to those chronic conditions used within the Wales Audit Office report 'The Management of Chronic Conditions by NHS Wales', which was published in December 2008. The ICD-10 4th Edition codes have been subject to review by clinical classification experts within the NHS Wales Informatics Service to ensure they adhere to current national clinical classification standards.

The 'LHB Provider Spell' method is used in the calculation of the number of emergency admissions for patients with one of the conditions specified in the basket of 8 chronic conditions. Use of the 'LHB Provider Spell' (often informally referred to as 'super spells') method for this measure has the additional benefit in that it avoids any confusion arising from concurrent Finished Consultant Episode (FCEs).

The 'LHB Provider Spell' method can be accessed here:

INCLUDE URL

The following data items and terms are relevant

Data Items:

[Primary ICD Diagnostic Code](#)

[Patient Classification](#)

[Admission Method](#)

[Start Date of Provider Spell](#)

Terms:**[Hospital Classification Categories](#)**

The definitions associated with all the data items and terms above can be accessed via the NHS Wales Data Dictionary - <http://www.datadictionary.wales.nhs.uk>.

Detailed Specification

Emergency hospital readmissions are defined as the same patient being admitted to the same LHB for the same chronic condition 'category' **within a 12 month period** (i.e. 365 days or less between the discharge date (from the original admission) and the admission date (of the subsequent admission)).

Readmissions include all subsequent **emergency** admissions with an ICD-10 code for the same chronic condition 'category' rather than just those admissions where the individual ICD-10 4th Edition code is the same as the original admission. The initial (first) emergency admission is not counted as part of the indicator, only subsequent emergency admissions.

The analysis of emergency admissions includes:

- Inpatient admissions only (patient class = '1')
 - All emergency admission methods, **excluding** emergency transfers (i.e. Admission Method '81' and Patient Classification '8').
 - Admissions where the admitting 'Hospital Classification Category' for the Site Code of Treatment (Hospital) is acute ('A'), major acute ('B') & specialist acute sites ('D') only. Admissions to community hospitals are excluded to avoid any possibility of double counting in the event of patients being admitted into an acute site and subsequently transferred to a community hospital.
 - Completed provider spells only.
 - An applicable ICD-10 4th Edition primary diagnosis code associated with the minimum (admitting) episode of the LHB Provider Spell.
 - Velindre NHS Trust is excluded.
-

Calculation:**Numerator**

Number of emergency hospital (provider spell) readmissions (within a year) for the basket of 8 chronic conditions.

Denominator

n/a

Reporting Frequency

Monthly

Monthly APC ds returns must be submitted and signed off by Local Health Boards (LHBs) / Trusts on the 17th calendar day of the month. If the 17th calendar day falls on a weekend (or bank holiday), the deadline for submissions is the next available working day.

Areas for Future Development

The following points reflect considerations raised by either the Analysis Methodologies Group or WISB in terms of aspects of the Analysis Method that require further investigation or development.

For a full breakdown of the issues considered, please refer to the formal WISB Outcome for this Analysis Method, which can be accessed via the Information Standards Assurance website:

<http://howis.wales.nhs.uk/sites3/page.cfm?orgid=742&pid=52532>

1. Review of Basket of 8 Chronic Conditions

A further review of the basket of chronic conditions could consider four aspects:

- A further review from a clinical classifications perspective to ensure that the codes included were complete, consistent and meaningful for the condition types proposed. A number of extant co-morbidity indices, designed to describe how ill a patient was exist. These include (but are not limited to) the American cancer co-morbidity indices, the 'Charlson Index' and current English guidance on co-morbidity coding for clinical coding.
- The second was a clinical review. It is recognised that the aim of this indicator is to identify high rates of admission within a set of categories, which would point to failings in community management of the symptoms. Existing intelligence cites known issues in relation to repeat admissions occurring which relate to age and multiple co-morbidity. Therefore, in trying to link individual admission / readmission categories to the original one, many relevant readmissions may be missed as they may be admitted / readmitted with any of the categories and not be counted.

Furthermore, there are several categories not included on the list which cause frequent readmissions such as low level sepsis, urinary tract infections, substance abuse and self-harm *inter alia*. Additionally some of the current categories appear to be incomplete. For example Alzheimers is included but not other causes of dementia.

- It would be useful to explore the existence of clinically-meaningful groupings of ICD-10 codes. This could include use of groupings within the classification itself.

2. Relevant ICD-10 4th Edition Codes

As it is difficult to distinguish between the reason for admission being directly related to a chronic condition, or simply the accompanying presence of a chronic condition, consideration should be given to including both primary and secondary diagnosis (ICD-10 4th Edition) codes in the indicator.

3. Review of Data Extraction Criteria

Admitting episode is currently being used to identify chronic condition admissions, as this aligns with other WG indicators and closely aligns to the approach taken for similar indicators in England. However, further analysis of existing data should be undertaken to identify which / how many episodes should be included in any method intended to link multiple admissions for the basket of chronic conditions. The possibility of including 'Source of Admission' in the analysis method could also be explored.

4. Assessment Unit Activity

It is recommended that a review into the current inconsistencies associated with the reporting of Assessment Unit activity within the APC ds be undertaken, in order to better understand its impact on the reporting of chronic conditions multiple admissions.

Appendix A – Additional Information

WAO Chronic Conditions ICD-10 4th Edition Codes and Descriptions:

ICD-10 Code	Diagnosis	Category
J40	Bronchitis, not specified as acute or chronic	Respiratory
J41	Simple and mucopurulent chronic bronchitis	Respiratory
J42	Unspecified chronic bronchitis	Respiratory
J43	Emphysema	Respiratory
J44	Other chronic obstructive pulmonary disease	Respiratory
J45	Asthma	Respiratory
J47	Bronchiectasis	Respiratory
I05	Rheumatic mitral valve diseases	Cardiovascular
I06	Rheumatic aortic valve diseases	Cardiovascular
I07	Rheumatic tricuspid valve diseases	Cardiovascular
I08	Multiple valve diseases	Cardiovascular
I09	Other rheumatic heart diseases	Cardiovascular
I10	Essential (primary) hypertension	Cardiovascular
I11	Hypertensive heart disease	Cardiovascular
I12	Hypertensive renal disease	Cardiovascular
I13	Hypertensive heart and renal disease	Cardiovascular
I15	Secondary hypertension	Cardiovascular
I20	Angina pectoris	Cardiovascular
I25	Chronic ischaemic heart disease	Cardiovascular
I34	Nonrheumatic mitral valve disorders	Cardiovascular
I35	Nonrheumatic aortic valve disorders	Cardiovascular
I36	Nonrheumatic tricuspid valve disorders	Cardiovascular
I37	Pulmonary valve disorders	Cardiovascular
I50	Heart failure	Cardiovascular
I70	Atherosclerosis	Cardiovascular
G20	Parkinson's disease	Neurological
G35	Multiple sclerosis	Neurological
G40	Epilepsy	Neurological
M02	Reactive arthropathies	Musculoskeletal
M03	Postinfective and reactive arthropathies in diseases EC	Musculoskeletal
M05	Seropostivie rheumatoid arthritis	Musculoskeletal
M06	Other rheumatoid arthritis	Musculoskeletal
M07	Psoriatic and enteropathic arthropathies	Musculoskeletal
M10	Gout	Musculoskeletal
M11	Other crystal arthropathies	Musculoskeletal
M12	Other specific arthropathies	Musculoskeletal
M13	Other arthritis	Musculoskeletal
M14	Arthropathies in other diseases classified elsewhere	Musculoskeletal
M15	Polyarthrosis	Musculoskeletal
M16	Coxarthrosis [arthrosis of hip]	Musculoskeletal
M17	Gonarthrosis [arthrosis of knee]	Musculoskeletal
M18	Arthrosis of first carppmetacarpal joint	Musculoskeletal
M19	Other arthrosis	Musculoskeletal
M40	Kyphosis and lordosis	Musculoskeletal

ICD-10 Code	Diagnosis	Category
M41	Scoliosis	Musculoskeletal
M42	Spinal osteochondrosis	Musculoskeletal
M43	Other deforming dorsopathies	Musculoskeletal
M45	Ankylosing spondylitis	Musculoskeletal
M46	Other inflammatory spondylopathies	Musculoskeletal
M47	Spondylosis	Musculoskeletal
M48	Other spondylopathies	Musculoskeletal
M49	Spondylopathies in diseases classified elsewhere	Musculoskeletal
M50	Cervical disc disorders	Musculoskeletal
M51	Other intervertebral disc disorders	Musculoskeletal
M53	Other dorsopathies, not elsewhere classified	Musculoskeletal
M54	Dorsalgia	Musculoskeletal
M80	Osteoporosis with pathological fracture	Musculoskeletal
M81	Osteoporosis without pathological fracture	Musculoskeletal
M82	Osteoporosis in diseases classified elsewhere	Musculoskeletal
E10	Insulin-dependant diabetes mellitus	Diabetes
E11	Non-insulin-dependent diabetes mellitus	Diabetes
E12	Malnutrition-related diabetes mellitus	Diabetes
E13	Other specified diabetes mellitus	Diabetes
E14	Unspecified diabetes mellitus	Diabetes
I60	Subarachnoid haemorrhage	CVA
I61	Intracerebral haemorrhage	CVA
I63	Cerebral infarction	CVA
I64	Stroke, not specified as haemorrhage or infarction	CVA
I65	Occlusion and stenosis of precerebral arteries, not resulting in cerebral infarction	CVA
I66	Occlusion and stenosis of cerebral arteries, not resulting in cerebral infarction	CVA
I48	Atrial fibrillation and flutter	Atrial Fibrillation
F00	Dementia in Alzheimer's disease	Alzheimers
G30	Alzheimer's disease	Alzheimers

Appendix B – SQL Code (where applicable)

Important: The following code is intended for information purposes only. It will contain references to specific references (servers, data item descriptions etc.) that are applicable within the NHS Wales Informatics Service only and therefore will not be suitable for direct application to local (LHB) data.

```
Print GETDATE()
Print('Running Chronic Conditions Script')

-- Create a temp copy of last month's data

Select * into ##last_month_CC
from userobj.ipu.AU_Activity_for_Chronic_Conditions

SELECT DISTINCT

patient_id = CASE WHEN
                s.nhs_no_new like '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]'
                and nhs_no_new not in
                ('000000000', '111111111', '222222222', '333333333', '444444444',
                '555555555', '666666666', '777777777', '888888888', '999999999',
                'ooooooooo') THEN nhs_no_new
                WHEN p.nhs_no is not null THEN p.nhs_no
                WHEN s.curr_prov_unit_code <> 'Eng' THEN
s.curr_prov_unit_code+s.crn
                ELSE s.prov_unit_code+s.crn
                END,

s.spell_no AS spell,
s.fin_admission_year+s.fin_admission_month AS 'Date',
s.admission_date,
s.discharge_date,
s.duration_of_spell,
[LHB_Residence_Code] = CASE WHEN
                s.curr_dha_of_residence like '7%' THEN s.curr_dha_of_residence
                ELSE 'Eng'
                END,
s.curr_prov_unit_code AS 'LHB_Provider_Code',
[condition] = CASE
                WHEN principal_diag in ('J40', 'J41', 'J42', 'J43', 'J44', 'J45', 'J47') THEN
'Respiratory'
                WHEN principal_diag in
                ('I05', 'I06', 'I07', 'I08', 'I09', 'I10', 'I11', 'I12', 'I13', 'I15', 'I20', 'I25', 'I34', 'I35',
                'I36', 'I37', 'I50', 'I70') Then 'Cardiovascular'
                WHEN principal_diag in ('G20', 'G35', 'G40') Then 'Neurological'
                WHEN principal_diag in
                ('M02', 'M03', 'M05', 'M06', 'M07', 'M10', 'M11', 'M12', 'M13', 'M14', 'M15', 'M16', 'M17', 'M18',
                'M19', 'M40', 'M41', 'M42', 'M43', 'M45', 'M46', 'M47', 'M48', 'M49', 'M50', 'M51', 'M53', 'M54', '
                M80', 'M81', 'M82') Then 'Muscuoloskeletal'
                WHEN principal_diag in ('E10', 'E11', 'E12', 'E13', 'E14') Then 'Diabetes'
                WHEN principal_diag in ('I60', 'I61', 'I63', 'I64', 'I65', 'I66') Then 'CVA'
                WHEN principal_diag in ('I48') Then 'Atrial Fibrillation'
                WHEN principal_diag in ('F00', 'G30') Then 'Alzheimers'
                END,

0 AS 'Readmission_Flag'

INTO ##ChronicConditions

FROM
```

```
userobj.analysis.super_spells_11_12 z WITH (NOLOCK) -- Provider spells ("super spells") table - contains only completed provider spells
```

```
JOIN warehousedb.dbo.apc_spell_subset s WITH (NOLOCK) on  
z.prov_unit_code = s.prov_unit_code  
and z.spell_no = s.spell_no
```

```
JOIN warehousedb.dbo.apc_episode_subset e WITH (NOLOCK) on  
s.prov_unit_code = e.prov_unit_code  
and s.spell_no = e.spell_no
```

```
JOIN warehousedb.dbo.apc_curr_site_code c on  
c.prov_unit_code = e.prov_unit_code  
and c.episode_no = e.episode_no  
and c.spell_no = e.spell_no
```

```
LEFT OUTER JOIN warehousedb.dbo.welsh_organisation ww on  
(ww.org_code = e.curr_prov_unit_code+e.curr_prov_site_code  
and e.curr_prov_unit_code <> 'rt7'  
and e.episode_end_date < '01-Apr-2006')
```

```
LEFT OUTER JOIN warehousedb.dbo.welsh_organisation wo on  
(wo.org_code = curr_site_code_of_treatment  
and (e.curr_prov_unit_code = 'rt7'  
or e.episode_end_date >= '01-Apr-2006'))
```

```
LEFT OUTER JOIN warehousedb.dbo.patient_nhs p on  
s.prov_unit_code = p.prov_unit_code  
and s.spell_no = p.spell_no
```

WHERE

```
(prov_spell_order = '1' or prov_spell_order is null) -- Admitting spell of provider spell
```

```
and e.episode_no = (SELECT MIN(episode_no)  
FROM warehousedb.dbo.apc_episode_subset m  
WHERE e.curr_prov_unit_code = m.curr_prov_unit_code  
and e.spell_no = m.spell_no  
GROUP BY curr_prov_unit_code,spell_no) -- Admitting episode of spell
```

```
and patient_class = '1' -- Inpatient Admissions only
```

```
and s.admission_method in ('21','22','23','24','25','27','28') -- All emergency admission methods, excluding emergency transfers
```

```
and (ww.org_type in ('A','B','D') or wo.org_type in ('A','B','D')) -- Acute sites
```

```
and s.curr_prov_unit_code <> 'Eng'
```

```
and s.curr_prov_unit_code <> 'RQF' -- Velindre Trust excluded
```

```
and fin_admission_year >= '2007'
```

```
and
```

```
(principal_diag in ('J40','J41','J42','J43','J44','J45','J47') -- Respiratory  
or principal_diag in  
('I05','I06','I07','I08','I09','I10','I11','I12','I13','I15','I20','I25','I34','I35','  
'I36','I37','I50','I70') -- Cardiovascular  
or principal_diag in ('G20','G35','G40') -- Neurological  
or principal_diag in  
('M02','M03','M05','M06','M07','M10','M11','M12','M13','M14','M15','M16','M17','M18','  
'M19','M40','M41','M42','M43','M45','M46','M47','M48','M49','M50','M51','M53','M54','  
'M80','M81','M82') -- Muscuoloskeletal  
or principal_diag in ('E10','E11','E12','E13','E14') -- Diabetes  
or principal_diag in ('I60','I61','I63','I64','I65','I66') -- CVA  
or principal_diag in ('I48') -- Atrial Fibrillation  
or principal_diag in ('F00','G30') -- Alzheimer's  
)
```

```
ORDER BY patient_id, s.admission_date, s.discharge_date
```



```

-----

Exec ('Alter table ##ChronicConditions
Add Index_Admission_Date smalldatetime null
alter table ##ChronicConditions
Add [id] [int] IDENTITY(1,1) NOT NULL')

-- Set the readmission flag to 1 if the spell has:
-- same patient id
-- same trust code
-- same condition
-- as a previous spell, the two spells are within a year of each other, and the
second spell comes after the first

Update ##ChronicConditions
Set Readmission_Flag = '1'
where LHB_Provider_Code+spell in (
Select distinct b.LHB_Provider_Code+b.spell from ##ChronicConditions a
join ##ChronicConditions b on
a.patient_id = b.patient_id and          -- same patient id
a.LHB_Provider_Code = b.LHB_Provider_Code and      -- same trust code
a.condition = b.condition          -- same condition
and b.id > a.id          -- the second spell comes after the first
and datediff(day, a.discharge_date, b.admission_date) between 0 and 365) -- the
two spells are within a year of each other

-- For each month in the report, the readmission and the first admission ("index
admission") must be within the previous 12 months#
-- For example in April 2006 in the report, both the readmission and the index
admission must have been in the period May 2005-April 2006

-- Get the index admission date for each readmitted spell

Select b.LHB_Provider_code+b.spell readmission_spell, a.admission_date
index_admission_date
into #get_index_admission_date_holding -- drop table
#get_index_admission_date_holding
from ##ChronicConditions a
join ##ChronicConditions b on
a.patient_id = b.patient_id and
a.LHB_Provider_Code = b.LHB_Provider_Code and
a.condition = b.condition
and b.id > a.id
and datediff(day, a.discharge_date, b.admission_date) between 0 and 365

-- Some spells will have more than one possible index admission date if there have
been multiple readmissions
-- We want to take the maximum index admission date for each readmission spell

Select distinct readmission_spell, max(index_admission_date) index_admission_date
into #get_index_admission_date -- drop table #get_index_admission_date
from #get_index_admission_date_holding
group by readmission_spell

-- Set the index admission date in the base table

Update ##ChronicConditions
set Index_Admission_Date = g.index_admission_date
from ##ChronicConditions a
join #get_index_admission_date g on
a.LHB_Provider_code+a.spell = g.readmission_spell

-- select top 100 * from ##ChronicConditions
-- select count(*) from ##ChronicConditions

```

```

-- Rolling 12 month multiple admissions

Truncate table userobj.ipu.AU_Activity_for_Chronic_Conditions

Insert into userobj.ipu.AU_Activity_for_Chronic_Conditions

Select distinct
'Rolling 12 Month Multiple Admissions' Indicator,
b.Reporting_Month_Label,
LHB_Residence_Code,
m2.org_name as LHB_Residence_Name,
LHB_Provider_Code,
m.org_name as LHB_Provider_Name,
Condition,
sum(duration_of_spell) as 'LOS',
count (distinct LHB_Provider_Code+spell) as 'Count'

from ##ChronicConditions e

join userobj.olap_admin.SaFF10_Dates b on
e.admission_date between b.start_date and b.end_date and
e.Index_admission_date between b.start_date and b.end_date

LEFT OUTER JOIN refdata.dbo.main_org m on
m.org_code = e.LHB_Provider_Code

LEFT OUTER JOIN refdata.dbo.main_org m2 on
m2.org_code = e.LHB_Residence_Code

where readmission_flag = '1'
and Reporting_Month_Label >= '200801'
and Reporting_Month_Label < cast(datepart(year, dateadd(month, -3,
IPU.FN_LastLoadMonth())) as char(4))+right(('00'+rtrim(cast(datepart(mm,
dateadd(month, -3, IPU.FN_LastLoadMonth())) as char))), 2)

group by b.Reporting_Month_Label,
LHB_Residence_Code,
m2.org_name,
LHB_Provider_Code,
m.org_name,
Condition

order by b.Reporting_Month_Label,
LHB_Residence_Code, LHB_Provider_Code, Condition

-- Monthly multiple admissions

Insert into userobj.ipu.AU_Activity_for_Chronic_Conditions

select distinct 'Monthly Multiple Admissions', [Date], LHB_Residence_Code,
m2.org_name as LHB_Residence_Name,
LHB_Provider_Code,
m.org_name as LHB_Provider_Name,
Condition,
sum(duration_of_spell), count (distinct LHB_Provider_Code+spell)

from ##ChronicConditions e

LEFT OUTER JOIN refdata.dbo.main_org m on
m.org_code = e.LHB_Provider_Code

LEFT OUTER JOIN refdata.dbo.main_org m2 on
m2.org_code = e.LHB_Residence_Code

```

```

where readmission_flag = '1'
and Date >= '200801'
and Date < cast(datepart(year, dateadd(month, -3, IPU.FN_LastLoadMonth())) as
char(4))+right(('00'+rtrim(cast(datepart(mm, dateadd(month, -3,
IPU.FN_LastLoadMonth())) as char))), 2)

group by [Date], LHB_Residence_Code,
m2.org_name,
LHB_Provider_Code,
m.org_name, Condition

order by [Date], LHB_Residence_Code, LHB_Provider_Code, Condition

-- Rolling 12 month admissions

Insert into userobj.ipu.AU_Activity_for_Chronic_Conditions

select distinct 'Rolling 12 Month Admissions', b.Reporting_Month_Label,
LHB_Residence_Code,
m2.org_name as LHB_Residence_Name,
LHB_Provider_Code,
m.org_name as LHB_Provider_Name, Condition,
sum(duration_of_spell),
count (distinct LHB_Provider_Code+spell)
from ##ChronicConditions e
join userobj.olap_admin.SaFF10_Dates b
on e.admission_date between b.start_date and b.end_date

LEFT OUTER JOIN refdata.dbo.main_org m on
m.org_code = e.LHB_Provider_Code

LEFT OUTER JOIN refdata.dbo.main_org m2 on
m2.org_code = e.LHB_Residence_Code

where Reporting_Month_Label < cast(datepart(year, dateadd(month, -3,
IPU.FN_LastLoadMonth())) as char(4))+right(('00'+rtrim(cast(datepart(mm,
dateadd(month, -3, IPU.FN_LastLoadMonth())) as char))), 2)
and Reporting_Month_Label >= '200801'

group by b.Reporting_Month_Label,
LHB_Residence_Code,
m2.org_name,
LHB_Provider_Code,
m.org_name, Condition

order by b.Reporting_Month_Label,
LHB_Residence_Code, LHB_Provider_Code, Condition

-- Monthly admissions

Insert into userobj.ipu.AU_Activity_for_Chronic_Conditions

select distinct 'Monthly Admissions', [Date], LHB_Residence_Code,
m2.org_name as LHB_Residence_Name,
LHB_Provider_Code,
m.org_name as LHB_Provider_Name, Condition,
sum(duration_of_spell), count (distinct LHB_Provider_Code+spell)

from ##ChronicConditions e

LEFT OUTER JOIN refdata.dbo.main_org m on
m.org_code = e.LHB_Provider_Code

LEFT OUTER JOIN refdata.dbo.main_org m2 on

```

```
m2.org_code = e.LHB_Residence_Code

where date < cast(datepart(year, dateadd(month, -3, IPU.FN_LastLoadMonth())) as
char(4))+right(('00'+rtrim(cast(datepart(mm, dateadd(month, -3,
IPU.FN_LastLoadMonth())) as char))), 2)
and Date >= '200801'

group by [Date], LHB_Residence_Code,
m2.org_name,
LHB_Provider_Code,
m.org_name, Condition
order by [Date], LHB_Residence_Code, LHB_Provider_Code, Condition

-- Create QA tables

Truncate table userobj.ipu.AU_Activity_for_Chronic_Conditions_lastmonth

Insert into userobj.ipu.AU_Activity_for_Chronic_Conditions_lastmonth
Select * from ##last_month_CC

Drop table ##last_month_CC

Print GETDATE()
Print('Chronic Conditions Script completed successfully')
```