/\*Super Utilizers Queries\*/

/\* BEFORE RUNNING FIND AND REPLACE Test\_HundredFile20160117 with current database name\*/

/\*Super Utilizers Queries\*/

/\*ED VISIT SECTION\*/

/\*find all OP/PB claims with ED visit codes\*/

drop table if exists ED\_CODES;

create table ED\_CODES as

select

claims\_combined.id,

claims\_combined.member\_id,

claims\_combined.master\_claim\_id,

claim\_line.claim\_id,

claim\_line.place\_of\_svc\_code,

claims\_combined.begin\_date,

claims\_combined.end\_date,

claims\_combined.allowed\_amt,

claims\_combined.claim\_line\_type\_code,

code.code\_value,

code.nomen,

code.principal

from code

join claims\_combined on

code.u\_c\_id=claims\_combined.id

join claim\_line on

claims\_combined.master\_claim\_id=claim\_line.master\_claim\_id

where nomen='REV' and code\_value in ('0450','0451','0452','0453','0454','0455','0456','0457','0458','0459','0981')

or nomen in ('CPT','HCPC') and code\_value in ('99281', '99282', '99283', '99284', '99285', 'G0380', 'G0381', 'G0382', 'G0383', 'G0384')

or place\_of\_svc\_code='23' and claims\_combined.claim\_line\_type\_code in ('OP','PB');

/\*create one row per ED claim\*/

drop table if exists ED\_Claims;

create table ED\_Claims as

select

member\_id,

claim\_id,

claim\_line\_type\_code

from ED\_CODES

group by member\_id, claim\_id, claim\_line\_type\_code;

/\* grab all OP claim lines for all OP claim with an ED code\*/

drop table if exists ED\_Claims\_all\_lines;

create table ED\_Claims\_all\_lines as

select

claims\_combined.id,

claim\_line.member\_id,

claim\_line.master\_claim\_id,

claim\_line.claim\_id,

claim\_line.begin\_date,

claim\_line.end\_date,

claim\_line.allowed\_amt,

claim\_line.claim\_line\_type\_code

from ED\_Claims

join claim\_line

on ED\_Claims.claim\_id=claim\_line.claim\_id

join claims\_combined

on claim\_line.master\_claim\_id=claims\_combined.master\_claim\_id

where claim\_line.claim\_line\_type\_code in ('OP','PB') ;

drop table if exists ED\_All\_Overlap;

create table ED\_All\_Overlap as

select

ED\_Claims\_all\_lines.member\_id,

ED\_Claims\_all\_lines.master\_claim\_id as ED\_master\_claim\_id,

ED\_Claims\_all\_lines.claim\_id as ed\_claim\_id,

ED\_Claims\_all\_lines.begin\_date as ED\_begin\_date,

ED\_Claims\_all\_lines.end\_date as ED\_end\_date,

claim\_line.master\_claim\_id,

claim\_line.claim\_id,

claim\_line.begin\_date,

claim\_line.end\_date,

claim\_line.allowed\_amt,

claim\_line.claim\_line\_type\_code

from ED\_Claims\_all\_lines

join claim\_line

on ED\_Claims\_all\_lines.member\_id=claim\_line.member\_id

where claim\_line.claim\_line\_type\_code in ('OP','PB') and claim\_line.begin\_date between ED\_Claims\_all\_lines.begin\_date and ED\_Claims\_all\_lines.end\_date;

drop table if exists ED\_All\_Overlap2;

create table ED\_All\_Overlap2 as

select

member\_id||'-'||ED\_begin\_date||'-'||ED\_end\_date as ED\_Visit\_ID,

member\_id,

ED\_begin\_date,

ED\_end\_date,

master\_claim\_id,

sum(allowed\_amt) as allowed\_amt

from ED\_All\_Overlap

group by member\_id,

ED\_begin\_date,

ED\_end\_date, master\_claim\_id;

drop table if exists ED\_Visits;

create table ED\_Visits as

select

ED\_Visit\_ID,

member\_id,

ED\_begin\_date,

ED\_end\_date,

sum(allowed\_amt) as Cost,

'M' as merge\_key

from ED\_All\_Overlap2

group by ED\_Visit\_ID, member\_id,

ED\_begin\_date,

ED\_end\_date;

drop table if exists SU\_dates;

create table SU\_dates as

select

'M' as merge\_key,

data\_start\_date,

data\_end\_date

from run;

drop table if exists ED\_Visits\_by\_yr;

create table ED\_Visits\_by\_yr as

select

ED\_Visit\_ID,

member\_id,

ED\_begin\_date,

ED\_end\_date,

Cost,

data\_start\_date,

data\_end\_date,

ED\_begin\_date-data\_start\_date,

case when ED\_begin\_date-data\_start\_date <=365 then 1

when ED\_begin\_date-data\_start\_date between 366 and 730 then 2

when ED\_begin\_date-data\_start\_date between 730 and 1096 then 3

when ED\_begin\_date-data\_start\_date between 1096 and 1460 then 4

when ED\_begin\_date-data\_start\_date between 1460 and 1825 then 5

when ED\_begin\_date-data\_start\_date between 1825 and 2190 then 6 end as Submission\_Year,

case when ED\_begin\_date-data\_start\_date <=365 then year(data\_start\_date)

when ED\_begin\_date-data\_start\_date between 366 and 730 then year(data\_start\_date) +1

when ED\_begin\_date-data\_start\_date between 730 and 1096 then year(data\_start\_date) +2

when ED\_begin\_date-data\_start\_date between 1096 and 1460 then year(data\_start\_date) +3

when ED\_begin\_date-data\_start\_date between 1460 and 1825 then year(data\_start\_date) +4

when ED\_begin\_date-data\_start\_date between 1825 and 2190 then year(data\_start\_date) +6 end as Submission\_Year\_Date

from ED\_Visits

join SU\_dates

on ED\_Visits.merge\_key = SU\_dates.merge\_key;

drop table if exists ED\_Visit\_Counts;

create table ED\_Visit\_Counts as

select

member\_id,

Submission\_Year,

Submission\_Year\_Date,

count(\*) as ED\_Counts,

sum(Cost) as ED\_Costs

from ED\_Visits\_by\_yr

group by member\_id, Submission\_Year, Submission\_Year\_Date;

drop table if exists ED\_CODES;

drop table if exists ED\_Claims;

drop table if exists ED\_DX\_Codes;

create table ED\_DX\_Codes as

select

ED\_Claims\_all\_lines.\*,

Test\_HundredFile20160117.code.code\_value,

Test\_HundredFile20160117.code.nomen,

Test\_HundredFile20160117.code.principal,

episode\_builder\_5\_4\_001.code.Description,

episode\_builder\_5\_4\_001.code.group\_id,

episode\_builder\_5\_4\_001.code.CCS\_Category,

episode\_builder\_5\_4\_001.code.CCS\_Type\_ID,

episode\_builder\_5\_4\_001.ccs.DESCRIPTION as CCS\_Cat\_Description

from ED\_Claims\_all\_lines

left join Test\_HundredFile20160117.code

on ED\_Claims\_all\_lines.id=Test\_HundredFile20160117.code.u\_c\_id

left join episode\_builder\_5\_4\_001.code

on Test\_HundredFile20160117.code.code\_value =episode\_builder\_5\_4\_001.code.value

left join episode\_builder\_5\_4\_001.ccs

on episode\_builder\_5\_4\_001.code.CCS\_Type\_ID=episode\_builder\_5\_4\_001.ccs.type\_id and episode\_builder\_5\_4\_001.code.CCS\_Category=episode\_builder\_5\_4\_001.ccs.category

where Test\_HundredFile20160117.code.nomen = 'DX' and episode\_builder\_5\_4\_001.code.type\_id = 'DX'

union

select

ED\_Claims\_all\_lines.\*,

Test\_HundredFile20160117.code.code\_value,

Test\_HundredFile20160117.code.nomen,

Test\_HundredFile20160117.code.principal,

episode\_builder\_5\_4\_001.code\_icd10.Description,

episode\_builder\_5\_4\_001.code\_icd10.group\_id,

episode\_builder\_5\_4\_001.code\_icd10.CCS\_Category,

episode\_builder\_5\_4\_001.code\_icd10.CCS\_Type\_ID,

episode\_builder\_5\_4\_001.ccs.DESCRIPTION as CCS\_Cat\_Description

from ED\_Claims\_all\_lines

left join Test\_HundredFile20160117.code

on ED\_Claims\_all\_lines.id=Test\_HundredFile20160117.code.u\_c\_id

left join episode\_builder\_5\_4\_001.code\_icd10

on Test\_HundredFile20160117.code.code\_value =episode\_builder\_5\_4\_001.code\_icd10.value

left join episode\_builder\_5\_4\_001.ccs

on episode\_builder\_5\_4\_001.code\_icd10.CCS\_Type\_ID=episode\_builder\_5\_4\_001.ccs.type\_id and episode\_builder\_5\_4\_001.code\_icd10.CCS\_Category=episode\_builder\_5\_4\_001.ccs.category

where Test\_HundredFile20160117.code.nomen = 'DXX' and episode\_builder\_5\_4\_001.code\_icd10.type\_id = 'DXX'

;

drop table if exists ED\_All\_Overlap;

drop table if exists ED\_All\_Overlap2;

drop table if exists ED\_Claims\_all\_lines;

drop table if exists ED\_Visits;

/\* find all IP stays\*/

drop table if exists IP\_STAYS;

drop table if exists IP\_STAYS\_by\_yr;

drop table if exists IP\_STAY\_Counts;

create table IP\_STAYS as

select

id,

member\_id,

master\_claim\_id,

begin\_date,

end\_date,

allowed\_amt,

'M' as merge\_key

from claims\_combined

where claim\_line\_type\_code = 'IP';

create table IP\_DX\_Codes as

select

IP\_STAYS.id,

IP\_STAYS.member\_id,

IP\_STAYS.master\_claim\_id,

IP\_STAYS.begin\_date,

IP\_STAYS.end\_date,

IP\_STAYS.allowed\_amt,

Test\_HundredFile20160117.code.code\_value,

Test\_HundredFile20160117.code.nomen,

Test\_HundredFile20160117.code.principal,

episode\_builder\_5\_4\_001.code.Description,

episode\_builder\_5\_4\_001.code.group\_id,

episode\_builder\_5\_4\_001.code.CCS\_Category,

episode\_builder\_5\_4\_001.code.CCS\_Type\_ID,

episode\_builder\_5\_4\_001.ccs.DESCRIPTION as CCS\_Cat\_Description

from IP\_STAYS

left join Test\_HundredFile20160117.code

on IP\_STAYS.id=Test\_HundredFile20160117.code.u\_c\_id

left join episode\_builder\_5\_4\_001.code

on Test\_HundredFile20160117.code.code\_value =episode\_builder\_5\_4\_001.code.value

left join episode\_builder\_5\_4\_001.ccs

on episode\_builder\_5\_4\_001.code.CCS\_Type\_ID=episode\_builder\_5\_4\_001.ccs.type\_id and episode\_builder\_5\_4\_001.code.CCS\_Category=episode\_builder\_5\_4\_001.ccs.category

where Test\_HundredFile20160117.code.nomen = 'DX' and episode\_builder\_5\_4\_001.code.type\_id = 'DX'

union

select

IP\_STAYS.id,

IP\_STAYS.member\_id,

IP\_STAYS.master\_claim\_id,

IP\_STAYS.begin\_date,

IP\_STAYS.end\_date,

IP\_STAYS.allowed\_amt,

Test\_HundredFile20160117.code.code\_value,

Test\_HundredFile20160117.code.nomen,

Test\_HundredFile20160117.code.principal,

episode\_builder\_5\_4\_001.code\_icd10.Description,

episode\_builder\_5\_4\_001.code\_icd10.group\_id,

episode\_builder\_5\_4\_001.code\_icd10.CCS\_Category,

episode\_builder\_5\_4\_001.code\_icd10.CCS\_Type\_ID,

episode\_builder\_5\_4\_001.ccs.DESCRIPTION as CCS\_Cat\_Description

from IP\_STAYS

left join code

on IP\_STAYS.id=Test\_HundredFile20160117.code.u\_c\_id

left join episode\_builder\_5\_4\_001.code\_icd10

on Test\_HundredFile20160117.code.code\_value =episode\_builder\_5\_4\_001.code\_icd10.value

left join episode\_builder\_5\_4\_001.ccs

on episode\_builder\_5\_4\_001.code\_icd10.CCS\_Type\_ID=episode\_builder\_5\_4\_001.ccs.type\_id and episode\_builder\_5\_4\_001.code\_icd10.CCS\_Category=episode\_builder\_5\_4\_001.ccs.category

where Test\_HundredFile20160117.code.nomen = 'DXX' and episode\_builder\_5\_4\_001.code\_icd10.type\_id = 'DXX'

;

create table IP\_STAYS\_by\_yr as

select

master\_claim\_id,

member\_id,

begin\_date,

end\_date,

allowed\_amt,

data\_start\_date,

data\_end\_date,

begin\_date-data\_start\_date,

case when begin\_date-data\_start\_date <=365 then 1

when begin\_date-data\_start\_date between 366 and 730 then 2

when begin\_date-data\_start\_date between 730 and 1096 then 3

when begin\_date-data\_start\_date between 1096 and 1460 then 4

when begin\_date-data\_start\_date between 1460 and 1825 then 5

when begin\_date-data\_start\_date between 1825 and 2190 then 6 end as Submission\_Year,

case when begin\_date-data\_start\_date <=365 then year(data\_start\_date)

when begin\_date-data\_start\_date between 366 and 730 then year(data\_start\_date) +1

when begin\_date-data\_start\_date between 730 and 1096 then year(data\_start\_date) +2

when begin\_date-data\_start\_date between 1096 and 1460 then year(data\_start\_date) +3

when begin\_date-data\_start\_date between 1460 and 1825 then year(data\_start\_date) +4

when begin\_date-data\_start\_date between 1825 and 2190 then year(data\_start\_date) +6 end as Submission\_Year\_Date

from IP\_STAYS

join SU\_dates

on IP\_STAYS.merge\_key = SU\_dates.merge\_key;

create table IP\_STAY\_Counts as

select

member\_id,

Submission\_Year,

Submission\_Year\_Date,

count(\*) as IP\_Counts,

sum(allowed\_amt) as IP\_Costs

from IP\_STAYS\_by\_yr

group by member\_id, Submission\_Year,Submission\_Year\_Date;

drop table if exists IP\_STAYS;

drop table if exists IP\_STAYS\_by\_yr;

/\*bring IP stay and ED visits information together\*/

drop table if exists years;

create table years as

select

Submission\_Year,

Submission\_Year\_Date,

'M' as merge\_key

from ED\_Visit\_Counts

group by Submission\_Year,

Submission\_Year\_Date;

drop table if exists SU\_member;

create table SU\_member as

select

member.member\_id,

gender\_code,

birth\_year,

'M' as merge\_key

from member;

drop table if exists SU1;

create table SU1 as

SELECT

member\_id,

gender\_code,

birth\_year,

Submission\_Year,

Submission\_Year\_Date

from SU\_member

join years

on SU\_member.merge\_key=years.merge\_key;

drop table if exists Super\_Utilizers;

create table Super\_Utilizers as

SELECT

SU1.\*,

ED\_Visit\_Counts.ED\_Counts,

ED\_Visit\_Counts.ED\_Costs,

IP\_STAY\_Counts.IP\_Counts,

IP\_STAY\_Counts.IP\_Costs

from SU1

left join ED\_Visit\_Counts

on SU1.member\_id=ED\_Visit\_Counts.member\_id and SU1.Submission\_Year = ED\_Visit\_Counts.Submission\_Year

left join IP\_STAY\_Counts

on SU1.member\_id=IP\_STAY\_Counts.member\_id and SU1.Submission\_Year = IP\_STAY\_Counts.Submission\_Year;

drop table if exists SU1;

drop table if exists SU\_member;

drop table if exists years;

drop table if exists SU\_dates;