Table of Contents

[Schema Overview](#_6zh4l79u0p37)

[Original schema](#_kupkf7oy4ijw)

[OpenMRS concept schema](#_rxqitvnmp9wb)

[Data warehouse schema](#_msa1uvxxrd2p)

[Views](#_ggt5ylkcr0bc)

[Using the MySql INFORMATION\_SCHEMA](#_427hj8ncae2k)

[Form structure](#_4g496cf7ug0)

[HIV Forms](#_o2fkmmxifv2w)

[Newer, non-HIV forms](#_2parvtgqmovw)

[Field name uniqueness](#_mwibzi21z916)

[Example iSanté queries](#_giraseltr0so)

[Find the three sites with the most HIV intakes during November 2013](#_cnpoufkxshzk)

[Find the ten most-diagnosed conditions](#_9fkabkn0pa36)

[Find all sites with results returned from OpenElis in the last quarter of 2013](#_4pzz34kqwhoq)

[How many patients were prescribed the same regimen more than 100 times](#_72dlmtkozdmc)

[Reverse engineering iSanté reports](#_e8mte9e190t6)

[Displaying the report query](#_tf7r79v1hjrs)

[SQL translation](#_8b1wad1uo22c)

[Running ad hoc queries](#_kfaai51aymc0)

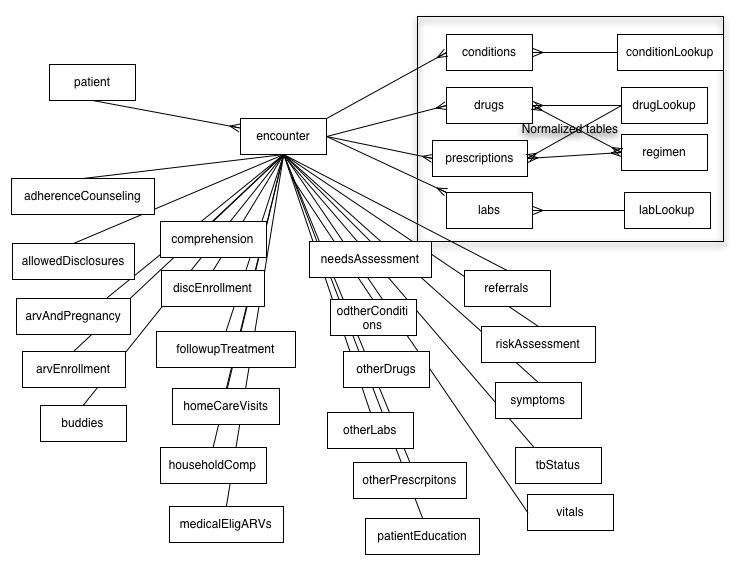
# 

# 

# Schema Overview

The iSanté database is a hybrid of the original de-normalized and normalized tables, lookup tables for the normalized tables, the OpenMRS concept schema tables, and the data warehouse tables. Each schema category is described below.

## Original schema



A patient has multiple encounters and an encounter has one record in each denormalized table and multiple records in the normalized tables. Denormalized tables comprise the majority of the tables while the normalized tables are in the upper right corner of the diagram.

The denormalized tables (meaning one row per encounter per table) usually hold precisely the data from a section of the HIV forms.

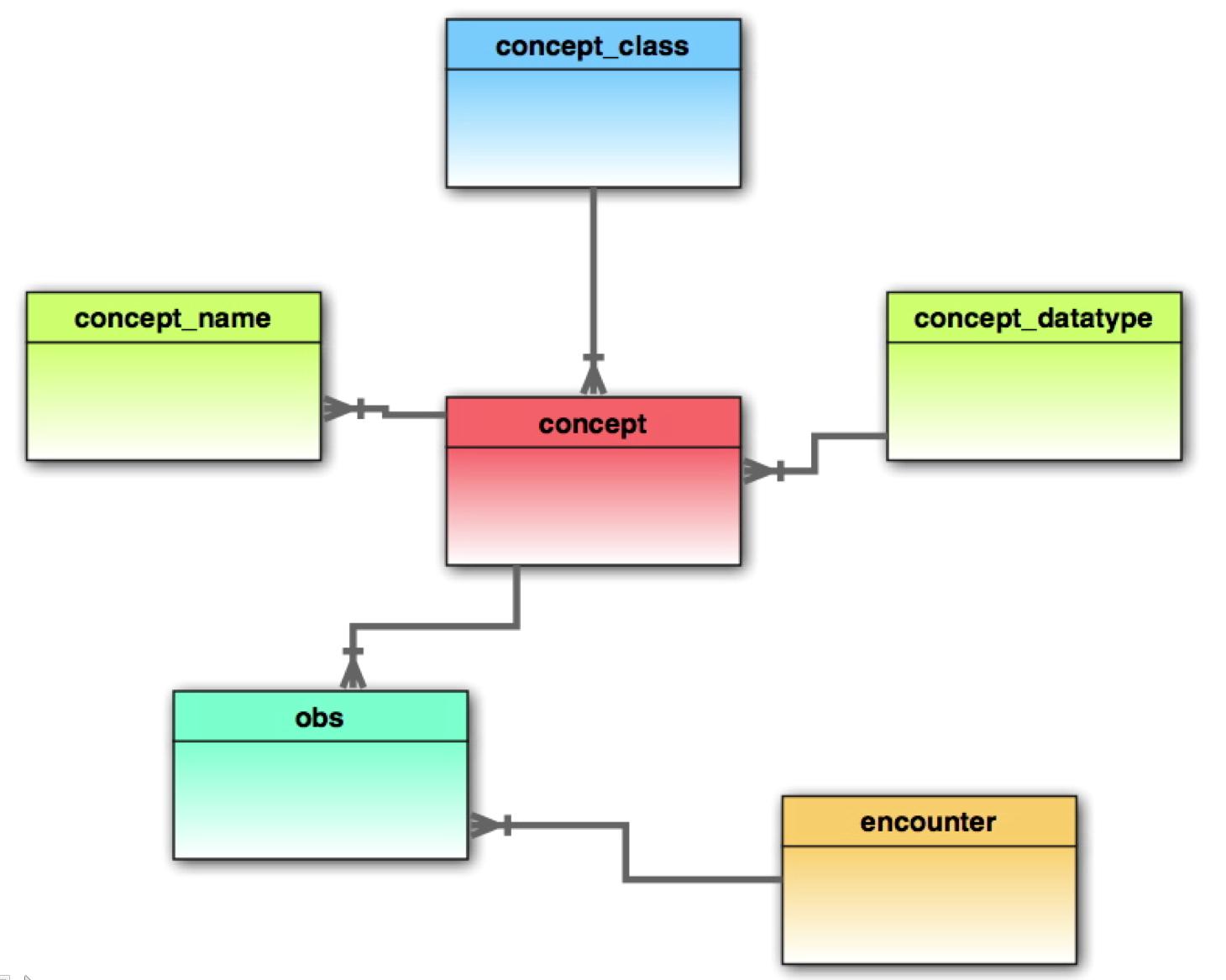
Likewise the normalized tables hold data from specific sections of the original HIV forms in multiple rows per encounter and have a foreign key to a paired lookup table (see below) for repeatable attribute information.

Lookup tables pair with normalized tables to provide foreign key attributes:

conditionLookup, discReasonLookup, drugGroupLookup, drugLookup, encTypeLookup, immunizationLookup, labelLookup, labGroupLookup, labLookup, labPanelLookup, networkLookup, patientLookup, patientStatusLookup, pedLabsLookup, queueStatusLookup, referralLookup, riskLookup

## OpenMRS concept schema

The OpenMRS concept schema was adopted for all forms added to iSanté after the original HIV forms, including the primary care and ob-gyn forms. These tables from the OpenMRS schema were added to the iSanté schema:



## Data warehouse schema

The data warehouse schema is loaded asynchronously to facilitate reporting. Example tables include:

'drugTable’, ’drugSummary’, ’cd4Table’,’pepfarTable’, 'patientStatusTemp’

These tables are not pertinent to direct analysis of visit transactions, but are used to enhance reporting functionality and performance.

The warehouse schema includes sets of tables for specific subject areas. These tables all begin with a dw\_ prefix. Each reporting subject area has a separate set of tables, substituting the subject **key** (tb, nutrition, malaria, etc) in the table names:

dw\_**key**\_snapshot

dw\_**key**ReportLookup

dw\_**key**\_patients

dw\_**key**\_slices

## Views

The view structure is designed to make selection of patients who are HIV positive or negative easier. Most denormalized tables have views for positive, negative, and all variations. Normalized table views include their respective lookup table attributes. The views are structured as follows:

v\_<table\_name> -- based upon the base view encValid -- HIV positive encounters

n\_<table\_name> -- based upon the base view encValidNeg -- HIV negative encounters

a\_<table\_name> -- based upon the base view encValidAll -- all encounters

## 

## 

## Using the MySql INFORMATION\_SCHEMA

In-depth analysis of the iSanté schema can be done with the MySql INFORMATION\_SCHEMA, in particular with the INFORMATION\_SCHEMA.tables and INFORMATION\_SCHEMA.columns views. Here are some example queries:

List all iSanté tables:

select table\_name

from INFORMATION\_SCHEMA.tables

where table\_schema = 'itech' and

table\_type = 'BASE TABLE' and

table\_name not like '%temp%' order by 1

List all views:

select table\_name

from INFORMATION\_SCHEMA.tables

where table\_schema = 'itech' and

table\_type = 'VIEW'

order by 1

List columns in the encounter table:

select distinct column\_name, column\_type, ordinal\_position

from INFORMATION\_SCHEMA.columns

where table\_schema = 'itech' and table\_name = 'encounter' order by 3

# 

# 

# Form structure

iSanté forms correspond to the various types of patient encounters. The full list of forms (encounterType or encType) is:

select encounterType as encType, enName, frName from encTypeLookup

|  |  |  |
| --- | --- | --- |
| encType | enName | frName |
| 10 | Registration | Enregistrement |
| 1 | Intake | Saisie Première |
| 2 | Followup | Suivi Visite |
| 3 | Couns. Intake | Couns. Enrôlement |
| 4 | Couns. Followup | Couns. Suivi |
| 5 | Prescription | Ord. Médicale |
| 6 | Laboratory | Analyses de Lab. |
| 7 | Home Visit | Visite à Domicile |
| 9 | Referral Tracking | Suivi de la Référence |
| 11 | Selection Committee Report | Rapp. du Comité de Sélection |
| 12 | Discontinuation | Discontinuation |
| 13 | External Lab Results | Résultats de laboratoire externe |
| 14 | Adherence Counseling | Conseils d'Adhérence |
| 15 | Pediatric Registration | Enregistrement pédiatrique |
| 16 | Pediatric Intake | Saisie Première pédiatrique |
| 17 | Pediatric Followup | Suivi Visite pédiatrique |
| 18 | Pediatric Prescription | Ord. Médicale pédiatrique |
| 19 | Pediatric Laboratory | Analyses de Lab. pédiatrique |
| 20 | Pediatric Adherence | Adhérence pédiatrique |
| 21 | Pediatric Discontinuation | Discontinuation pédiatrique |
| 24 | Ob/gyn intake | Saisie Première ob/gyn |
| 25 | Ob/gyn followup | Ob/gyn Suivi |
| 26 | Labor & delivery | Travail et d'accouchement |
| 27 | Primary care--intake form | Soins de santé primaire--premiére consultation |
| 28 | Primary care--followup form | Soins de santé primaire--consultation |
| 29 | Pediatric primary care--intake form | Soins de santé primaire--premiére con. p |
| 30 | Records request | Demande de dossier |
| 31 | Pediatric primary care--followup form | Soins de santé primaire--con. pédiatriqu |

## HIV Forms

iSanté HIV forms (in blue above) are divided into **sections** containing **fields** which correspond to **columns** in the denormalized tables. For instance, the adult HIV intake form contains a vitals section which contains fields like height, weight, and blood pressure, each of which is a column in the vitals table. If you can determine the field name for a field, you can backtrack to its table in the database with this MySql information schema query:

select table\_name from INFORMATION\_SCHEMA.COLUMNS

where column\_name = 'vitalWeight’

Determine the field name for any form field in iSanté while in the Chrome or Firefox browsers as follows:

Chrome:

* From the 'customize and control' menu, select Tools → Developer Tools
* Notice the small magnifying glass that appears at the bottom left of the tools area -- click this once and hover over the field of interest. The ID of the form field will be displayed. With the numeric or [] part removed, this ID is exactly the field name in iSante.
* Click the field to see the full code for this form field (not really necessary to obtain the field name)
* Click the magnifying glass again to exit this mode

Firefox:

* Select Tools --> Web Developer --> Developer Tools
* Click the wrench at the bottom right
* Click the Inspector button at the top of the tool window
* Hover over the field as above

## Newer, non-HIV forms

Newer forms use the OpenMRS concept schema for storing data. The technique for finding the field name is the same. But now, instead of column\_name, the concept.short\_name is used to find the concept\_id in which the field value is stored:

select concept\_id from concept where short\_name = 'supplementVitA’

## Field name uniqueness

All original schema column\_names and concept short\_names are unique within the entire schema (across all tables, not including key fields). The following queries verify this (they return zero rows in each case) for column names and concept short\_names, respectively:

select column\_name, count(\*) from INFORMATION\_SCHEMA.columns c, INFORMATION\_SCHEMA.tables t where

c.table\_schema = 'itech' and c.column\_name not in ('visitdatedd', 'visitdatemm', 'visitdateyy', 'seqnum', 'dbsite', 'sitecode', 'labid','drugid','patientid') and

c.table\_name in ('adherenceCounseling', 'allergies', 'allowedDisclosures', 'arvAndPregnancy', 'arvEnrollment', 'buddies', 'comprehension', 'conditions', 'discEnrollment', 'drugs', 'followupTreatment', 'homeCareVisits', 'householdComp', 'immunizations', 'labs', 'medicalEligARVs', 'needsAssessment', 'patientEducation', 'pedHistory', 'pedLabs', 'prescriptions', 'referrals', 'tbStatus', 'vitals', 'riskAssessment', 'riskAssessments', 'prescriptionOtherFields') and

c.table\_name = t.table\_name and c.table\_schema = t.table\_schema and t.table\_type = 'BASE TABLE'

group by 1 having count(\*) > 1;

select short\_name, count(\*) from concept group by 1 having count(\*) > 1

# 

# 

# Example iSanté queries

The encounter table (also encValid, encValidAll, and encValidNeg views) is the heart of the schema – most queries include it. For instance:

## Find the three sites with the most HIV intakes during November 2013

select c.clinic, count(\*) as hivIntakes

from encValid e, clinicLookup c

where c.sitecode = e.sitecode and

e.encountertype in (1,16) and

visitdate between '2013-11-01' and '2013-11-31'

group by 1

order by 2 desc

limit 3

Notes:

* This query joins encValid and clinicLookup on sitecode to get clinic name
* The encValid view automatically restricts to HIV patients only
* The encounterType restriction limits to adult and pediatric HIV intake forms
* In MySql, ***limit*** restricts the number of rows returned

Many queries involve conditions, lab tests, and prescriptions. Below are examples of each:

## Find the ten most-diagnosed conditions

select conditionNameEn, conditionNameFr, count(distinct patientid)

from a\_conditions

group by 1,2

order by 3 desc limit 10

Notes:

* This query uses the a\_conditions view, which is based on the conditions and conditionLookup tables.
* Count is **distinct patientid** to avoid multiple counts of the same Dx for the same patient.

## Find all sites with results returned from OpenElis in the last quarter of 2013

select clinic, count(distinct patientid+testnamefr)

from a\_labs l, clinicLookup c

where l.sitecode = c.sitecode and

sendingsiteid is not null and

year(visitdate) = 2013 and quarter(visitdate) = 4

group by 1 order by 2

Notes:

* This query uses the a\_labs view, based upon the labs and labLookup tables.
* Results that originate from OpenElis always have a non-null sendingsiteid.
* Notice the use of the ***month*** and ***quarter*** date/time functions. These are used extensively in iSanté reporting.

## How many patients were prescribed the same regimen more than 100 times

select count(distinct a.patientid)

from (

select patientid, regimen, count(\*) from pepfarTable group by 1,2 having count(\*) > 100) a

Notes:

* This query uses the warehouse table **pepfarTable**, which records the computed regimen for each patient dispense-date, based on the component ART drugs dispensed.
* The query has an inner-query as its ***from*** clause. The results of the inner-query are used in the outer target list, which counts distinct patientids.

# 

# 

# Reverse engineering iSanté reports

## Displaying the report query

With superuser privilege and the debugFlag set, a user can review the underlying query or queries used to produce any report. For instance, the header area for the primary care report ***Tous les patients diagnostiqués au cours des 7 derniers jours*** looks like this when the debugFlag is set:



Copying and analyzing the underlying queries from reports can provide insight into the schema and help with structuring other queries. Here are some notes on the query in the header above:

select case when p.clinicPatientID is not null then p.clinicPatientID else concat(lname,fname) end as clinicPatientID, case when datediff(dd, ymdToDate(dobyy,dobmm,dobdd),getDate()) between 0 and 30 then '1. 0-1 mons' when datediff(dd, ymdToDate(dobyy,dobmm,dobdd), getDate()) between 31 and 365 then '2. 1 mons-1 ans' when datediff(dd, ymdToDate(dobyy,dobmm,dobdd), getDate()) between 366 and 1825 then '3. 1-5 ans' when datediff(dd, ymdToDate(dobyy,dobmm,dobdd), getDate()) between 1826 and 3650 then '4. 6 ans-10 ans' else '5. 10 ans+' end as 'Le groupe d’âge', lname as lName, fname as fName, p.nationalID, CASE WHEN sex = 1 THEN 'F' WHEN sex = 2 THEN 'H' ELSE 'I' END as sex, date(ymdToDate(dobyy,dobmm,dobdd)) as Dob, CONCAT\_WS('|', p.addrDistrict, p.addrSection, p.addrTown) AS Address, p.telephone AS Phone, p.contact AS Contact, p.patientid as patientID, GROUP\_CONCAT(CONCAT(UCASE(SUBSTRING(n.name, 1, 1)), SUBSTRING(n.name, 2))) as 'Dx' from patient p, encValidAll e, obs o, concept c, concept\_name n where p.patientid = e.patientid and e.encounter\_id = o.encounter\_id and datediff(dd, e.visitDate, getDate()) between 0 and 7 and e.sitecode = o.location\_id and o.location\_id = 73103 and o.concept\_id = c.concept\_id and c.class\_id = 4 and o.value\_boolean is true and c.concept\_id = n.concept\_id and n.locale = 'fr' group by 11 order by 2,3,4,5,6,7,8,9,10,11

Notes:

* This query joins the patient, obs, and concept tables with the encValidAll view to produce all patients diagnosed during the past 7 days, grouped by age group.
* The patient table stores birth date in three separate fields; the ymdToDate function combines these into a single MySql date value; the computed birthdate field is used to group the patients.
* The CONCAT\_WS (concat with specified separator) and GROUP\_CONCAT (concat values across grouping variables) are used to accumulate values and generate a single row for each patient.

## SQL translation

Most report queries in iSanté are written with Microsoft SQL rather than MySql SQL (iSanté originally used Microsoft SQL Server). While the two variations of SQL are mostly identical, their built-in functions and arguments differ, and there are some other minor semantical differences. iSanté has a built-in translator for bridging the two. The query above is Microsoft SQL – iSanté automatically translates this to MySql SQL before submitting it for processing. It is also possible to translate it manually in iSanté by pasting it into the window displayed when running this URL:

<https://hostname/isante/sqlProcess.php>

resulting in:

select case when p . clinicPatientID is not null then p . clinicPatientID else concat(lname , fname) end as clinicPatientID , case when datediff(now() , ymdToDate(dobyy , dobmm , dobdd)) between 0 and 30 then '1. 0-1 mons' when datediff(now() , ymdToDate(dobyy , dobmm , dobdd)) between 31 and 365 then '2. 1 mons-1 ans' when datediff(now() , ymdToDate(dobyy , dobmm , dobdd)) between 366 and 1825 then '3. 1-5 ans' when datediff(now() , ymdToDate(dobyy , dobmm , dobdd)) between 1826 and 3650 then '4. 6 ans-10 ans' else '5. 10 ans+' end as 'Le groupe d’âge' , lname as lName , fname as fName , p . nationalID , CASE WHEN sex = 1 THEN 'F' WHEN sex = 2 THEN 'H' ELSE 'I' END as sex , date(ymdToDate(dobyy , dobmm , dobdd)) as Dob , CONCAT\_WS('|' , p . addrDistrict , p . addrSection , p . addrTown) AS Address , p . telephone AS Phone , p . contact AS Contact , p . patientid as patientID , GROUP\_CONCAT(CONCAT(UCASE(SUBSTRING(n . name , 1 , 1)) , SUBSTRING(n . name , 2))) as 'Dx' from patient p , encValidAll e , obs o , concept c , concept\_name n where p . patientid = e . patientid and e . encounter\_id = o . encounter\_id and datediff(now() , e . visitDate) between 0 and 7 and e . sitecode = o . location\_id and o . location\_id = 73103 and o . concept\_id = c . concept\_id and c . class\_id = 4 and o . value\_boolean is true and c . concept\_id = n . concept\_id and n . locale = 'fr' group by 11 order by 2 , 3 , 4 , 5 , 6 , 7 , 8 , 9 , 10 , 11

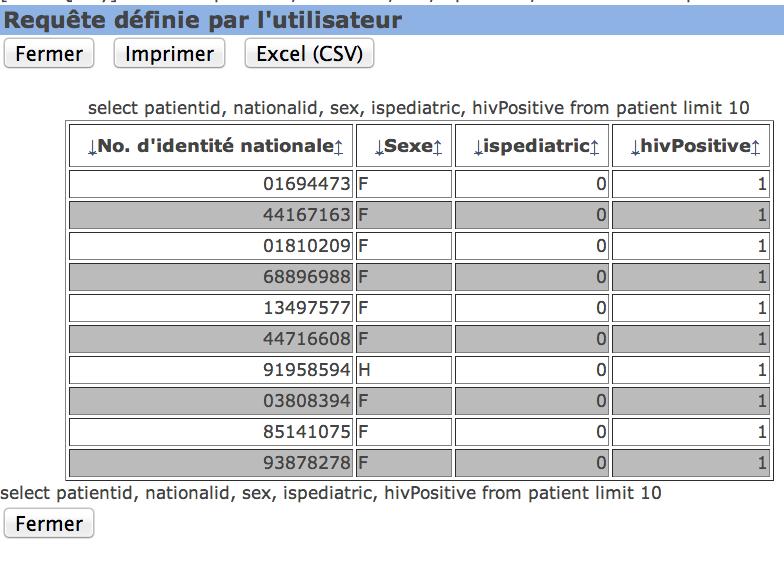
In this case, the differences are primarily in the DATEDIFF function calls. *It is always safest to run a report query through the manual translator before using it or modifying it for use in MySql.*

## Running ad hoc queries

iSanté has a built-in ad hoc query window available on the Administration menu for super administrators only. Any MySql SELECT statements can be executed in this window and the results are delivered in an output window that has buttons for printing and for csv formatted export. It is not possible to execute INSERT, UPDATE, DELETE, or CREATE statements in the window.

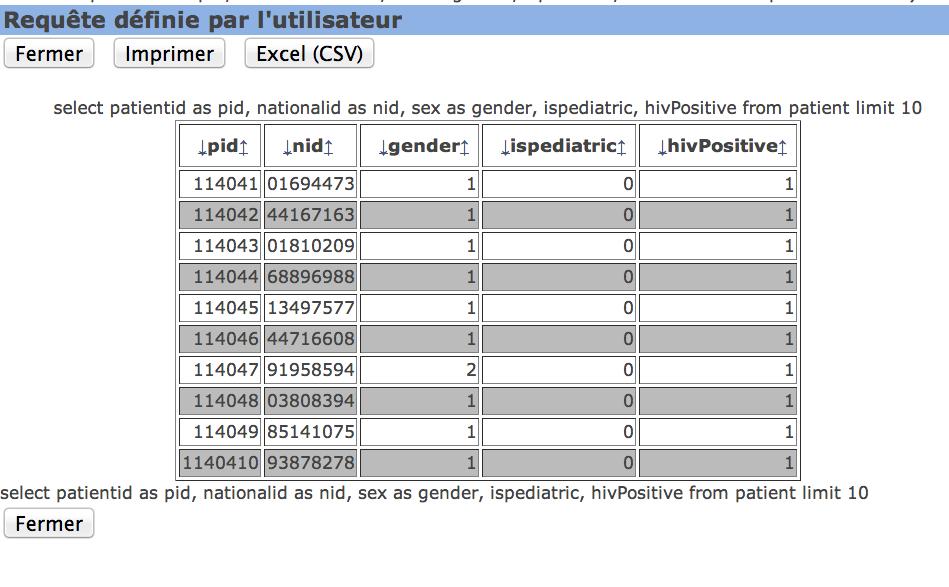


The result window for the above query looks like this:



Notice in the above that the patientid does not appear and the nationalid and sex column headings are formatted. This is a function of how iSanté displays query output in general – many iSanté columns are post-processed for final display. Patientid is always hidden, and if possible, links are automatically generated so that the user can jump immediately to a patient page.

To avoid this translation, just rename the columns in the query. For example, this query can display exactly what is asked for if written as follows:



Column names containing the string ***date*** will be localized for French or English (US) date format, even if they are not actually date fields, so to get a standard date format (yyyy-mm-dd) via the ad hoc window, it is best to do something like:

select date(visitdate) as visitD

this would return, for example, ***2013-10-28***.