**Handout for the Analytics Academy (Part 2: Answers to the Exercises)**

**Exercise #1**

SELECT DISTINCT

 value\_set\_name,

 code\_system\_name,

 code\_value,

 code\_description
FROM hedis\_value\_set\_codes
WHERE latest = 'TRUE'
 AND value\_set\_name = 'Alcohol Use Disorder'
 AND code\_system\_name = 'ICD10CM'
ORDER BY value\_set\_name, code\_system\_name, code\_value, code\_description

**Exercise #2**

**Step #1:** Find the Value Set(s) referring to pneumococcal vaccines. Look for codes in both the HEDIS and eCQM Value Sets

SELECT DISTINCT

 value\_set\_name,

 code\_system\_name,

 code\_value,

 code\_description

FROM hedis\_value\_set\_codes

WHERE latest = 'TRUE'

 AND value\_set\_name ILIKE '%pneumococc%'

 AND code\_system\_name = 'CVX'

ORDER BY value\_set\_name, code\_system\_name, code\_value, code\_description

SELECT DISTINCT

 value\_set\_name,

 code\_system\_name,

 code\_value,

 code\_description

FROM cqm\_value\_set\_codes

WHERE latest = 'TRUE'

 AND value\_set\_name ILIKE '%pneumococc%'

ORDER BY value\_set\_name, code\_system\_name, code\_value, code\_description

**Step #2:** JOIN the HEDIS Value Set table to the immunizations table (eCW) or the relevant\_immunizations table (eCW and NextGen) and use the appropriate Value Set OID

SELECT DISTINCT
 vaccinename,
 count(patientid) AS count
FROM immunizations
 INNER JOIN hedis\_value\_set\_codes ON hedis\_value\_set\_codes.code\_value = immunizations.cvx\_code
 AND hedis\_value\_set\_codes.latest = 'TRUE'
 AND hedis\_value\_set\_codes.value\_set\_oid = '2.16.840.1.113883.3.464.1004.1921'
WHERE givendate BETWEEN '2021-05-01' AND '2022-04-30'
GROUP BY vaccinename

SELECT DISTINCT
 vaccine\_name,
 count(patient\_id) AS count
FROM relevant\_immunizations
 INNER JOIN hedis\_value\_set\_codes ON hedis\_value\_set\_codes.code\_value = relevant\_immunizations.cvx\_code :: VARCHAR
 AND hedis\_value\_set\_codes.latest = 'TRUE'
 AND hedis\_value\_set\_codes.value\_set\_oid = '2.16.840.1.113883.3.464.1004.1921'
WHERE applied\_on BETWEEN '2021-05-01' AND '2022-04-30'
GROUP BY vaccine\_name

**Exercise #3**

**Step #1:** Find and explore the Value Set

SELECT DISTINCT

 value\_set\_name,

 value\_set\_oid,

 code\_system\_name,

 code\_value,

 code\_description

FROM hedis\_value\_set\_codes

WHERE latest = 'TRUE'

 AND value\_set\_name = 'Alcohol Counseling or Other Follow Up Care'

ORDER BY value\_set\_name, code\_system\_name, code\_value, code\_description

**Step #2:** Link the Value Set codes to billed CPT codes and assessment ICD-10 codes

**(eCW)**

SELECT DISTINCT \*

FROM(SELECT

 relevant\_patients.id AS patient\_id,

 servicedt :: DATE AS performed\_on

 FROM relevant\_patients

 INNER JOIN edi\_invoice ON edi\_invoice.patientid = relevant\_patients.id

 AND edi\_invoice.deleteflag = 0

 AND edi\_invoice.voidflag = 0

 INNER JOIN edi\_inv\_cpt ON edi\_inv\_cpt.invoiceid = edi\_invoice.id

 INNER JOIN hedis\_value\_set\_codes

 ON hedis\_value\_set\_codes.code\_value = LEFT(edi\_inv\_cpt.code, 5)

 AND hedis\_value\_set\_codes.latest = 'TRUE'

 AND hedis\_value\_set\_codes.value\_set\_oid = '2.16.840.1.113883.3.464.1004.1437'

 AND hedis\_value\_set\_codes.code\_system\_name = 'CPT'

 UNION

 SELECT

 relevant\_visits.patient\_id,

 relevant\_visits.visit\_date :: DATE AS performed\_on

 FROM relevant\_visits

 INNER JOIN diagnosis ON diagnosis.EncounterId = relevant\_visits.id

 INNER JOIN itemdetail ON itemdetail.itemid = diagnosis.itemid AND propID = 13

 INNER JOIN hedis\_value\_set\_codes

 ON hedis\_value\_set\_codes.code\_value = itemdetail.value

 AND hedis\_value\_set\_codes.latest = 'TRUE'

 AND hedis\_value\_set\_codes.value\_set\_oid = '2.16.840.1.113883.3.464.1004.1437'

 AND hedis\_value\_set\_codes.code\_system\_name = 'ICD10CM') AS alc\_counsel\_temp

**(NextGen)**

SELECT DISTINCT \*

FROM(SELECT

 person\_surrogate\_key.person\_key AS patient\_id,

 patient\_encounter.enc\_timestamp :: DATE AS performed\_on

 FROM patient\_encounter

 INNER JOIN charges ON charges.source\_id = patient\_encounter.enc\_id

 INNER JOIN person\_surrogate\_key ON person\_surrogate\_key.person\_id = patient\_encounter.person\_id

 INNER JOIN hedis\_value\_set\_codes

 ON hedis\_value\_set\_codes.code\_value = charges.service\_item\_id

 AND hedis\_value\_set\_codes.latest = 'TRUE'

 AND hedis\_value\_set\_codes.value\_set\_oid = '2.16.840.1.113883.3.464.1004.1437'

 AND hedis\_value\_set\_codes.code\_system\_name = 'CPT'

 UNION

 SELECT

 person\_surrogate\_key.person\_key AS patient\_id,

 patient\_encounter.enc\_timestamp :: DATE AS performed\_on

 FROM encounter\_diags

 INNER JOIN person\_surrogate\_key ON person\_surrogate\_key.person\_id = encounter\_diags.person\_id

 INNER JOIN patient\_encounter ON patient\_encounter.enc\_id = encounter\_diags.enc\_id

 INNER JOIN hedis\_value\_set\_codes

 ON hedis\_value\_set\_codes.code\_value = encounter\_diags.icd9cm\_code\_id

 AND hedis\_value\_set\_codes.latest = 'TRUE'

 AND hedis\_value\_set\_codes.value\_set\_oid = '2.16.840.1.113883.3.464.1004.1437'

 AND hedis\_value\_set\_codes.code\_system\_name = 'ICD10CM') AS alc\_counsel\_temp

**Exercise #4**

Answers do not involve SQL code. See the answer slides.

**Exercise #5**

--Defines the universe

DROP TABLE IF EXISTS universe;

CREATE TEMPORARY TABLE universe AS

SELECT DISTINCT

 relevant\_patients.id,

 relevant\_patients.mrn,

 relevant\_patients.last\_name,

 relevant\_patients.first\_name,

 relevant\_patients.date\_of\_birth,

 extract(YEAR FROM age('2022-01-01', relevant\_patients.date\_of\_birth)) AS age

FROM relevant\_patients

WHERE extract(YEAR FROM age('2022-01-01', relevant\_patients.date\_of\_birth)) <= 12

 AND relevant\_patients.inactive = 'f'

 AND relevant\_patients.deceased = 'f'

 AND EXISTS(SELECT FROM relevant\_visits

 WHERE relevant\_patients.id = relevant\_visits.patient\_id

 AND uds\_medical IS TRUE

 AND visit\_date BETWEEN '2022-01-01' AND '2022-04-30');

--Risk factor #1 is persistent asthma diagnosis

DROP TABLE IF EXISTS relevant\_asthma\_temp;

CREATE TEMPORARY TABLE relevant\_asthma\_temp AS

SELECT DISTINCT patientid

FROM relevant\_asthma

WHERE persistent IS TRUE

 AND started\_on <= '2022-04-30';

--Risk factor #2 is BMI Percentile above 75

DROP TABLE IF EXISTS relevant\_bmi\_percentile\_temp;

CREATE TEMPORARY TABLE relevant\_bmi\_percentile\_temp AS

SELECT patientid

FROM(SELECT DISTINCT ON (patientid)

 patientid,

 value,

 date

 FROM relevant\_bmi\_percentile

 WHERE date <= '2022-04-30'

 ORDER BY patientid, date DESC) AS last\_bmi

WHERE value > 75;

--Risk factor #3 is ever had positive blood lead lab

DROP TABLE IF EXISTS lead\_lab\_temp;

CREATE TEMPORARY TABLE lead\_lab\_temp AS

SELECT DISTINCT patient\_id

FROM srhc\_lead\_labs

WHERE lab\_result ILIKE '%abnormal%';

SELECT

 universe.mrn,

 universe.last\_name,

 universe.first\_name,

 universe.date\_of\_birth,

 universe.age,

 CASE WHEN relevant\_asthma\_temp.patientid IS NULL THEN FALSE ELSE TRUE END AS has\_asthma,

 CASE WHEN relevant\_bmi\_percentile\_temp.patientid IS NULL THEN FALSE ELSE TRUE END AS has\_high\_bmi,

 CASE WHEN lead\_lab\_temp.patient\_id IS NULL THEN FALSE ELSE TRUE END AS has\_positive\_lead\_lab

FROM universe

 LEFT JOIN relevant\_asthma\_temp ON relevant\_asthma\_temp.patientid = universe.id

 LEFT JOIN relevant\_bmi\_percentile\_temp ON relevant\_bmi\_percentile\_temp.patientid = universe.id

 LEFT JOIN lead\_lab\_temp ON lead\_lab\_temp.patient\_id = universe.id

WHERE relevant\_asthma\_temp.patientid IS NOT NULL

 OR relevant\_bmi\_percentile\_temp.patientid IS NOT NULL

 OR lead\_lab\_temp.patient\_id IS NOT NULL;