```
DROP TABLE IF EXISTS ds_data_prep.icg_patient_flat_table_updated;
CREATE TABLE ds_data_prep.icg_patient_flat_table_updated AS (
SELECT
 icg.patient_id,
 icg.patient_birth_year,
 icg.patient_gender,
 icg.age_during_first_diagnosis,
 icg.age_during_latest_diagnosis,
 icg.BMI_latest,

 icg.common_wt_cm_dx_yn,
 icg.overweight_dx_yn,
 icg.any_wt_cm_dx_yn,
 icg.obesity_dx_yn,
 icg.baom_label_adult_yn,
 icg.baom_label_adolescent_yn,
 icg.overweight_and_wt_cm_dx_yn,
 icg.obesity_or_ow_and_cm_yn,

 icg.group_consult_yn,
 icg.count_group_consult,
 icg.individual_consult_yn,
 icg.count_individual_consult,
 icg.screening_yn,
 icg.count_screening,
 icg.surgery_yn,
 icg.count_surgery,
 icg.first_consult_service_date,
 icg.last_consult_service_date,
 icg.first_surgery_service_date,
 icg.last_surgery_service_date,

 icg.total_rx_claims,
 icg.total_pd_claims,

 icg.total_pd_Saxenda_claims,
 icg.total_pd_Contrave_claims,
 icg.total_pd_Qsymia_claims,
 icg.total_pd_Belviq_claims,
 icg.total_pd_Generic_claims,

 icg.stdaln_PD_nonlifecycle_claims,
 icg.stdaln_PD_lifecycle_claims,
 icg.final_PD_claims,
 icg.stdaln_RJ_nonlifecycle_claims,
 icg.stdaln_RJ_lifecycle_claims,
 icg.stdaln_RV_nonlifecycle_claims,
 icg.stdaln_RV_lifecycle_claims,
```

icg.initial_RV_claims,
icg.initial_RJ_claims,
icg.final_RJ_claims,
icg.final_RV_claims,

icg.prescribed_Saxenda_yn,
icg.prescribed_other_BRANDED_AOMS_yn,
icg.prescribed_GENERIC_AOMS_yn,
icg.total_opc_saxenda,
icg.avg_opc_saxenda,
icg.total_opc_other_branded_AOMS,
icg.avg_opc_other_branded_AOMS,
icg.total_opc_generic_AOMS,
icg.avg_opc_generic_AOMS,

icg.dx_most_freq_prescriber_id,
icg.dx_most_freq_state,
icg.dx_most_freq_zip,
icg.dx_most_freq_plantrak_id,

icg.first_diagnosis_date,
icg.dx_first_prescriber_id,
icg.dx_first_state,
icg.dx_first_zip,
icg.dx_first_plantrak_id,

icg.latest_diagnosis_date,
icg.dx_latest_prescriber_id,
icg.dx_latest_state,
icg.dx_latest_zip,
icg.dx_latest_plantrak_id,

icg.rx_most_freq_prescriber_id,
icg.rx_most_freq_state,
icg.rx_most_freq_zip,
icg.rx_most_freq_plantrak_id,

icg.first_prescription_date,
icg.first_paid_prescription_date,
icg.rx_first_prescriber_id,
icg.rx_first_prescriber_state,
icg.rx_first_prescriber_zip,
icg.rx_first_plantrak_id,
icg.first_brand_prescribed_Saxenda_yn,
icg.first_brand_prescribed_other_branded_AOMs_yn,
icg.first_brand_prescribed_generic_AOMS_yn,

icg.latest_prescription_date,
icg.latest_paid_prescription_date,
icg.rx_latest_prescriber_id,
icg.rx_latest_prescriber_state,
icg.rx_latest_prescriber_zip,
icg.rx_latest_plantrak_id,

```
  icg.latest_brand_prescribed_Saxenda_yn,
  icg.latest_brand_prescribed_other_branded_AOMs_yn,
  icg.latest_brand_prescribed_generic_AOMS_yn,

  icg.joined_prescriber_id,
  icg.nni_saxenda_gsb,
  icg.nni_saxenda_target,
  icg.zip,
  icg.state,
  icg.joined_plantrak_id,
  icg.method_of_payment,
  icg.model_type,

  icg.days_between_first_diag_latest_diag,
  icg.days_between_first_consult_latest_consult,
  icg.days_between_first_surgery_latest_surgery,
  icg.days_between_first_prescr_latest_prescr,
  icg.days_between_first_PD_prescr_latest_PD_prescr,
  icg.days_between_first_consult_latest_surgery,
  icg.days_between_first_diag_latest_prescr,
  icg.days_between_first_diag_first_prescr,
  icg.days_between_latest_diag_latest_prescr,
  icg.days_between_latest_diag_first_prescr

FROM (
  SELECT
    combined.patient_id,
    combined.patient_birth_year,
    combined.patient_gender,
    combined.age_during_first_diagnosis,
    combined.age_during_latest_diagnosis,
    combined.BMI_latest,
    combined.common_wt_cm_dx_yn,
    combined.overweight_dx_yn,
    combined.any_wt_cm_dx_yn,
    combined.obesity_dx_yn,
    combined.baom_label_adult_yn,
    combined.baom_label_adolescent_yn,
    combined.overweight_and_wt_cm_dx_yn,
    combined.obesity_or_ow_and_cm_yn,

    rpd.group_consult_yn,
    rpd.count_group_consult,
    rpd.individual_consult_yn,
    rpd.count_individual_consult,
    rpd.screening_yn,
    rpd.count_screening,
    rpd.surgery_yn,
    rpd.count_surgery,
    rpd.first_consult_service_date,
    rpd.last_consult_service_date,
    rpd.first_surgery_service_date,
    rpd.last_surgery_service_date,
```

r2.total_rx_claims,
r2.total_pd_claims,

r2.total_pd_Saxenda_claims,
r2.total_pd_Contrave_claims,
r2.total_pd_Qsymia_claims,
r2.total_pd_Belviq_claims,
r2.total_pd_Generic_claims,

r2.stdaln_PD_nonlifecycle_claims,
r2.stdaln_PD_lifecycle_claims,
r2.final_PD_claims,
r2.stdaln_RJ_nonlifecycle_claims,
r2.stdaln_RJ_lifecycle_claims,
r2.stdaln_RV_nonlifecycle_claims,
r2.stdaln_RV_lifecycle_claims,
r2.initial_RV_claims,
r2.initial_RJ_claims,
r2.final_RJ_claims,
r2.final_RV_claims,

r2.prescribed_Saxenda_yn,
r2.prescribed_other_BRANDED_AOMS_yn,
r2.prescribed_GENERIC_AOMS_yn,
r2.total_opc_saxenda,
r2.avg_opc_saxenda,
r2.total_opc_other_branded_AOMS,
r2.avg_opc_other_branded_AOMS,
r2.total_opc_generic_AOMS,
r2.avg_opc_generic_AOMS,

rpd.rx_most_freq_prescriber_id,
rpd.rx_most_freq_state,
rpd.rx_most_freq_zip,
rpd.rx_most_freq_plantrak_id,

rpd.first_prescription_date,
rpd.rx_first_prescriber_id,
rpd.rx_first_prescriber_state,
rpd.rx_first_prescriber_zip,
rpd.rx_first_plantrak_id,

rpd.latest_prescription_date,
rpd.rx_latest_prescriber_id,
rpd.rx_latest_prescriber_state,
rpd.rx_latest_prescriber_zip,
rpd.rx_latest_plantrak_id,

rpd.first_brand_prescribed_Saxenda_yn,
rpd.first_brand_prescribed_other_branded_AOMs_yn,
rpd.first_brand_prescribed_generic_AOMS_yn,
rpd.latest_brand_prescribed_Saxenda_yn,

```
   rpd.latest_brand_prescribed_other_branded_AOMs_yn,
   rpd.latest_brand_prescribed_generic_AOMS_yn,

   rpd.dx_most_freq_prescriber_id,
   rpd.dx_most_freq_state,
   rpd.dx_most_freq_zip,
   rpd.dx_most_freq_plantrak_id,

   rpd.first_diagnosis_date,
   rpd.dx_first_prescriber_id,
   rpd.dx_first_state,
   rpd.dx_first_zip,
   rpd.dx_first_plantrak_id,

   rpd.latest_diagnosis_date,
   rpd.dx_latest_prescriber_id,
   rpd.dx_latest_state,
   rpd.dx_latest_zip,
   rpd.dx_latest_plantrak_id,

   rpd.days_between_first_diag_latest_diag,
   rpd.days_between_first_consult_latest_consult,
   rpd.days_between_first_surgery_latest_surgery,
   rpd.days_between_first_prescr_latest_prescr,
   rpd.days_between_first_consult_latest_surgery,
   rpd.days_between_first_diag_latest_prescr,
   rpd.days_between_first_diag_first_prescr,
   rpd.days_between_latest_diag_latest_prescr,
   rpd.days_between_latest_diag_first_prescr,

   rpd.joined_prescriber_id,
   rpd.nni_saxenda_gsb,
   rpd.nni_saxenda_target,
   rpd.zip,
   rpd.state,
   rpd.joined_plantrak_id,
   rpd.method_of_payment,
   rpd.model_type,

   r6.first_paid_prescription_date,
   r6.latest_paid_prescription_date,
   (r6.latest_paid_prescription_date - r6.first_paid_prescription_date) AS
days_between_first_PD_prescr_latest_PD_prescr

  FROM (
   SELECT
     r.patient_id AS rr,
     d.patient_id AS dd,
     (CASE WHEN r.patient_id IS NULL AND d.patient_id IS NOT NULL THEN d.patient_id ELSE r.patient_id
END) AS patient_id,
     (CASE WHEN r.patient_id IS NULL AND d.patient_id IS NOT NULL THEN d.patient_birth_year ELSE
r.patient_birth_year END) AS patient_birth_year,
     (CASE WHEN r.patient_id IS NULL AND d.patient_id IS NOT NULL THEN d.patient_gender ELSE
```

```
r.patient_gender END) AS patient_gender,

    DATE_PART_YEAR(MIN(d.service_date)) - d.patient_birth_year AS age_during_first_diagnosis,
    DATE_PART_YEAR(MAX(d.service_date)) - d.patient_birth_year AS age_during_latest_diagnosis,

    (CASE WHEN age_during_latest_diagnosis > 20 AND d.obesity_dx_yn = 'Y' THEN '>30'
     WHEN age_during_latest_diagnosis > 20 AND d.obesity_dx_yn = 'N' THEN '27'
     ELSE NULL END) AS BMI_latest,

    d.common_wt_cm_dx_yn,
    d.overweight_dx_yn,
    d.any_wt_cm_dx_yn,
    d.obesity_dx_yn,
    d.baom_label_adult_yn,
    d.baom_label_adolescent_yn,
    d.overweight_and_wt_cm_dx_yn,
    d.obesity_or_ow_and_cm_yn
  FROM laad_aom.rx_fact r
  FULL OUTER JOIN laad_aom.dx_branded_aom d
    ON r.patient_id = d.patient_id
  GROUP BY
    r.patient_id,
    d.patient_id,
    r.patient_birth_year,
    d.patient_birth_year,
    r.patient_gender,
    d.patient_gender,
    d.common_wt_cm_dx_yn,
    d.overweight_dx_yn,
    d.any_wt_cm_dx_yn,
    d.obesity_dx_yn,
    d.baom_label_adult_yn,
    d.baom_label_adolescent_yn,
    d.overweight_and_wt_cm_dx_yn,
    d.obesity_or_ow_and_cm_yn
 ) combined

 --LEFT JOINING this subquery to get the count of each type of claim
 LEFT JOIN (
  SELECT
    r.patient_id,

    COUNT(r.claim_id) AS total_rx_claims,
    COUNT(CASE WHEN r.claim_type = 'PD' THEN r.claim_id ELSE NULL END) AS total_pd_claims,

    COUNT(CASE WHEN r.claim_type = 'PD' AND r.brand_name = 'SAXENDA' THEN r.claim_id ELSE NULL
END) AS total_pd_Saxenda_claims,
    COUNT(CASE WHEN r.claim_type = 'PD' AND r.brand_name = 'CONTRAVE' THEN r.claim_id ELSE NULL
END) AS total_pd_Contrave_claims,
    COUNT(CASE WHEN r.claim_type = 'PD' AND r.brand_name = 'QSYMIA' THEN r.claim_id ELSE NULL
END) AS total_pd_Qsymia_claims,
    COUNT(CASE WHEN r.claim_type = 'PD' AND r.brand_name LIKE 'BELVIQ%' THEN r.claim_id ELSE
NULL END) AS total_pd_Belviq_claims,
```

```
    COUNT(CASE WHEN r.claim_type = 'PD' AND (r.brand_name <> 'SAXENDA' AND r.brand_name <>
'CONTRAVE' AND r.brand_name <> 'QSYMIA' AND r.brand_name NOT LIKE 'BELVIQ%') THEN r.claim_id
ELSE NULL END) AS total_pd_Generic_claims,

    COUNT(CASE WHEN r.claim_type = 'PD' AND r.claim_status = 'S' AND r.life_cycle_claims_yn = 'N'
THEN r.claim_id ELSE NULL END) AS stdaln_PD_nonlifecycle_claims,
    COUNT(CASE WHEN r.claim_type = 'PD' AND r.claim_status = 'S' AND r.life_cycle_claims_yn = 'Y'
THEN r.claim_id ELSE NULL END) AS stdaln_PD_lifecycle_claims,
    COUNT(CASE WHEN r.claim_type = 'RJ' AND r.claim_status = 'S' AND r.life_cycle_claims_yn = 'N'
THEN r.claim_id ELSE NULL END) AS stdaln_RJ_nonlifecycle_claims,
    COUNT(CASE WHEN r.claim_type = 'RJ' AND r.claim_status = 'S' AND r.life_cycle_claims_yn = 'Y' THEN
r.claim_id ELSE NULL END) AS stdaln_RJ_lifecycle_claims,
    COUNT(CASE WHEN r.claim_type = 'RV' AND r.claim_status = 'S' AND r.life_cycle_claims_yn = 'N'
THEN r.claim_id ELSE NULL END) AS stdaln_RV_nonlifecycle_claims,
    COUNT(CASE WHEN r.claim_type = 'RV' AND r.claim_status = 'S' AND r.life_cycle_claims_yn = 'Y'
THEN r.claim_id ELSE NULL END) AS stdaln_RV_lifecycle_claims,
    COUNT(CASE WHEN r.claim_type = 'RV' AND r.claim_status = 'I' THEN r.claim_id ELSE NULL END) AS
initial_RV_claims,
    COUNT(CASE WHEN r.claim_type = 'RJ' AND r.claim_status = 'I' THEN r.claim_id ELSE NULL END) AS
initial_RJ_claims,
    COUNT(CASE WHEN r.claim_type = 'RJ' AND r.claim_status = 'F' THEN r.claim_id ELSE NULL END) AS
final_RJ_claims,
    COUNT(CASE WHEN r.claim_type = 'PD' AND r.claim_status = 'F' THEN r.claim_id ELSE NULL END) AS
final_PD_claims,
    COUNT(CASE WHEN r.claim_type = 'RV' AND r.claim_status = 'F' THEN r.claim_id ELSE NULL END) AS
final_RV_claims,

    (CASE WHEN COUNT(CASE WHEN r.brand_name = 'SAXENDA' THEN r.claim_id ELSE NULL END) >= 1
THEN 'Y' ELSE 'N' END) AS prescribed_Saxenda_yn,
    (CASE WHEN COUNT(CASE WHEN r.brand_name LIKE 'BELVIQ%' OR r.brand_name = 'CONTRAVE' OR
r.brand_name = 'QSYMIA' THEN r.claim_id ELSE NULL END) >= 1 THEN 'Y' ELSE 'N' END) AS
prescribed_other_BRANDED_AOMS_yn,
    (CASE WHEN COUNT(CASE WHEN r.brand_name <> 'SAXENDA' AND r.brand_name NOT LIKE
'BELVIQ%' AND r.brand_name <> 'CONTRAVE' AND r.brand_name <> 'QSYMIA' THEN r.claim_id ELSE
NULL END) >= 1 THEN 'Y' ELSE 'N' END) AS prescribed_GENERIC_AOMS_yn,

    (CASE WHEN prescribed_Saxenda_yn = 'Y' THEN (SUM(CASE WHEN r.brand_name = 'SAXENDA' AND
r.claim_type = 'PD' AND r.days_supply <> 0 THEN (r.patient_final_opc / r.days_supply) * 30 ELSE NULL
END)) ELSE NULL END) AS total_opc_saxenda,
    total_opc_saxenda / COUNT(DISTINCT CASE WHEN r.brand_name = 'SAXENDA' AND r.claim_type =
'PD' AND r.days_supply <> 0 THEN r.claim_id ELSE NULL END) AS avg_opc_saxenda,
    (CASE WHEN prescribed_other_BRANDED_AOMS_yn = 'Y' THEN (SUM(CASE WHEN (r.brand_name
LIKE 'BELVIQ%' OR r.brand_name = 'CONTRAVE' OR r.brand_name = 'QSYMIA') AND r.claim_type = 'PD'
AND r.days_supply <> 0 THEN (r.patient_final_opc / r.days_supply) * 30 ELSE NULL END)) ELSE NULL
END) AS total_opc_other_branded_AOMS,
    total_opc_other_branded_AOMS / COUNT(DISTINCT CASE WHEN (r.brand_name LIKE 'BELVIQ%' OR
r.brand_name = 'CONTRAVE' OR r.brand_name = 'QSYMIA') AND r.claim_type = 'PD' AND r.days_supply
<> 0 THEN r.claim_id ELSE NULL END) AS avg_opc_other_branded_AOMS,
    (CASE WHEN prescribed_GENERIC_AOMS_yn = 'Y' THEN (SUM(CASE WHEN (r.brand_name <>
'SAXENDA' AND r.brand_name NOT LIKE 'BELVIQ%' AND r.brand_name <> 'CONTRAVE' AND
r.brand_name <> 'QSYMIA') AND r.claim_type = 'PD' AND r.days_supply <> 0 THEN (r.patient_final_opc /
r.days_supply) * 30 ELSE NULL END)) ELSE NULL END) AS total_opc_generic_AOMS,
    total_opc_generic_AOMS / COUNT(DISTINCT CASE WHEN (r.brand_name <> 'SAXENDA' AND
```

```
r.brand_name NOT LIKE 'BELVIQ%' AND r.brand_name <> 'CONTRAVE' AND r.brand_name <> 'QSYMIA')
AND r.claim_type = 'PD' AND r.days_supply <> 0 THEN r.claim_id ELSE NULL END) AS
avg_opc_generic_AOMS

   FROM laad_aom.rx_fact r
   GROUP BY
    r.patient_id
 ) r2
   ON r2.patient_id = combined.patient_id

 -- LEFT JOIN this mega subquery to look for Px data, most_frequent, first and latest prescriber and
plantrak data
 LEFT JOIN (
   SELECT
    compiled.patient_id,

    compiled.group_consult_yn,
    compiled.count_group_consult,
    compiled.individual_consult_yn,
    compiled.count_individual_consult,
    compiled.screening_yn,
    compiled.count_screening,
    compiled.surgery_yn,
    compiled.count_surgery,
    compiled.first_consult_service_date,
    compiled.last_consult_service_date,
    compiled.first_surgery_service_date,
    compiled.last_surgery_service_date,

    compiled.rx_most_freq_prescriber_id,
    compiled.rx_most_freq_state,
    compiled.rx_most_freq_zip,
    compiled.rx_most_freq_plantrak_id,

    compiled.first_prescription_date,
    compiled.rx_first_prescriber_id,
    compiled.rx_first_prescriber_state,
    compiled.rx_first_prescriber_zip,
    compiled.rx_first_plantrak_id,

    compiled.latest_prescription_date,
    compiled.rx_latest_prescriber_id,
    compiled.rx_latest_prescriber_state,
    compiled.rx_latest_prescriber_zip,
    compiled.rx_latest_plantrak_id,

    compiled.first_brand_prescribed_Saxenda_yn,
    compiled.first_brand_prescribed_other_branded_AOMs_yn,
    compiled.first_brand_prescribed_generic_AOMS_yn,
    compiled.latest_brand_prescribed_Saxenda_yn,
    compiled.latest_brand_prescribed_other_branded_AOMs_yn,
    compiled.latest_brand_prescribed_generic_AOMS_yn,
```

```
 compiled.dx_most_freq_prescriber_id,
 compiled.dx_most_freq_state,
 compiled.dx_most_freq_zip,
 compiled.dx_most_freq_plantrak_id,

 compiled.first_diagnosis_date,
 compiled.dx_first_prescriber_id,
 compiled.dx_first_state,
 compiled.dx_first_zip,
 compiled.dx_first_plantrak_id,

 compiled.latest_diagnosis_date,
 compiled.dx_latest_prescriber_id,
 compiled.dx_latest_state,
 compiled.dx_latest_zip,
 compiled.dx_latest_plantrak_id,

 compiled.days_between_first_diag_latest_diag,
 compiled.days_between_first_consult_latest_consult,
 compiled.days_between_first_surgery_latest_surgery,
 compiled.days_between_first_prescr_latest_prescr,
 compiled.days_between_first_consult_latest_surgery,
 compiled.days_between_first_diag_latest_prescr,
 compiled.days_between_first_diag_first_prescr,
 compiled.days_between_latest_diag_latest_prescr,
 compiled.days_between_latest_diag_first_prescr,

 compiled.joined_prescriber_id,
 pr.nni_saxenda_gsb,
 pr.nni_saxenda_target,
 pr.zip,
 pr.state,
 compiled.joined_plantrak_id,
 pl.method_of_payment,
 pl.model_type

FROM (
 SELECT
  alls.patient_id,

  px.group_consult_yn,
  px.count_group_consult,
  px.individual_consult_yn,
  px.count_individual_consult,
  px.screening_yn,
  px.count_screening,
  px.surgery_yn,
  px.count_surgery,
  px.first_consult_service_date,
  px.last_consult_service_date,
  px.first_surgery_service_date,
  px.last_surgery_service_date,
```

```
    r3.a AS rx_most_freq_prescriber_id,
    r3.rx_most_freq_state,
    r3.rx_most_freq_zip,
    r3.aa AS rx_most_freq_plantrak_id,

    r4.first_prescription_date,
    r4.b AS rx_first_prescriber_id,
    r4.rx_first_prescriber_state,
    r4.rx_first_prescriber_zip,
    r4.bb AS rx_first_plantrak_id,

    r5.latest_prescription_date,
    r5.c AS rx_latest_prescriber_id,
    r5.rx_latest_prescriber_state,
    r5.rx_latest_prescriber_zip,
    r5.cc AS rx_latest_plantrak_id,

    r4.first_brand_prescribed_Saxenda_yn,
    r4.first_brand_prescribed_other_branded_AOMs_yn,
    r4.first_brand_prescribed_generic_AOMS_yn,
    r5.latest_brand_prescribed_Saxenda_yn,
    r5.latest_brand_prescribed_other_branded_AOMs_yn,
    r5.latest_brand_prescribed_generic_AOMS_yn,

    d3.d AS dx_most_freq_prescriber_id,
    d3.dx_most_freq_state,
    d3.dx_most_freq_zip,
    d3.dd AS dx_most_freq_plantrak_id,

    d4.first_diagnosis_date,
    d4.e AS dx_first_prescriber_id,
    d4.dx_first_state,
    d4.dx_first_zip,
    d4.ee AS dx_first_plantrak_id,

    d5.latest_diagnosis_date,
    d5.f AS dx_latest_prescriber_id,
    d5.dx_latest_state,
    d5.dx_latest_zip,
    d5.ff AS dx_latest_plantrak_id,

    (d5.latest_diagnosis_date - d4.first_diagnosis_date) AS days_between_first_diag_latest_diag,
    (px.last_consult_service_date - px.first_consult_service_date) AS
days_between_first_consult_latest_consult,
    (px.last_surgery_service_date - px.first_surgery_service_date) AS
days_between_first_surgery_latest_surgery,
    (r5.latest_prescription_date - r4.first_prescription_date) AS
days_between_first_prescr_latest_prescr,
    (px.last_surgery_service_date - px.first_consult_service_date) AS
days_between_first_consult_latest_surgery,
    (r5.latest_prescription_date - d4.first_diagnosis_date) AS days_between_first_diag_latest_prescr,
    (r4.first_prescription_date - d4.first_diagnosis_date) AS days_between_first_diag_first_prescr,
    (r5.latest_prescription_date - d5.latest_diagnosis_date) AS
```

```
days_between_latest_diag_latest_prescr,
      (r4.first_prescription_date - d5.latest_diagnosis_date) AS days_between_latest_diag_first_prescr,

      (CASE WHEN (d3.d IS NULL AND r3.a IS NOT NULL) THEN (CASE WHEN r4.b = r5.c THEN r4.b ELSE
r3.a END)::BIGINT
            WHEN (d3.d IS NOT NULL AND r3.a IS NULL) THEN (CASE WHEN d4.e = d5.f THEN d4.e ELSE d3.d
END)::BIGINT
            WHEN (r3.a = r4.b AND r3.a = r5.c) OR (r3.a = r4.b AND r3.a = d3.d) OR (r3.a = r4.b AND r3.a =
d4.e) OR (r3.a = r4.b AND r3.a = d5.f) OR (r3.a = r5.c AND r3.a = d3.d) OR (r3.a = r5.c AND r3.a = d4.e) OR
(r3.a = r5.c AND r3.a = d5.f) OR (r3.a = d3.d AND r3.a = d4.e) OR (r3.a = d3.d AND r3.a = d5.f) OR (r3.a =
d4.e AND r3.a = d5.f) THEN r3.a::BIGINT
            WHEN (d3.d = d4.e AND d3.d = d5.f AND (r3.a <> r4.b OR r3.a <> r5.c)) OR (d3.d = d5.f AND d3.d
= r5.c AND (r3.a <> r4.b OR r3.a <> d4.e)) OR (d3.d = d5.f AND d3.d = r4.b AND (r3.a <> r5.c OR r3.a <>
d4.e)) OR (d3.d = d4.e AND d3.d = r5.c AND (r3.a <> r4.b OR r3.a <> d5.f)) OR (d3.d = d4.e AND d3.d =
r4.b AND (r3.a <> r5.c OR r3.a <> d5.f)) OR (d3.d = r5.c AND d3.d = r4.b AND (r3.a <> d4.e OR r3.a <>
d5.f)) THEN d3.d::BIGINT
            WHEN (r4.b = r5.c AND r4.b = d4.e) OR (r4.b = r5.c AND r4.b = d5.f) OR (r4.b = d4.e AND r4.b =
d5.f) THEN r4.b::BIGINT
            WHEN (r5.c = d4.e AND r5.c = d5.f) THEN r5.c::BIGINT ELSE r3.a::BIGINT END) AS
joined_prescriber_id,

      (CASE WHEN (d3.dd IS NULL AND r3.aa IS NOT NULL) THEN (CASE WHEN r4.bb = r5.cc THEN r4.bb
ELSE r3.aa END)::VARCHAR
            WHEN (d3.dd IS NOT NULL AND r3.aa IS NULL) THEN (CASE WHEN d4.ee = d5.ff THEN d4.ee ELSE
d3.dd END)::VARCHAR
            WHEN (r3.aa = r4.bb AND r3.aa = r5.cc) OR (r3.aa = r4.bb AND r3.aa = d3.dd) OR (r3.aa = r4.bb
AND r3.aa = d4.ee) OR (r3.aa = r4.bb AND r3.aa = d5.ff) OR (r3.aa = r5.cc AND r3.aa = d3.dd) OR (r3.aa =
r5.cc AND r3.aa = d4.ee) OR (r3.aa = r5.cc AND r3.aa = d5.ff) OR (r3.aa = d3.dd AND r3.aa = d4.ee) OR
(r3.aa = d3.dd AND r3.aa = d5.ff) OR (r3.aa = d4.ee AND r3.aa = d5.ff) THEN r3.aa::VARCHAR
            WHEN (d3.dd = d4.ee AND d3.dd = d5.ff AND (r3.aa <> r4.bb OR r3.aa <> r5.cc)) OR (d3.dd = d5.ff
AND d3.dd = r5.cc AND (r3.aa <> r4.bb OR r3.aa <> d4.ee)) OR (d3.dd = d5.ff AND d3.dd = r4.bb AND
(r3.aa <> r5.cc OR r3.aa <> d4.ee)) OR (d3.dd = d4.ee AND d3.dd = r5.cc AND (r3.aa <> r4.bb OR r3.aa <>
d5.ff)) OR (d3.dd = d4.ee AND d3.dd = r4.bb AND (r3.aa <> r5.cc OR r3.aa <> d5.ff)) OR (d3.dd = r5.cc
AND d3.dd = r4.bb AND (r3.aa <> d4.ee OR r3.aa <> d5.ff)) THEN d3.dd::VARCHAR
            WHEN (r4.bb = r5.cc AND r4.bb = d4.ee) OR (r4.bb = r5.cc AND r4.bb = d5.ff) OR (r4.bb = d4.ee
AND r4.bb = d5.ff) THEN bb::VARCHAR
            WHEN (r5.cc = d4.ee AND r5.cc = d5.ff) THEN r5.cc::VARCHAR ELSE r3.aa::VARCHAR END) AS
joined_plantrak_id

    --FROM laad_aom.rx_fact r
    FROM (
      SELECT
        a.patient_id
      FROM (
        SELECT
          r.patient_id AS rr,
          d.patient_id AS dd,
          (CASE WHEN r.patient_id IS NULL AND d.patient_id IS NOT NULL THEN d.patient_id ELSE
r.patient_id END) AS patient_id
        FROM laad_aom.rx_fact r
        FULL OUTER JOIN laad_aom.dx_branded_aom d
          ON r.patient_id = d.patient_id
        GROUP BY
```

```
        r.patient_id,
        d.patient_id
      ) a
    ) alls

    --LEFT JOINING this subquery to get data from Px_flat table
    LEFT JOIN (
      SELECT * FROM
      (
          SELECT
            patient_id,
            CASE WHEN SUM(temp2.group_consult_yn)>0 THEN 'Y' ELSE 'N' END AS group_consult_yn,
            SUM(temp2.group_consult_yn) AS count_group_consult,
            CASE WHEN SUM(temp2.individual_consult_yn)>0 THEN 'Y' ELSE 'N' END AS
individual_consult_yn,
            SUM(temp2.individual_consult_yn) AS count_individual_consult,
            CASE WHEN SUM(temp2.screening_yn)>0 THEN 'Y' ELSE 'N' END AS screening_yn,
            SUM(temp2.screening_yn) AS count_screening,
            CASE WHEN SUM(temp2.surgery_yn)>0 THEN 'Y' ELSE 'N' END AS surgery_yn,
            SUM(temp2.surgery_yn) AS count_surgery,


            MIN(consult_service_date) AS first_consult_service_date,
            MAX(consult_service_date) AS last_consult_service_date,
            MIN(surgery_service_date) AS first_surgery_service_date,
            MAX(surgery_service_date) AS last_surgery_service_date

          FROM
            (
              SELECT
                    patient_id, service_date, procedure_code, group_consult_yn, individual_consult_yn,
screening_yn, surgery_yn,
                    CASE
                      WHEN group_consult_yn = 1 or individual_consult_yn = 1 or screening_yn = 1
                      THEN service_date
                      ELSE NULL
                      END AS consult_service_date,
                    CASE
                      WHEN surgery_yn = 1
                      THEN service_date
                      ELSE NULL
                      END AS surgery_service_date


              FROM
                    (
                    SELECT
                          p.patient_id, p.service_date, p.procedure_code,
                          CASE
                            WHEN p.procedure_code='G0473'
                            THEN 1
                            ELSE 0
                            END AS group_consult_yn,
```

```
                        CASE
                          WHEN p.procedure_code='G0447'
                          THEN 1
                          ELSE 0
                          END AS individual_consult_yn,
                         CASE
                          WHEN p.procedure_code='G0449'
                          THEN 1
                          ELSE 0
                          END AS screening_yn,
                        CASE
                          WHEN p.procedure_code in
('43775','43644','S2083','43774','43659','43999','43770','43632','43633','43845','43848','43772','43645','
43771','43647','43888','43846','43773','43886','43887','43648','43843','43847','0316T','43842','0312T','0
314T','0313T','0317T','0315T','0157T','0158T','4496')
                          THEN 1
                          ELSE 0
                          END AS surgery_YN

                    FROM
                      laad_aom.px_fact p

                    WHERE patient_id IN
                       (
                       SELECT DISTINCT p.patient_id
                       FROM laad_aom.px_fact p
                       INNER JOIN laad_aom.dx_branded_aom d
                       ON p.patient_id = d.patient_id
                       INNER JOIN laad_aom.rx_fact r
                       ON p.patient_id = r.patient_id
                       ORDER BY p.patient_id
                       )
                    ORDER BY p.patient_id) AS temp
                    ) AS temp2

               GROUP BY patient_id
      )
    ) px
      ON alls.patient_id = px.patient_id

    -- LEFT JOINING this subquery to get the most frequent rx_prescriber and plantrak
    LEFT JOIN(
      SELECT
        r1.patient_id,
        r1.rx_most_freq_prescriber_ID AS a,
        pr.state AS rx_most_freq_state,
        pr.zip AS rx_most_freq_zip,
        r1.rx_most_freq_plantrak_ID AS aa
      FROM ds_data_prep.icg_prescriber_flat_table pr

      RIGHT JOIN (
        SELECT
          DISTINCT r.patient_id,
```

```
        MAX(r2.MODE_prescrib_ID) AS rx_most_freq_prescriber_ID,
        MAX(r2.MODE_plantrak_id) AS rx_most_freq_plantrak_ID
        FROM laad_aom.rx_fact r
        LEFT JOIN (
          SELECT
            r.patient_id,
            r.prescriber_id,
            (CASE WHEN DENSE_RANK() OVER (PARTITION BY r.patient_id ORDER BY
COUNT(r.prescriber_id) DESC) = 1 THEN r.prescriber_id ELSE NULL END) AS MODE_prescrib_ID,
            (CASE WHEN DENSE_RANK() OVER (PARTITION BY r.patient_id ORDER BY COUNT(r.plantrak_id)
DESC) = 1 THEN r.plantrak_id ELSE NULL END) AS MODE_plantrak_ID
          FROM
            laad_aom.rx_fact r
          GROUP BY
            r.patient_id,
            r.prescriber_id,
            r.plantrak_id
        ) r2
          ON r2.patient_id = r.patient_id

        GROUP BY
          r.patient_id,
          r.prescriber_id,
          r.plantrak_id
      ) r1
        ON r1.rx_most_freq_prescriber_ID = pr.prescriber_id
    ) r3
      ON alls.patient_id = r3.patient_id

    --LEFT JOINING this subquery to get the first rx_prescriber and plantrak
    LEFT JOIN (
      SELECT
        r1.patient_id,
        r1.first_prescription_date,
        r1.first_prescriber_id AS b,
        pr.state AS rx_first_prescriber_state,
        pr.zip AS rx_first_prescriber_zip,
        r1.first_plantrak_id AS bb,
        r1.first_brand_prescribed_Saxenda_yn,
        r1.first_brand_prescribed_other_branded_AOMs_yn,
        r1.first_brand_prescribed_generic_AOMS_yn
      FROM ds_data_prep.icg_prescriber_flat_table pr

      RIGHT JOIN (
        SELECT
          DISTINCT r.patient_id,
          r1.first_prescription_date,
          r1.first_brand_prescribed_Saxenda_yn,
          r1.first_brand_prescribed_other_branded_AOMs_yn,
          r1.first_brand_prescribed_generic_AOMS_yn,
          r1.first_prescriber_id,
          r1.first_plantrak_id
        FROM laad_aom.rx_fact r
```

```
        LEFT JOIN (
          SELECT
            r.patient_id,
            MIN(r.service_date) AS first_prescription_date,

            r3.first_brand_prescribed_Saxenda_yn,
            r3.first_brand_prescribed_other_branded_AOMs_yn,
            r3.first_brand_prescribed_generic_AOMS_yn,

            MAX(r2.first_pres_ID) AS first_prescriber_id,
            MAX(r2.first_plan_ID) AS first_plantrak_id,

            DENSE_RANK() OVER (PARTITION BY r.patient_id ORDER BY MIN(r.service_date)) AS ranking
          FROM laad_aom.rx_fact r

          LEFT JOIN (
            SELECT
              r.patient_id,
              r.service_date,
              (CASE WHEN DENSE_RANK() OVER (PARTITION BY r.patient_id ORDER BY MIN(r.service_date))
= 1 THEN r.prescriber_id ELSE NULL END) AS first_pres_ID,
              (CASE WHEN DENSE_RANK() OVER (PARTITION BY r.patient_id ORDER BY MIN(r.service_date))
= 1 THEN r.plantrak_id ELSE NULL END) AS first_plan_ID
            FROM laad_aom.rx_fact r
            GROUP BY
              r.patient_id,
              r.service_date,
              r.prescriber_id,
              r.plantrak_id
          ) r2
            ON
              r.patient_id = r2.patient_id

          LEFT JOIN (
            SELECT
              r.patient_id,
              r.service_date,
              (CASE WHEN COUNT(CASE WHEN r.brand_name = 'SAXENDA' THEN r.brand_name ELSE NULL
END) >= 1 THEN 'Y' ELSE 'N' END) AS first_brand_prescribed_Saxenda_yn,
              (CASE WHEN COUNT(CASE WHEN r.brand_name LIKE 'BELVIQ%' OR r.brand_name =
'CONTRAVE' OR r.brand_name = 'QSYMIA' THEN r.brand_name ELSE NULL END) >= 1 THEN 'Y' ELSE 'N'
END) AS first_brand_prescribed_other_branded_AOMs_yn,
              (CASE WHEN COUNT(CASE WHEN r.brand_name <> 'SAXENDA' AND r.brand_name NOT LIKE
'BELVIQ%' AND r.brand_name <> 'CONTRAVE' AND r.brand_name <> 'QSYMIA' THEN r.brand_name ELSE
NULL END) >= 1 THEN 'Y' ELSE 'N' END) AS first_brand_prescribed_generic_AOMS_yn,
              DENSE_RANK() OVER (PARTITION BY r.patient_id ORDER BY MIN(r.service_date)) AS ranker
            FROM laad_aom.rx_fact r
            GROUP BY
              r.patient_id,
              r.service_date
            ORDER BY
              r.patient_id,
              r.service_date
```

```
    ) r3
      ON r3.patient_id = r.patient_id

    WHERE r3.ranker = 1
    GROUP BY
      r.patient_id,
      r3.first_brand_prescribed_Saxenda_yn,
      r3.first_brand_prescribed_other_branded_AOMs_yn,
      r3.first_brand_prescribed_generic_AOMS_yn,

      r.prescriber_id,
      r.plantrak_id
   ) r1
   ON r1.patient_id = r.patient_id
   WHERE r1.ranking = 1
   AND first_prescriber_id IS NOT NULL
   AND first_plantrak_id IS NOT NULL
  ) r1
   ON r1.first_prescriber_id = pr.prescriber_id
) r4
  ON alls.patient_id = r4.patient_id

-- LEFT JOINING this subquery to get latest rx_prescriber and plantrak
LEFT JOIN (
  SELECT
    r1.patient_id,
    r1.latest_prescription_date,
    r1.latest_prescriber_id AS c,
    pr.state AS rx_latest_prescriber_state,
    pr.zip AS rx_latest_prescriber_zip,
    r1.latest_plantrak_id AS cc,
    r1.latest_brand_prescribed_Saxenda_yn,
    r1.latest_brand_prescribed_other_branded_AOMs_yn,
    r1.latest_brand_prescribed_generic_AOMS_yn
  FROM ds_data_prep.icg_prescriber_flat_table pr

  RIGHT JOIN (
    SELECT
      DISTINCT r.patient_id,
      r1.latest_prescription_date,
      r1.latest_brand_prescribed_Saxenda_yn,
      r1.latest_brand_prescribed_other_branded_AOMs_yn,
      r1.latest_brand_prescribed_generic_AOMS_yn,
      r1.latest_prescriber_id,
      r1.latest_plantrak_id
    FROM laad_aom.rx_fact r
    LEFT JOIN (
      SELECT
        r.patient_id,
        MAX(r.service_date) AS latest_prescription_date,

        r3.latest_brand_prescribed_Saxenda_yn,
        r3.latest_brand_prescribed_other_branded_AOMs_yn,
```

```
        r3.latest_brand_prescribed_generic_AOMS_yn,

        MAX(r2.latest_pres_ID) AS latest_prescriber_id,
        MAX(r2.latest_plan_ID) AS latest_plantrak_id,

        DENSE_RANK() OVER (PARTITION BY r.patient_id ORDER BY MAX(r.service_date) DESC) AS
ranking

      FROM laad_aom.rx_fact r

      LEFT JOIN (
        SELECT
          r.patient_id,
          r.service_date,
          (CASE WHEN DENSE_RANK() OVER (PARTITION BY r.patient_id ORDER BY MAX(r.service_date)
DESC) = 1 THEN r.prescriber_id ELSE NULL END) AS latest_pres_ID,
          (CASE WHEN DENSE_RANK() OVER (PARTITION BY r.patient_id ORDER BY MAX(r.service_date)
DESC) = 1 THEN r.plantrak_id ELSE NULL END) AS latest_plan_ID
        FROM laad_aom.rx_fact r
        GROUP BY
          r.patient_id,
          r.service_date,
          r.prescriber_id,
          r.plantrak_id
      ) r2
        ON
          r.patient_id = r2.patient_id

      LEFT JOIN (
        SELECT
          r.patient_id,
          r.service_date,
          (CASE WHEN COUNT(CASE WHEN r.brand_name = 'SAXENDA' THEN r.brand_name ELSE NULL
END) >= 1 THEN 'Y' ELSE 'N' END) AS latest_brand_prescribed_Saxenda_yn,
          (CASE WHEN COUNT(CASE WHEN r.brand_name LIKE 'BELVIQ%' OR r.brand_name =
'CONTRAVE' OR r.brand_name = 'QSYMIA' THEN r.brand_name ELSE NULL END) >= 1 THEN 'Y' ELSE 'N'
END) AS latest_brand_prescribed_other_branded_AOMs_yn,
          (CASE WHEN COUNT(CASE WHEN r.brand_name <> 'SAXENDA' AND r.brand_name NOT LIKE
'BELVIQ%' AND r.brand_name <> 'CONTRAVE' AND r.brand_name <> 'QSYMIA' THEN r.brand_name ELSE
NULL END) >= 1 THEN 'Y' ELSE 'N' END) AS latest_brand_prescribed_generic_AOMS_yn,
          DENSE_RANK() OVER (PARTITION BY r.patient_id ORDER BY MAX(r.service_date) DESC) AS
ranker
        FROM laad_aom.rx_fact r
        GROUP BY
          r.patient_id,
          r.service_date
        ORDER BY
          r.patient_id,
          r.service_date
      ) r3
        ON r3.patient_id = r.patient_id

      WHERE r3.ranker = 1
```

```
        GROUP BY
          r.patient_id,
          r3.latest_brand_prescribed_Saxenda_yn,
          r3.latest_brand_prescribed_other_branded_AOMs_yn,
          r3.latest_brand_prescribed_generic_AOMS_yn,
          r.prescriber_id,
          r.plantrak_id
        ) r1
          ON r1.patient_id = r.patient_id
          WHERE r1.ranking = 1
          AND latest_prescriber_id IS NOT NULL
          AND latest_plantrak_id IS NOT NULL
      ) r1
        ON r1.latest_prescriber_id = pr.prescriber_id
    ) r5
      ON alls.patient_id = r5.patient_id

    --LEFT JOINING this subquery to get most frequent diagnosis doctor and plantrak
    LEFT JOIN (
      SELECT
        d3.patient_id,
        d3.dx_most_freq_prescriber_id AS d,
        pr.state AS dx_most_freq_state,
        pr.zip AS dx_most_freq_zip,
        d3.dx_most_freq_plantrak_id AS dd
      FROM ds_data_prep.icg_prescriber_flat_table pr
      RIGHT JOIN (
        SELECT
          d1.patient_id,
          MAX(d1.prescriber_id) AS dx_most_freq_prescriber_id,
          MAX(d2.plantrak_id) AS dx_most_freq_plantrak_id
        FROM (
          SELECT
            d.patient_id,
            d.prescriber_id,
            DENSE_RANK() OVER (PARTITION BY d.patient_id ORDER BY COUNT(d.prescriber_id) DESC) AS
ranking
          FROM laad_aom.dx_branded_aom d
          GROUP BY
            d.patient_id,
            d.prescriber_id
        ) d1
        LEFT JOIN (
          SELECT
            d.patient_id,
            d.plantrak_id,
            DENSE_RANK() OVER (PARTITION BY d.patient_id ORDER BY COUNT(d.plantrak_id) DESC) AS
ranking
          FROM laad_aom.dx_branded_aom d
          GROUP BY
            d.patient_id,
            d.plantrak_id
```

```
   ) d2
     ON d1.patient_id = d2.patient_id
   WHERE d1.ranking = 1
   AND d2.ranking = 1
   GROUP BY
    d1.patient_id
  ) d3
   ON d3.dx_most_freq_prescriber_id = pr.prescriber_id
) d3
  ON alls.patient_id = d3.patient_id

--LEFT JOINING this subquery to get first diagnosis doctor and plantrak
LEFT JOIN (
  SELECT
   d4.patient_id,
   d4.first_diagnosis_date,
   d4.dx_first_prescriber_id AS e,
   pr.state AS dx_first_state,
   pr.zip AS dx_first_zip,
   d4.dx_first_plantrak_id AS ee
  FROM ds_data_prep.icg_prescriber_flat_table pr
  RIGHT JOIN (
  SELECT
   d1.patient_id,
   d1.first_diagnosis_date,
   MAX(d1.dx_first_prescriber_id) AS dx_first_prescriber_id,
   MAX(d1.dx_first_plantrak_id) AS dx_first_plantrak_id
  FROM (
   SELECT
    d.patient_id,
    MIN(d.service_date) AS first_diagnosis_date,
    d.prescriber_id AS dx_first_prescriber_id,
    d.plantrak_id AS dx_first_plantrak_id,
    DENSE_RANK() OVER (PARTITION BY d.patient_id ORDER BY MIN(d.service_date)) AS ranking
   FROM laad_aom.dx_branded_aom d
   GROUP BY
    d.patient_id,
    d.prescriber_id,
    d.plantrak_id
   ) d1
  WHERE d1.ranking = 1
  GROUP BY
   d1.patient_id,
   d1.first_diagnosis_date
  ) d4
   ON d4.dx_first_prescriber_id = pr.prescriber_id
) d4
  ON alls.patient_id = d4.patient_id

--LEFT JOINING this subquery to get latest diagnosis doctor and plantrak
LEFT JOIN (
  SELECT
   d5.patient_id,
```

```
        d5.latest_diagnosis_date,
        d5.dx_latest_prescriber_id AS f,
        pr.state AS dx_latest_state,
        pr.zip AS dx_latest_zip,
        d5.dx_latest_plantrak_id AS ff
       FROM ds_data_prep.icg_prescriber_flat_table pr
       RIGHT JOIN (
       SELECT
        d1.patient_id,
        d1.latest_diagnosis_date,
        MAX(d1.dx_latest_prescriber_id) AS dx_latest_prescriber_id,
        MAX(d1.dx_latest_plantrak_id) AS dx_latest_plantrak_id
       FROM (
        SELECT
         d.patient_id,
         MAX(d.service_date) AS latest_diagnosis_date,
         d.prescriber_id AS dx_latest_prescriber_id,
         d.plantrak_id AS dx_latest_plantrak_id,
         DENSE_RANK() OVER (PARTITION BY d.patient_id ORDER BY MAX(d.service_date) DESC) AS
ranking
        FROM laad_aom.dx_branded_aom d
        GROUP BY
         d.patient_id,
         d.prescriber_id,
         d.plantrak_id
        ) d1
       WHERE d1.ranking = 1
       GROUP BY
        d1.patient_id,
        d1.latest_diagnosis_date
       ) d5
        ON d5.dx_latest_prescriber_id = pr.prescriber_id
     ) d5
      ON alls.patient_id = d5.patient_id
   ) compiled

   LEFT JOIN ds_data_prep.icg_prescriber_flat_table pr
    ON compiled.joined_prescriber_id = pr.prescriber_id

   LEFT JOIN ds_data_prep.icg_plantrak_flat_table pl
    ON compiled.joined_plantrak_id = pl.plantrak_id

  ) rpd
   ON combined.patient_id = rpd.patient_id

  --LEFT JOINING this subquery to get first and latest prescription date with paid claims
  LEFT JOIN (
   SELECT
    r.patient_id,
    MIN(r1.service_date_paid_claim) AS first_paid_prescription_date,
    MAX(r1.service_date_paid_claim) AS latest_paid_prescription_date
   FROM laad_aom.rx_fact r
   LEFT JOIN (
```

```
    SELECT
      r.patient_id,
      r.service_date,
      r.claim_type,
      (CASE r.claim_type WHEN 'PD' THEN r.service_date ELSE NULL END) AS service_date_paid_claim
    FROM laad_aom.rx_fact r
      GROUP BY
        r.patient_id,
        r.service_date,
        r.claim_type
  ) r1
    ON r1.patient_id = r.patient_id
  GROUP BY
    r.patient_id
) r6
  ON combined.patient_id = r6.patient_id

GROUP BY
  combined.patient_id,
  combined.patient_birth_year,
  combined.patient_gender,
  combined.age_during_first_diagnosis,
  combined.age_during_latest_diagnosis,
  combined.BMI_latest,
  combined.common_wt_cm_dx_yn,
  combined.overweight_dx_yn,
  combined.any_wt_cm_dx_yn,
  combined.obesity_dx_yn,
  combined.baom_label_adult_yn,
  combined.baom_label_adolescent_yn,
  combined.overweight_and_wt_cm_dx_yn,
  combined.obesity_or_ow_and_cm_yn,

  rpd.group_consult_yn,
  rpd.count_group_consult,
  rpd.individual_consult_yn,
  rpd.count_individual_consult,
  rpd.screening_yn,
  rpd.count_screening,
  rpd.surgery_yn,
  rpd.count_surgery,
  rpd.first_consult_service_date,
  rpd.last_consult_service_date,
  rpd.first_surgery_service_date,
  rpd.last_surgery_service_date,

  r2.total_rx_claims,
  r2.total_pd_claims,

  r2.total_pd_Saxenda_claims,
  r2.total_pd_Contrave_claims,
  r2.total_pd_Qsymia_claims,
  r2.total_pd_Belviq_claims,
```

r2.total_pd_Generic_claims,

r2.stdaln_PD_nonlifecycle_claims,
r2.stdaln_PD_lifecycle_claims,
r2.final_PD_claims,
r2.stdaln_RJ_nonlifecycle_claims,
r2.stdaln_RJ_lifecycle_claims,
r2.stdaln_RV_nonlifecycle_claims,
r2.stdaln_RV_lifecycle_claims,
r2.initial_RV_claims,
r2.initial_RJ_claims,
r2.final_RJ_claims,
r2.final_RV_claims,

r2.prescribed_Saxenda_yn,
r2.prescribed_other_BRANDED_AOMS_yn,
r2.prescribed_GENERIC_AOMS_yn,
r2.total_opc_saxenda,
r2.avg_opc_saxenda,
r2.total_opc_other_branded_AOMS,
r2.avg_opc_other_branded_AOMS,
r2.total_opc_generic_AOMS,
r2.avg_opc_generic_AOMS,

rpd.rx_most_freq_prescriber_id,
rpd.rx_most_freq_state,
rpd.rx_most_freq_zip,
rpd.rx_most_freq_plantrak_id,

rpd.first_prescription_date,
rpd.rx_first_prescriber_id,
rpd.rx_first_prescriber_state,
rpd.rx_first_prescriber_zip,
rpd.rx_first_plantrak_id,

rpd.latest_prescription_date,
rpd.rx_latest_prescriber_id,
rpd.rx_latest_prescriber_state,
rpd.rx_latest_prescriber_zip,
rpd.rx_latest_plantrak_id,

rpd.first_brand_prescribed_Saxenda_yn,
rpd.first_brand_prescribed_other_branded_AOMs_yn,
rpd.first_brand_prescribed_generic_AOMS_yn,
rpd.latest_brand_prescribed_Saxenda_yn,
rpd.latest_brand_prescribed_other_branded_AOMs_yn,
rpd.latest_brand_prescribed_generic_AOMS_yn,

rpd.dx_most_freq_prescriber_id,
rpd.dx_most_freq_state,
rpd.dx_most_freq_zip,
rpd.dx_most_freq_plantrak_id,

```
   rpd.first_diagnosis_date,
   rpd.dx_first_prescriber_id,
   rpd.dx_first_state,
   rpd.dx_first_zip,
   rpd.dx_first_plantrak_id,

   rpd.latest_diagnosis_date,
   rpd.dx_latest_prescriber_id,
   rpd.dx_latest_state,
   rpd.dx_latest_zip,
   rpd.dx_latest_plantrak_id,

   rpd.days_between_first_diag_latest_diag,
   rpd.days_between_first_consult_latest_consult,
   rpd.days_between_first_surgery_latest_surgery,
   rpd.days_between_first_prescr_latest_prescr,
   rpd.days_between_first_consult_latest_surgery,
   rpd.days_between_first_diag_latest_prescr,
   rpd.days_between_first_diag_first_prescr,
   rpd.days_between_latest_diag_latest_prescr,
   rpd.days_between_latest_diag_first_prescr,

   rpd.joined_prescriber_id,
   rpd.nni_saxenda_gsb,
   rpd.nni_saxenda_target,
   rpd.zip,
   rpd.state,
   rpd.joined_plantrak_id,
   rpd.method_of_payment,
   rpd.model_type,

   r6.first_paid_prescription_date,
   r6.latest_paid_prescription_date
) icg
WHERE icg.patient_birth_year <> 0
AND icg.joined_prescriber_id IS NOT NULL
AND icg.joined_plantrak_id IS NOT NULL
AND ((icg.first_prescription_date IS NOT NULL AND icg.latest_prescription_date IS NOT NULL
      AND ((icg.first_diagnosis_date IS NOT NULL AND icg.latest_diagnosis_date IS NOT NULL)
       OR (icg.first_diagnosis_date IS NULL AND icg.latest_diagnosis_date IS NULL)))
   OR (icg.first_diagnosis_date IS NOT NULL AND icg.latest_diagnosis_date IS NOT NULL
      AND ((icg.first_prescription_date IS NOT NULL AND icg.latest_prescription_date IS NOT NULL)
       OR (icg.first_prescription_date IS NULL and icg.latest_prescription_date IS NULL))))
)
```