```
Imports System
Imports MySql.Data.MySqlClient
Imports System.Data
Imports System.ComponentModel
Imports System.Threading
Public Class procedure_ei_code

    Sub vmt_pop_ab2gai_avg_scenario(ByVal scene As String, ByVal worker As BackgroundWorker, _
        ByVal e As DoWorkEventArgs)

        'Procedure used to allocate VMT and Pop by air basin to GAI

        Dim percent_complete As Integer
        Dim sql_db As New procedure_gui
        Dim vmt_pop_ab, vmt_pop_ab_temp, vmt_pop_gai, vmt_pop_gai_temp, vmt_pop_gai_temp1 As String

        vmt_pop_ab = "vmt_pop_" & scene
        vmt_pop_ab_temp = database_outputs & ".vmt_pop_" & scene & "_temp"
        vmt_pop_gai = database_outputs & ".vmt_pop_" & scene & "_gai"
        vmt_pop_gai_temp = database_outputs & ".vmt_pop_" & scene & "_gai_temp"
        vmt_pop_gai_temp1 = database_outputs & ".vmt_pop_" & scene & "_gai_temp1"

        'add index to AB2GAI table

        If sql_db.index_exist("vmt_spatial_allocation_within_ab_avg", "ab2gai") = False Then

            sql = "ALTER TABLE `vmt_spatial_allocation_within_ab_avg` " & _
            "ADD INDEX `ab2gai` (`vehicle_class` ASC, `AB` ASC);"

            sql_db.MySQL_Docmd(sql)

        End If

        'adjust population for fraction in CA

        If sql_db.index_exist("time_in_ca", "fleet_CY") = False Then

            sql = "ALTER TABLE `time_in_ca` " & _
            "ADD INDEX `fleet_cy` (`fleet` ASC, `CY` ASC);"
```

```
        sql_db.MySQL_Docmd(sql)

    End If

    percent_complete = Int(1 * 100 / 11)
    worker.ReportProgress(percent_complete)

    'change SFBAB to SFAB

    sql = "UPDATE `" & vmt_pop_ab & "` SET air_basin = 'SFAB' WHERE air_basin = 'SFBAB';"
    sql_db.MySQL_Docmd(sql)

    percent_complete = Int(2 * 100 / 11)
    worker.ReportProgress(percent_complete)

    'add inventory_category_ab2gai`

    If sql_db.index_exist(vmt_pop_ab, "Fleet") = False Then

        sql = "ALTER TABLE `" & vmt_pop_ab & "` " & _
              "ADD INDEX `Fleet` ( `Fleet` ASC) ;"
        sql_db.MySQL_Docmd(sql)

        percent_complete = Int(3 * 100 / 11)
        worker.ReportProgress(percent_complete)

    End If

    sql = "DROP Table IF EXISTS " & vmt_pop_ab_temp & ";"
    sql_db.MySQL_Docmd(sql)

    percent_complete = Int(4 * 100 / 11)
    worker.ReportProgress(percent_complete)

    sql = "Create Table " & vmt_pop_ab_temp & " ENGINE = MYISAM as " & _
    "SELECT v.*, f.inventory_category_ab2gai " & _
    "FROM " & vmt_pop_ab & " v INNER JOIN vehicle_fleet_category f " & _
    "ON v.Fleet = f.Inventory_Category;"

    sql_db.MySQL_Docmd(sql)
```

```
percent_complete = Int(5 * 100 / 11)
worker.ReportProgress(percent_complete)

Dim tablename_temp As String
tablename_temp = "vmt_pop_" & scene & "_temp"

sql = "select * from information_schema.statistics " & _
"where table_name = '" & tablename_temp & "' and index_name = 'ab_fleet' " & _
"and table_schema = '" & database_outputs & "';"

If sql_db.component_exist(sql) = False Then

    sql = "ALTER TABLE " & vmt_pop_ab_temp & "  ADD INDEX `ab_fleet` " & _
       "(`air_basin` ASC, `inventory_category_ab2gai` ASC) ;"

    sql_db.MySQL_Docmd(sql)

    percent_complete = Int(6 * 100 / 11)
    worker.ReportProgress(percent_complete)

End If

sql = "DROP Table IF EXISTS " & vmt_pop_gai_temp1 & ";"
sql_db.MySQL_Docmd(sql)

'sql = "Create Table " & vmt_pop_gai_temp1 & " ENGINE = MYISAM as " & _
'"SELECT v.CY, v.Fleet, v.Air_Basin, " & _
'"s.gai, v.MY, v.Age, pm_control, " & _
'"v.pop*s.Fraction_AB as pop, " & _
'"v.VMT*s.Fraction_AB as vmt " & _
'"FROM " & vmt_pop_ab_temp & " v INNER JOIN vmt_spatial_allocation_within_ab_avg s ON " & _
'"(v.inventory_category_ab2gai = s.Vehicle_class) " & _
'"AND (v.Air_Basin = s.AB);"

'Revision 02/10/2014
'To adjust chasis model year -engine year mismatch, let MY= MY - 1
sql = "Create Table " & vmt_pop_gai_temp1 & " ENGINE = MYISAM as " & _
"SELECT v.CY, v.Fleet, v.Air_Basin, " & _
"s.gai, v.MY-1 as MY, v.Age, pm_control, " & _
```

```
    "v.pop*s.Fraction_AB as pop, " & _
    "v.VMT*s.Fraction_AB as vmt " & _
    "FROM " & vmt_pop_ab_temp & " v INNER JOIN vmt_spatial_allocation_within_ab_avg s ON " & _
    "(v.inventory_category_ab2gai = s.Vehicle_class) " & _
    "AND (v.Air_Basin = s.AB);"

    sql_db.MySQL_Docmd(sql)

    sql = "ALTER TABLE " & vmt_pop_gai_temp1 & " " & _
       "ADD INDEX `fleet_cy` (`fleet` ASC, `CY` ASC);"

    sql_db.MySQL_Docmd(sql)

    sql = "DROP Table IF EXISTS " & vmt_pop_gai_temp & ";"
    sql_db.MySQL_Docmd(sql)

    'adjust for the fraction of time in CA

    sql = "Create Table " & vmt_pop_gai_temp & " ENGINE = MYISAM as " & _
    "SELECT e.*, e.pop*t.fraction_in_ca as pop_ca from " & vmt_pop_gai_temp1 & " e INNER JOIN " & _
    "time_in_ca t ON (e.fleet = t.Fleet) AND " & _
    "(e.CY = t.CY); "

    sql_db.MySQL_Docmd(sql)

    percent_complete = Int(7 * 100 / 11)
    worker.ReportProgress(percent_complete)

    sql = "select * from information_schema.statistics " & _
"where table_name = '" & vmt_pop_gai_temp & "' and index_name = 'my' " & _
"and table_schema = '" & database_outputs & "';"

    If sql_db.component_exist(sql) = False Then

        sql = "ALTER TABLE " & vmt_pop_gai_temp & " " & _
       "ADD INDEX `MY` (`MY` ASC) ;"

        sql_db.MySQL_Docmd(sql)

        percent_complete = Int(8 * 100 / 11)
```

```
        worker.ReportProgress(percent_complete)

    End If

    If sql_db.index_exist("year_range", "my") = False Then

        sql = "ALTER TABLE `year_range` " & _
            "ADD INDEX `my` (`Year` ASC) ;"

        sql_db.MySQL_Docmd(sql)

    End If

    sql = "DROP Table IF EXISTS " & vmt_pop_gai & ";"
    sql_db.MySQL_Docmd(sql)

    'Create table to add column er_my
    sql = "Create Table " & vmt_pop_gai & " ENGINE = MYISAM as " & _
    "SELECT v.*, y.er_my FROM " & vmt_pop_gai_temp & " v INNER JOIN year_range y ON " & _
    "v.MY = y.Year;"

    sql_db.MySQL_Docmd(sql)

    percent_complete = Int(9 * 100 / 11)
    worker.ReportProgress(percent_complete)

    'For running emissions, need an index to speed up pop vmt to speed bin.

    'Same error as before:
    'For running emissions, need the index to speed up pop vmt to speed bin.
    Dim vmt_pop_gai_2 As String

    vmt_pop_gai_2 = "vmt_pop_" & scene & "_gai"

    sql = "select * from information_schema.statistics " & _
    "where table_name = '" & vmt_pop_gai_2 & "' and index_name = 'emission_index' " & _
    "and table_schema = '" & database_outputs & "';"


    If sql_db.component_exist(sql) = False Then
```

```
            sql = "ALTER TABLE " & vmt_pop_gai & "  ADD INDEX `emission_index` " & _
            "(`CY` ASC, `fleet` ASC, `gai` ASC, `er_MY` ASC, `Age` ASC) ;"

            sql_db.MySQL_Docmd(sql)

            percent_complete = Int(10 * 100 / 11)
            worker.ReportProgress(percent_complete)

        End If

        'Delete temporary tables

        sql = "DROP Table IF EXISTS " & vmt_pop_ab_temp & ";"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & vmt_pop_gai_temp & ";"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & vmt_pop_gai_temp1 & ";"
        sql_db.MySQL_Docmd(sql)

        percent_complete = Int(11 * 100 / 11)
        worker.ReportProgress(percent_complete)

    End Sub

    Sub running_er_by_spd(ByVal year As Integer, ByVal g As Integer)

        'Procedure to calculate running emission rates

        Dim sql_db As New procedure_gui
        Dim ei_table As String

        'Check whether tables already exist

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates1_ODO;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates2_Deterioration;"
```

```
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates3_Reflash;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".vehicle_fleet_spd;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates4_fcf;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates4_fcf1;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates6_speed_correction;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates7_speed_correction;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".speed_cy_hr;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates12_speed_correction;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates9_speed_correction;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates10_Humidity;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates11_Humidity_Factor;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates_w;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates_w2;"
        sql_db.MySQL_Docmd(sql)
```

```
sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates_Month;"
sql_db.MySQL_Docmd(sql)

sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates_month_hr;"
sql_db.MySQL_Docmd(sql)


sql = "DROP Table IF EXISTS " & database_outputs & ".emission_rates_hr;"
sql_db.MySQL_Docmd(sql)

sql = "DROP Table IF EXISTS " & database_outputs & ".er_running_by_hr;"
sql_db.MySQL_Docmd(sql)

sql = "DROP Table IF EXISTS " & database_outputs & ".er_running_by_month;"
sql_db.MySQL_Docmd(sql)

sql = "DROP Table IF EXISTS " & database_outputs & ".er_running12;"
sql_db.MySQL_Docmd(sql)

sql = "DROP Table IF EXISTS " & database_outputs & ".er_running_by_spd;"
sql_db.MySQL_Docmd(sql)


'Estimated Odometer reading for selected calendar year

'sql = "Create Table " & database_outputs & ".EmissionRates1_ODO ENGINE = MYISAM as " & _
'"SELECT " & year & " as CY, Fleet_deter, " & _
'"If((" & year & "-age)>=1965, " & year & "-age, 1965) as MY, " & _
'"Age, vehicle_class_er, ODO FROM accrual_rate_deter;"

'Revision 02/10/2014
'Adjust for chasis model year-engine year mismatch: convert accrual age (model age) to engine model year
' Engine model year = Chasis model year-1 = Cyr - vehicle age - 1

sql = "Create Table " & database_outputs & ".EmissionRates1_ODO ENGINE = MYISAM as SELECT " & year & " as CY,
Fleet_deter, " & _
"If((" & year & "-age-1)>=1965, " & year & "-age-1, 1965) as MY, " & _
"Age, vehicle_class_er, ODO FROM accrual_rate_deter;"
```

```
sql_db.MySQL_Docmd(sql)

'Add combined index
sql = "ALTER TABLE " & database_outputs & ".EmissionRates1_ODO " & _
"ADD INDEX `Vehicle_MY` (`vehicle_class_er` ASC, `MY` ASC); "

sql_db.MySQL_Docmd(sql)

'Deterioration

sql = "Create Table " & database_outputs & ".EmissionRates2_Deterioration ENGINE = MYISAM as " & _
"SELECT e.CY, e.MY, e.Age, " & _
"e.Fleet_deter, b.vehicle_class, " & _
"b.HC_ZM+e.ODO/10000*b.HC_DR AS HC_det, " & _
"b.CO_ZM+e.ODO/10000*b.CO_DR AS CO_det, " & _
"b.NOX_ZM+e.ODO/10000*b.NOX_DR AS NOX_det, " & _
"b.PM_ZM+e.ODO/10000*b.PM_DR AS PM_det, " & _
"b.CO2_ZM+e.ODO/10000*b.CO2_DR AS CO2_det " & _
"FROM " & database_outputs & ".EmissionRates1_ODO e " & _
"INNER JOIN base_emission_factor b ON (e.vehicle_class_er = b.vehicle_class) " & _
"AND (e.MY = b.MY);"

sql_db.MySQL_Docmd(sql)

'Add index
sql = "ALTER TABLE " & database_outputs & ".EmissionRates2_Deterioration " & _
"ADD INDEX `Vehicle_CY_MY` (`vehicle_class` ASC, `CY` ASC, `MY` ASC); "

sql_db.MySQL_Docmd(sql)

'Reflash
sql = "Create Table " & database_outputs & ".EmissionRates3_Reflash ENGINE = MYISAM as " _
& "SELECT e.CY, e.MY, " & _
"e.Age, e.Fleet_deter, " & _
"e.vehicle_class, HC_det*HC_AF AS HC_reflash, " & _
"CO_det*CO_AF AS CO_reflash, PM_det*PM_AF AS PM_reflash, NOx_det*NOx_AF AS NOx_reflash, " & _
"CO2_det*CO2_AF AS CO2_reflash " & _
"FROM " & database_outputs & ".EmissionRates2_Deterioration e INNER JOIN reflash f " & _
"ON (e.vehicle_class = f.vehicle_class) AND (e.CY = f.Calendar_Year) " & _
"AND (e.MY = f.Model_Year);"
```

```
sql_db.MySQL_Docmd(sql)

'Add index
sql = "ALTER TABLE " & database_outputs & ".EmissionRates3_Reflash " & _
"ADD INDEX `Vehicle_CY_MY` (`vehicle_class` ASC, `CY` ASC, `MY` ASC); "

sql_db.MySQL_Docmd(sql)

'Fuel Correction

sql = "Create Table " & database_outputs & ".EmissionRates4_fcf1 ENGINE = MYISAM as " & _
"SELECT e.CY, g.gai, " & _
"e.Fleet_deter, e.vehicle_class, " & _
"f.Model_Year, e.Age, HC_reflash*HC_fcf AS HC_fcfed, CO_reflash*CO_fcf AS CO_fcfed, " & _
"NOX_reflash*NOX_fcf AS NOX_fcfed, PM_reflash*PM_fcf AS PM_fcfed, CO2_reflash*CO2_fcf AS CO2_fcfed " & _
"FROM " & database_outputs & ".EmissionRates3_Reflash e INNER JOIN " & _
"(fuel_correction_factor f INNER JOIN geographic_regions g " & _
"ON f.Area = g.fcf_area) ON (e.vehicle_class = " & _
"f.vehicle_class) AND (e.CY = f.Calendar_Year) AND " & _
"(e.MY = f.Model_Year) WHERE (((g.gai)= " & g & "));"

sql_db.MySQL_Docmd(sql)

'Add index
sql = "ALTER TABLE " & database_outputs & ".EmissionRates4_fcf1 ADD INDEX `fleet_deter` (`Fleet_deter` ASC); "

sql_db.MySQL_Docmd(sql)

'Create new table to add columns

sql = "Create Table " & database_outputs & ".EmissionRates4_fcf ENGINE = MYISAM as " & _
"SELECT f.*, v.Inventory_Category " & _
"FROM " & database_outputs & ".EmissionRates4_fcf1 f INNER JOIN vehicle_fleet_category v " & _
"ON f.Fleet_deter = v.fleet_deter;"

sql_db.MySQL_Docmd(sql)

'Relative humidity correction
```

```
        Dim A As Double = -0.09132
        Dim B As Double = 0.01594
        Dim C As Double = -0.00029
        Dim D As Double = 0.00000437

        sql = "Create Table " & database_outputs & ".EmissionRates10_Humidity ENGINE = MYISAM as " & _
        "SELECT t.gai, t.Month, t.Hour, " & _
        "if(temperature >=40, Humidity*(" & A & " + " & B & "*temperature + " & C & "*power(temperature, 2)" & _
        "+ " & D & "*power(temperature, 3)), humidity*0.36196) As Humidity1 " & _
        "FROM temperature t INNER JOIN humidity_relative h ON (t.gai = h.gai) AND " & _
        "(t.Month = h.Month) AND (t.Hour = h.Hour) WHERE t.gai = " & g & ";"


        sql_db.MySQL_Docmd(sql)


        'Cap humidity to 200

        sql = "UPDATE " & database_outputs & ".EmissionRates10_Humidity SET Humidity1 = 200 " & _
            "WHERE (((" & database_outputs & ".EmissionRates10_Humidity.Humidity1)>200));"

        sql_db.MySQL_Docmd(sql)

        Dim MClass As Double = -0.0026
        Dim Htest As Integer = 75

        sql = "Create Table " & database_outputs & ".EmissionRates11_Humidity_Factor ENGINE = MYISAM as " & _
        "SELECT gai, Month, Hour, " & _
       "((1+(-0.0047)*( " & Htest & "-75))*(1+" & MClass & "*(Humidity1-75)))/" & _
       "(1+" & MClass & "*( " & Htest & "-75)) As RHUM_CF " & _
       "FROM " & database_outputs & ".EmissionRates10_Humidity;"

        sql_db.MySQL_Docmd(sql)

        'Nox humidity correction

        sql = "Create Table " & database_outputs & ".EmissionRates_month_hr  ENGINE = MYISAM as " & _
        "SELECT s.CY, s.gai, vehicle_class, s.Model_Year, s.Age, s.Inventory_Category, " & _
        "month, h.hour as Hr,  NOX_fcfed*RHUM_CF as NOx," & _
        "HC_fcfed as HC, CO_fcfed as CO, PM_fcfed as PM, " & _
```

```
    "CO2_fcfed as CO2 FROM " & database_outputs & ".EmissionRates4_fcf s INNER JOIN " & _
    "" & database_outputs & ".EmissionRates11_Humidity_Factor h ON " & _
    "(s.gai = h.gai);"

    sql_db.MySQL_Docmd(sql)


    ei_table = database_outputs & ".er_running_by_hr"

    'Annual emission rates
    sql = "Create Table " & ei_table & " ENGINE = MYISAM as  " & _
    "SELECT e.CY, Inventory_Category, vehicle_class, e.gai, e.Model_Year, e.Age, e.Hr, 'a' as Season, " & _
    "Sum(e.NOx)/12 AS NOx, Sum(e.HC)/12 AS HC, " & _
    "Sum(e.CO)/12 AS CO, Sum(e.PM)/12 AS PM, " & _
    "Sum(e.CO2)/12 AS CO2 FROM " & database_outputs & ".EmissionRates_month_hr e " & _
    "where e.month<13 " & _
    "GROUP BY e.CY, e.Inventory_Category, vehicle_class, e.gai, e.Model_Year, e.Age, e.Hr, Season;"

    sql_db.MySQL_Docmd(sql)

    'Append seasonal emission rates

    sql = "Insert into " & ei_table & " (CY, Inventory_Category, vehicle_class, gai,  " & _
"Model_Year, Age, Hr, season, NOx, HC, CO, PM, CO2) " & _
"SELECT e.CY, Inventory_Category, vehicle_class, e.gai, e.Model_Year, e.Age, e.Hr, 's' as Season, " & _
"Sum(e.NOx) AS NOx, Sum(e.HC) AS HC, " & _
"Sum(e.CO) AS CO, Sum(e.PM) AS PM, " & _
"Sum(e.CO2) AS CO2 FROM " & database_outputs & ".EmissionRates_month_hr e " & _
"where (e.Month=13) " & _
"GROUP BY e.CY, e.Inventory_Category, vehicle_class, e.gai, e.Model_Year, e.Age, e.Hr, season;"

    sql_db.MySQL_Docmd(sql)

    sql = "Insert into " & ei_table & " (CY, Inventory_Category, vehicle_class, gai,  " & _
    "Model_Year, Age, Hr, season, NOx, HC, CO, PM, CO2) " & _
    "SELECT e.CY, Inventory_Category, vehicle_class, e.gai, e.Model_Year, e.Age, e.Hr, 'w' as Season, " & _
    "Sum(e.NOx) AS NOx, Sum(e.HC) AS HC, " & _
    "Sum(e.CO) AS CO, Sum(e.PM) AS PM, " & _
    "Sum(e.CO2) AS CO2 FROM " & database_outputs & ".EmissionRates_month_hr e " & _
    "where (e.Month=14) " & _
```

```
"GROUP BY e.CY, e.Inventory_Category, vehicle_class, e.gai, e.Model_Year, e.Age, e.Hr, season;"

sql_db.MySQL_Docmd(sql)

'combine er hr
sql = "Create Table " & database_outputs & ".er_running12  ENGINE = MYISAM as " & _
"SELECT e.CY, t.inventory_category, vehicle_class, e.gai, " & _
"e.Model_Year, e.Age, season, " & _
"Sum(e.NOx*t.Percentage)/100  AS NOx, " & _
"Sum(e.PM*t.Percentage)/100  AS PM, " & _
"Sum(e.CO*t.Percentage)/100  AS CO, " & _
"Sum(e.HC*t.Percentage)/100  AS HC, " & _
"Sum(e.CO2*t.Percentage)/100  AS CO2 " & _
"FROM " & database_outputs & ".er_running_by_hr e INNER JOIN temporal_hr_d_truck t ON (e.Hr = " & _
"t.Hour) AND (e.Inventory_Category = t.Inventory_Category) " & _
"GROUP BY e.CY, t.inventory_category, vehicle_class, " & _
"e.gai, e.Model_Year, e.Age, season;"

sql_db.MySQL_Docmd(sql)

sql = "ALTER TABLE " & database_outputs & ".er_running12  " & _
"ADD INDEX `veh_MY` (`Model_Year` ASC, `vehicle_class` ASC) ;"

sql_db.MySQL_Docmd(sql)

'Calculate Speed correction factor

sql = "Create Table " & database_outputs & ".EmissionRates6_speed_correction ENGINE = MYISAM as " _
& "SELECT speed_correction.vehicle_class_er, speed_bin.Speed_Bin, " & _
"speed_bin.Speed, speed_correction.Model_Year, " & _
"HC_spd2*power(Speed_bin_mp, 2)-HC_spd*Speed_bin_mp+HC_CON AS HC_speed_correction, " & _
"CO_spd2*power(Speed_bin_mp, 2)-CO_spd*Speed_bin_mp+CO_CON AS CO_speed_correction, " & _
"NOX_spd2*power(Speed_bin_mp, 2)-NOX_spd*Speed_bin_mp+NOX_CON AS NOX_speed_correction, " & _
"PM_spd2*power(Speed_bin_mp, 2)-PM_spd*Speed_bin_mp+PM_CON AS PM_speed_correction, " & _
"CO2_spd2*power(Speed_bin_mp, 2)-CO2_spd*Speed_bin_mp+CO2_CON AS CO2_speed_correction " & _
"FROM speed_correction INNER JOIN speed_bin ON speed_correction.SPEED=speed_bin.Speed;"

sql_db.MySQL_Docmd(sql)

'Add index
```

```
sql = "ALTER TABLE " & database_outputs & ".EmissionRates6_speed_correction " & _
    "ADD INDEX `Vehicle_MY` (`vehicle_class_er` ASC, Model_Year ASC); "

sql_db.MySQL_Docmd(sql)


'Apply Speed correction factor to emissions rates

sql = "Create Table " & database_outputs & ".er_running_by_spd ENGINE = MYISAM as " & _
"SELECT e.CY, e.gai, Inventory_Category, " & _
"e.Model_Year, e.Age, s.Speed_Bin, season, " & _
"HC*HC_speed_correction as HC, CO*CO_speed_correction as CO, " & _
"NOX*NOX_speed_correction as NOX, PM*PM_speed_correction as PM, " & _
"CO2*CO2_speed_correction as CO2 " & _
"FROM " & database_outputs & ".er_running12 e INNER JOIN " & _
"" & database_outputs & ".EmissionRates6_speed_correction s ON " & _
"(e.Model_Year = s.Model_Year) AND " & _
"(e.vehicle_class = s.vehicle_class_er);"

sql_db.MySQL_Docmd(sql)

sql = "ALTER TABLE " & database_outputs & ".er_running_by_spd " & _
"ADD INDEX `ei_running` (`CY` ASC, `gai` ASC, `Inventory_Category` ASC, `Model_Year` ASC, " & _
"`Age` ASC, `Speed_Bin` ASC) ;"

sql_db.MySQL_Docmd(sql)


'Drop temporary tables

sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates1_ODO;"
sql_db.MySQL_Docmd(sql)

sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates2_Deterioration;"
sql_db.MySQL_Docmd(sql)

sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates3_Reflash;"
sql_db.MySQL_Docmd(sql)

sql = "DROP Table IF EXISTS " & database_outputs & ".vehicle_fleet_spd;"
```

```
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates4_fcf;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates4_fcf1;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates6_speed_correction;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates7_speed_correction;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".speed_cy_hr;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates12_speed_correction;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates9_speed_correction;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates10_Humidity;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates11_Humidity_Factor;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates_w;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates_w2;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates_Month;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & database_outputs & ".EmissionRates_month_hr;"
        sql_db.MySQL_Docmd(sql)
```

```
    sql = "DROP Table IF EXISTS " & database_outputs & ".emission_rates_hr;"
    sql_db.MySQL_Docmd(sql)

    sql = "DROP Table IF EXISTS " & database_outputs & ".er_running_by_hr;"
    sql_db.MySQL_Docmd(sql)


    sql = "DROP Table IF EXISTS " & database_outputs & ".er_running_by_month;"
    sql_db.MySQL_Docmd(sql)

    sql = "DROP Table IF EXISTS " & database_outputs & ".er_running12;"
    sql_db.MySQL_Docmd(sql)


End Sub

Sub vmt_pop2_spd(ByVal scene As String, ByVal i As Integer, ByVal g As Integer)

    'Procedure to allocate vmt pop to spd bin

    Dim sql_db As New procedure_gui
    Dim vmt_pop_gai, vmt_pop_gai_hr, vmt_pop_gai_spd As String

    vmt_pop_gai = database_outputs & ".vmt_pop_" & scene & "_gai"
    vmt_pop_gai_spd = database_outputs & ".vmt_pop_" & scene & "_gai_spd"
    vmt_pop_gai_hr = database_outputs & ".vmt_pop_" & scene & "_gai_hr"

    If sql_db.index_exist("temporal_hr_d_truck", "fleet") = False Then
        sql = "ALTER TABLE `temporal_hr_d_truck` ADD INDEX `fleet` (`inventory_category` ASC) ;"
        sql_db.MySQL_Docmd(sql)
    End If

    If sql_db.index_exist("speed_cy", "index") = False Then
        sql = "ALTER TABLE `speed_cy` ADD INDEX `index` " & _
    "(`GAI` ASC, `speed_cy` ASC, `vehicle_class_spd` ASC) ;"
        sql_db.MySQL_Docmd(sql)
    End If

    If sql_db.index_exist("speed_cy_gai_hr", "index") = False Then
```

```
    sql = "ALTER TABLE `speed_cy_gai_hr` ADD INDEX `index` " & _
"(`gai` ASC, `cy` ASC, `vehicle_class_spd` ASC, `period_code` ASC) ;"
    sql_db.MySQL_Docmd(sql)
End If

If sql_db.index_exist("speed_profile", "index") = False Then
    sql = "ALTER TABLE `speed_profile` ADD INDEX `index` " & _
"(`GAI` ASC, `vehicle_class_spd` ASC, `calendar_year` ASC, `period_code` ASC) ;"
    sql_db.MySQL_Docmd(sql)
End If

sql = "DROP Table IF EXISTS " & database_outputs & ".speed_cy_hr;"
sql_db.MySQL_Docmd(sql)

sql = "Create Table " & database_outputs & ".speed_cy_hr ENGINE = MYISAM as " & _
"SELECT s.gai, s.Vehicle_Class_spd, c.CY, h.Hr," & _
"s.Speed_Bin, s.Percentage " & _
"FROM speed_cy_gai_hr h INNER JOIN (speed_profile s INNER JOIN speed_cy c ON (s.gai = c.gai) " & _
"AND (s.Vehicle_Class_spd = c.Vehicle_Class_spd) " & _
"AND (s.Calendar_Year = c.speed_cy)) ON (h.gai = s.gai) AND (h.cy = s.Calendar_Year) AND " & _
"(h.Period_Code = s.Period_Code) AND (h.Vehicle_Class_spd = c.Vehicle_Class_spd) Where c.cy = " & i & " and s.gai = "
& g & ";"

sql_db.MySQL_Docmd(sql)

sql = "select * from information_schema.statistics " & _
"where table_name = 'speed_cy_hr' and index_name = 'index' " & _
"and table_schema = '" & database_outputs & "';"

If sql_db.component_exist(sql) = False Then

    sql = "ALTER TABLE " & database_outputs & ".speed_cy_hr ADD INDEX `index` " & _
    "(`vehicle_class_spd` ASC, `gai` ASC, `cy` ASC, `hr` ASC) ;"
    sql_db.MySQL_Docmd(sql)

End If

sql = "DROP Table IF EXISTS " & database_outputs & ".vmt_pop_temp;"
sql_db.MySQL_Docmd(sql)
```

```
    sql = " Create Table " & database_outputs & ".vmt_pop_temp ENGINE = MYISAM as " & _
    "SELECT v.CY, v.Fleet, v.Air_Basin, v.gai, v.MY, v.er_my, v.Age, h.Hour, " & _
    "v.pm_control, v.pop_ca*h.percentage/100 pop_ca, v.vmt*h.percentage/100 vmt  " & _
    "FROM " & vmt_pop_gai & " v INNER JOIN temporal_hr_d_truck h " & _
    "ON v.Fleet = h.inventory_category " & _
    "Where  v.cy = " & i & " and v.gai = " & g & ";"

    sql_db.MySQL_Docmd(sql)

    'Create table to add new vehicle_class_spd column

    sql = "DROP Table IF EXISTS " & vmt_pop_gai_hr & ";"
    sql_db.MySQL_Docmd(sql)

    sql = "Create Table " & vmt_pop_gai_hr & " ENGINE = MYISAM as " & _
    "SELECT v.*, f.vehicle_class_spd  " & _
    "FROM " & database_outputs & ".vmt_pop_temp v INNER JOIN vehicle_fleet_category f " & _
    "ON  v.Fleet = f.Inventory_Category;"

    sql_db.MySQL_Docmd(sql)

    sql = "select * from information_schema.statistics " & _
    "where table_name = 'vmt_pop_" & scene & "_gai_hr' and index_name = 'index' " & _
    "and table_schema = '" & database_outputs & "';"

    If sql_db.component_exist(sql) = False Then
        sql = "ALTER TABLE " & vmt_pop_gai_hr & " ADD INDEX `index` " & _
    "(`vehicle_class_spd` ASC, `gai` ASC, `cy` ASC, `hour` ASC) ;"
        sql_db.MySQL_Docmd(sql)
    End If

    sql = "DROP Table IF EXISTS " & vmt_pop_gai_spd & ";"
    sql_db.MySQL_Docmd(sql)

    'Remove hour column
    sql = " Create Table " & vmt_pop_gai_spd & " ENGINE = MYISAM as " & _
"SELECT v.CY, v.Fleet, v.Air_Basin, v.gai, v.MY, v.er_my, v.Age, " & _
"v.pm_control, s.speed_bin, " & _
"sum(vmt*Percentage)/100 vmt, " & _
"sum(pop_ca*Percentage)/100 pop_ca " & _
```

```
    "FROM " & vmt_pop_gai_hr & " v INNER JOIN " & database_outputs & ".speed_cy_hr s ON " & _
    "(v.vehicle_class_spd = s.Vehicle_Class_spd) and " & _
    "(v.CY = s.CY) AND " & _
    "(v.gai = s.gai) AND " & _
    "(v.hour = s.hr) group by v.CY, v.Fleet, v.Air_Basin, v.gai, v.MY, v.er_my, v.Age, " & _
    "v.pm_control, s.speed_bin;"

      sql_db.MySQL_Docmd(sql)

      sql = "ALTER TABLE " & vmt_pop_gai_spd & " " & _
      "ADD INDEX `ei_running` (`CY` ASC, `Fleet` ASC, `gai` ASC, `er_my` ASC, `Age` ASC, `speed_bin` ASC) ;"

      sql_db.MySQL_Docmd(sql)

      'Drop temporary tables
      sql = "DROP Table IF EXISTS " & database_outputs & ".speed_cy_hr;"
      sql_db.MySQL_Docmd(sql)

      sql = "DROP Table IF EXISTS " & database_outputs & ".vmt_pop_temp;"
      sql_db.MySQL_Docmd(sql)

      sql = "DROP Table IF EXISTS " & vmt_pop_gai_hr & ";"
      sql_db.MySQL_Docmd(sql)

End Sub

Sub running_ei_by_spd(ByVal scene As String, ByVal cy As Integer, ByVal gai As Integer)

      'Procedure to calculate running emissions
      Dim sql_db As New procedure_gui
      Dim ei_running, ei_running_temp1, ei_running_temp2, ei_running_temp3, er_running, vmt_pop_table As String

      ei_running = database_outputs & ".ei_running_by_spd"
      ei_running_temp1 = database_outputs & ".ei_running_by_spd_temp1"
      ei_running_temp2 = database_outputs & ".ei_running_by_spd_temp2"
      ei_running_temp3 = database_outputs & ".ei_running_by_spd_temp3"
      er_running = database_outputs & ".er_running_by_spd"

      vmt_pop_table = database_outputs & ".vmt_pop_" & scene & "_gai_spd"
```

```
sql = "DROP Table IF EXISTS " & ei_running_temp1 & ";"
sql_db.MySQL_Docmd(sql)

'May 25, 2011, calculate Daily emissions

sql = "Create Table " & ei_running_temp1 & " ENGINE = MYISAM as " & _
 "SELECT e.CY, e.gai, v.fleet, v.MY, v.er_MY, e.speed_bin, season, pm_control, pop_ca, v.vmt/operation_day vmt, " & _
 "v.VMT*e.Nox/907184.74/operation_day as NOx, v.VMT*e.PM/907184.74*pm_control/operation_day as PM, " & _
 "v.VMT*e.CO/907184.74*pm_control/operation_day as CO, " & _
 "v.VMT*e.HC/907184.74*pm_control/operation_day as HC, v.VMT*e.CO2/907184.74/operation_day as CO2 " & _
 "FROM (" & er_running & " e INNER JOIN operation_day O ON e.inventory_category = O.fleet) " & _
 "INNER JOIN " & vmt_pop_table & " v ON (e.Age = v.Age) " & _
 "AND (e.Model_Year = v.er_MY) AND (e.gai = v.gai) " & _
 "AND (e.inventory_category = v.Fleet) AND (e.CY = v.CY) AND (e.speed_bin = v.speed_bin) " & _
"where" & season_sql_annual & season_sql_summer & season_sql_winter & ";"

sql_db.MySQL_Docmd(sql)

'2% increase on CO2 with PM control
sql = "UPDATE " & ei_running_temp1 & "  SET CO2= CO2*1.02 where pm_control = 0.15;"

sql_db.MySQL_Docmd(sql)

sql = "DROP Table IF EXISTS " & ei_running_temp2 & ";"
sql_db.MySQL_Docmd(sql)

sql = "Create Table " & ei_running_temp2 & " ENGINE = MYISAM as " & _
    "SELECT g.area, e.*, e.CO2/3.1445*s.sulfur_ppm/1000000*2 SOx " & _
    "from ((sulfur_content s INNER JOIN geographic_regions g ON " & _
    "s.fcf_area = g.FCF_area) INNER JOIN  " & ei_running_temp1 & " e ON g.gai = e.gai) " & _
    "INNER JOIN year_range ON (year_range.Year = e.CY)  " & _
    "And (s.CY_sulfur = year_range.cy_sulfur);"

sql_db.MySQL_Docmd(sql)

sql = "DROP Table IF EXISTS " & ei_running_temp3 & ";"
sql_db.MySQL_Docmd(sql)

sql = "Create Table " & ei_running_temp3 & " ENGINE = MYISAM as " & _
    "SELECT e.area, e.gai, e.CY CalYr, e.fleet Veh2011, v.EMFAC2007_category Veh2007, " & _
```

```
        " e.season, e.speed_bin, 'DSL' fuel, e.er_MY MdlYr, e.pop_ca, e.vmt, " & _
        "e.NOx, e.PM PM10, e.PM*0.92 PM2_5, e.HC*1.4417 TOG, e.HC*1.2664 ROG, e.CO, e.CO2, e.SOx, " & _
        "e.CO2*0.09 Fuel_DSL from vehicle_fleet_category v INNER JOIN  " & ei_running_temp2 & " e " & _
        "ON (v.inventory_category = e.fleet);"

    sql_db.MySQL_Docmd(sql)

    sql = "DROP Table IF EXISTS " & ei_running & ";"
    sql_db.MySQL_Docmd(sql)

    sql = "Create Table " & ei_running & " ENGINE = MYISAM as " & _
        "SELECT 'RUNEX' as Process, e.area, e.gai, e.CalYr, e.season, " & emfac_cat & " Veh, " & _
        " e.fuel, e.MdlYr, e.speed_bin, sum(e.pop_ca) pop_CA, sum(e.vmt) vmt, " & _
        "sum(e.NOx) NOx, sum(e.PM10) PM10, sum(e.PM2_5) PM2_5, " & _
        "sum(e.TOG) TOG, sum(e.ROG) ROG, sum(e.CO) CO, sum(e.CO2) CO2, sum(e.SOx) SOx, " & _
        "sum(e.Fuel_DSL) Fuel_DSL from " & ei_running_temp3 & " e group by e.area, e.gai, " & _
        " e.CalYr, " & emfac_cat & ", e.season, e.speed_bin, e.fuel, e.MdlYr;"

    sql_db.MySQL_Docmd(sql)

    'export running emissions by speed file

    If save2folder = True Then

        If output_by_spd = True Then

            Dim filename As String = "emfac_hd_output_gai_" & gai.ToString & "_cy_" & cy.ToString & "_by_spd.csv"
            filename = savepath & "\" & filename
            filename = Replace(filename, "\", "\\")

            sql = "SELECT 'Process', 'Area', 'GAI', 'CalYr', 'Season', 'Veh', 'Fuel', " & _
        "'MdlYr', 'Speed_bin', 'Pop_CA', 'VMT', 'NOx', 'PM10', 'PM2_5', 'TOG', 'ROG', 'CO', 'CO2', 'SOx', 'Fuel_DSL' " & _
          "UNION SELECT * INTO OUTFILE '" & filename & "' " & _
          "FIELDS TERMINATED BY ',' FROM " & ei_running & ";"

            sql_db.MySQL_Docmd(sql)

        End If  'whether to export ei by spd

    End If    'whether to save file to folder
```

```
If save2server = True Then

    tablename = database_outputs & ".emfac_hd_ei_output_by_spd_gai_" & gai.ToString

    'save running emissions by speed outputs to server

    sql = "Insert into " & tablename & "  " & _
    "SELECT * from " & ei_running & " ;"

    sql_db.MySQL_Docmd(sql)

End If

'summary table

Dim by_process As String = ""
Dim by_my As String = ""
Dim group_by_my As String = ""

If ei_by_process = True Then
    by_process = "'RUNEX'"
Else
    by_process = "'All'"
End If

If ei_by_my = True Then
    by_my = "e.MdlYr"
    group_by_my = ", e.MdlYr"
Else
    by_my = "'All' as MdlYr"
    group_by_my = ""
End If

'Table to hold emissions estimation for all processes by CY by GAI
tablename = database_outputs & ".emfac_hd_ei_output_summary"

sql = "Insert into " & tablename & " (Process, Area, GAI, CalYr, Season, Veh, Fuel,  MdlYr, " & _
"pop_CA, VMT, NOx, PM10, PM2_5, TOG, ROG, CO, CO2, SOx, Fuel_DSL) " & _
```

```
        "SELECT " & by_process & " as Process, e.area, e.gai, e.CalYr, e.season, e.Veh, e.fuel, " & _
        "" & by_my & ", sum(pop_CA) Pop_CA, sum(e.vmt) vmt, " & _
        "sum(e.NOx) NOx, sum(e.PM10) PM10, sum(e.PM2_5) PM2_5, " & _
        "sum(e.TOG) TOG, sum(e.ROG) ROG, sum(e.CO) CO, sum(e.CO2) CO2, sum(e.SOx) SOx, " & _
        "sum(e.Fuel_DSL) Fuel_DSL from " & ei_running & " e group by e.area, e.gai, " & _
        " e.CalYr, e.season, e.fuel, Veh" & group_by_my & ";"

        sql_db.MySQL_Docmd(sql)

        'Delete temporary tables

        sql = "DROP Table IF EXISTS " & ei_running_temp1 & ";"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & ei_running_temp2 & ";"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP Table IF EXISTS " & ei_running_temp3 & ";"
        sql_db.MySQL_Docmd(sql)

    End Sub

    Sub idle_er(ByVal cy As Integer, ByVal gai As Integer)

        'Procedure to calculate idle emission rates
        Dim sql_db As New procedure_gui
        Dim table As String = ""

        table = "idle_emission_factor"

        'Check whether table already exists

        sql = "DROP TABLE IF EXISTS " & database_outputs & ".EmissionRatesHighIdle_winter;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP TABLE IF EXISTS " & database_outputs & ".EmissionRatesHighIdle_summer;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP TABLE IF EXISTS " & database_outputs & ".er_idle_1;"
        sql_db.MySQL_Docmd(sql)
```

```
sql = "DROP TABLE IF EXISTS " & database_outputs & ".er_idle;"
sql_db.MySQL_Docmd(sql)

'Winter highe idle

sql = "Create Table " & database_outputs & ".EmissionRatesHighIdle_winter ENGINE = MYISAM as  " & _
"SELECT f.Season, e.vehicle_class_Abb, " & _
"e.Model_Year, e.MY_Range, HC_IDLE*HC_ICF as HC_winter, " & _
"CO_IDLE*CO_ICF as CO_winter, NOX_IDLE*NOX_ICF as NOX_winter, PM_IDLE*PM_ICF as PM_winter, " & _
"CO2_IDLE*CO2_ICF as CO2_winter FROM " & table & " e INNER JOIN idle_factor f " & _
"ON e.vehicle_class_Abb = f.vehicle_class_Abb " & _
"WHERE (((f.Season)='Winter'));"

sql_db.MySQL_Docmd(sql)

'Summer high idle

sql = "Create Table " & database_outputs & ".EmissionRatesHighIdle_summer ENGINE = MYISAM  as " & _
"SELECT f.Season, e.vehicle_class_Abb, " & _
"e.Model_Year, e.MY_Range, HC_IDLE*HC_ICF as HC_summer, " & _
"CO_IDLE*CO_ICF as CO_summer, NOX_IDLE*NOX_ICF as NOX_summer, PM_IDLE*PM_ICF as PM_summer, " & _
"CO2_IDLE*CO2_ICF as CO2_summer FROM " & table & " e INNER JOIN idle_factor f " & _
"ON e.vehicle_class_Abb = f.vehicle_class_Abb " & _
"WHERE (((f.Season)='Summer'));"

sql_db.MySQL_Docmd(sql)

'annual idle emissions rates
''a' as Season,

sql = "Create Table " & database_outputs & ".er_idle_1 ENGINE = MYISAM  as " & _
"SELECT e.vehicle_class_Abb, e.Fuel_Type, " & _
"e.Model_Year, e.MY_Range, trim('a') as season, " & _
"HC_IDLE*0.61+0.39*(0.58*HC_Summer+0.42*HC_winter) As HC, " & _
"CO_IDLE*0.61+0.39*(0.58*CO_Summer+0.42*CO_winter) as CO, " & _
"NOX_IDLE*0.61+0.39*(0.58*NOX_Summer+0.42*NOX_winter) as NOx, " & _
"PM_IDLE*0.61+0.39*(0.58*PM_Summer+0.42*PM_winter) as PM, " & _
"CO2_IDLE*0.61+0.39*(0.58*CO2_Summer+0.42*CO2_winter) as CO2 " & _
"FROM (" & table & " e INNER JOIN " & database_outputs & ".EmissionRatesHighIdle_summer s ON " & _
```

```
"(e.Model_Year = s.Model_Year) AND (e.vehicle_class_Abb = " & _
"s.vehicle_class_Abb)) INNER JOIN " & database_outputs & ".EmissionRatesHighIdle_winter w ON " & _
"(s.Model_Year = w.Model_Year) AND " & _
"(s.vehicle_class_Abb = w.vehicle_class_Abb);"

sql_db.MySQL_Docmd(sql)

'Append Summer idle emission rates

sql = "INSERT INTO " & database_outputs & ".er_idle_1 " & _
"( vehicle_class_Abb, Fuel_Type, Model_Year, MY_Range, season, HC, CO, NOx, PM, CO2 ) " & _
"SELECT e.vehicle_class_Abb, e.Fuel_Type, " & _
"e.Model_Year, e.MY_Range, trim('s') as season, " & _
"HC_IDLE*0.61+0.39*HC_Summer As HC, CO_IDLE*0.61+0.39*CO_Summer as CO, " & _
"NOX_IDLE*0.61+0.39*NOX_Summer as NOx, PM_IDLE*0.61+0.39*PM_Summer as PM, " & _
"CO2_IDLE*0.61+0.39*CO2_Summer as CO2 " & _
"FROM (" & table & " e INNER JOIN " & database_outputs & ".EmissionRatesHighIdle_summer s ON " & _
"(e.Model_Year = s.Model_Year) AND (e.vehicle_class_Abb = s.vehicle_class_Abb));"

sql_db.MySQL_Docmd(sql)

'Append winter idle emissions rates

sql = "INSERT INTO " & database_outputs & ".er_idle_1 " & _
"( vehicle_class_Abb, Fuel_Type, Model_Year, MY_Range, season, HC, CO, NOx, PM, CO2 ) " & _
"SELECT e.vehicle_class_Abb, e.Fuel_Type, " & _
"e.Model_Year, e.MY_Range, trim('w') as season, " & _
"HC_IDLE*0.61+0.39*HC_winter As HC, CO_IDLE*0.61+0.39*CO_winter as CO, " & _
"NOX_IDLE*0.61+0.39*NOX_winter as NOx, PM_IDLE*0.61+0.39*PM_winter as PM, " & _
"CO2_IDLE*0.61+0.39*CO2_winter as CO2 " & _
"FROM (" & table & " e INNER JOIN " & database_outputs & ".EmissionRatesHighIdle_winter w ON " & _
"(e.Model_Year = w.Model_Year) AND (e.vehicle_class_Abb = " & _
"w.vehicle_class_Abb));"

sql_db.MySQL_Docmd(sql)

'idle fuel correction

sql = "Create Table " & database_outputs & ".er_idle ENGINE = MYISAM as " & _
"SELECT f.calendar_year CY, g.gai, e.vehicle_class_Abb, e.Fuel_Type, e.Model_Year, e.MY_Range, " & _
```

```
        "e.season, HC*HC_fcf AS HC, CO*CO_fcf AS CO, " & _
        "NOX*NOX_fcf AS NOX, PM*PM_fcf AS PM, CO2*CO2_fcf AS CO2 " & _
        "FROM " & database_outputs & ".er_idle_1 e INNER JOIN (fuel_correction_factor f INNER JOIN geographic_regions g " & _
        "ON f.Area = g.fcf_area) ON (e.vehicle_class_abb = " & _
        "f.vehicle_class) AND (e.Model_Year = f.Model_Year)  " & _
        "Where f.calendar_year = " & cy & " and g.gai = " & gai & ";"

        sql_db.MySQL_Docmd(sql)

        'add index

        sql = "ALTER TABLE " & database_outputs & ".er_idle " & _
       "ADD INDEX `idel_index` (`vehicle_class_Abb` ASC, `Model_Year` ASC, `CY` ASC, `gai` ASC ) ;"

        sql_db.MySQL_Docmd(sql)

        'Drop temporary tables, if table exists, drop it

        sql = "DROP TABLE IF EXISTS " & database_outputs & ".EmissionRatesHighIdle_winter;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP TABLE IF EXISTS " & database_outputs & ".EmissionRatesHighIdle_summer;"
        sql_db.MySQL_Docmd(sql)

        sql = "DROP TABLE IF EXISTS " & database_outputs & ".er_idle_1;"
        sql_db.MySQL_Docmd(sql)

    End Sub

    Sub idle_ei(ByVal scenario As String, ByVal cy As Integer, ByVal gai As Integer)

        'calculate idle emissions

        Dim sql_db As New procedure_gui

        If sql_db.index_exist("year_range", "cy") = False Then
            sql = "ALTER TABLE `year_range` ADD INDEX `cy` (`Year` ASC) ;"
            sql_db.MySQL_Docmd(sql)
        End If
```

```
    Dim ei_table As String
    Dim ei_table_aps As String
    Dim er_table As String

    er_table = database_outputs & ".ei_idle_pop_er"

    'check whether table exists
    ei_table = database_outputs & ".ei_idle"
    ei_table_aps = database_outputs & ".ei_idle_aps"

    If sql_db.Table_exist(ei_table) = True Then
        sql = "Truncate table " & ei_table & ";"
        sql_db.MySQL_Docmd(sql)

    End If


    Dim i As Integer

    i = cy

    sql = "DROP Table IF EXISTS " & er_table & ";"
    sql_db.MySQL_Docmd(sql)


    'pm_control

    sql = "Create Table " & er_table & " ENGINE = MYISAM as " & _
    "SELECT v.CY, v.Fleet Veh2011, f.EMFAC2007_category Veh2007, v.gai, v.er_MY MdlYr, e.season, " & _
    "v.pop_ca, pm_control, " & _
    "e.HC*pm_control as HC, e.CO*pm_control as CO, e.NOx, e.pm *pm_control as pm, e.CO2 " & _
    "FROM vehicle_fleet_category f INNER JOIN (" & database_outputs & ".vmt_pop_" & scenario & "_gai v " & _
    "INNER JOIN " & database_outputs & ".er_idle e ON v.er_MY = e.Model_Year and v.CY = e.CY " & _
    " and v.GAI = e.GAI) ON (e.vehicle_class_Abb = f.Vehicle_class_ER) " & _
    "AND (f.Inventory_Category = v.Fleet);"

    sql_db.MySQL_Docmd(sql)

    '2% increase on CO2 with PM control
    sql = "UPDATE " & er_table & "  " & _
```

```
    "SET CO2= CO2*1.02 where pm_control = 0.15;"

    sql_db.MySQL_Docmd(sql)

    'add index

    sql = "ALTER TABLE " & er_table & " " & _
   "ADD INDEX `fleet` (`Veh2011` ASC) ;"

    sql_db.MySQL_Docmd(sql)

    sql = "DROP Table IF EXISTS " & ei_table & ";"
    sql_db.MySQL_Docmd(sql)

    'Add 2007 category

    sql = "Create Table " & ei_table & " ENGINE = MYISAM as  " & _
        "SELECT p.CY, Veh2011, Veh2007, p.gai, p.MdlYr, p.season, p.pop_ca, p.pop_ca*p.HC*h.hr/907184.74 HC,  " & _
        "p.pop_ca*p.CO*h.hr/907184.74 CO, p.pop_ca*p.NOx*h.hr/907184.74 NOx, " & _
        "p.pop_ca*p.pm*h.hr/907184.74 PM, p.pop_ca*p.CO2*h.hr/907184.74 CO2, p.CO2 as sox " & _
        "FROM (vehicle_fleet_category v INNER JOIN " & er_table & " p  " & _
        "ON v.Inventory_Category = p.Veh2011) INNER JOIN idle_hour h  " & _
        "ON v.inventory_category_idle = h.vehicle_class " & _
        "WHERE (((h.vehicle_class) <> 'T7')); "

    sql_db.MySQL_Docmd(sql)

    'Add index
    sql = "ALTER TABLE " & er_table & " " & _
    "ADD INDEX `idle_port` (`Veh2011` ASC, `CY` ASC) ;"

    sql_db.MySQL_Docmd(sql)

    'append T7 port truck        '

    sql = "INSERT INTO " & ei_table & " " & _
    "( CY, Veh2011, Veh2007, gai, MdlYr, season, pop_ca, HC, CO, NOx, PM, CO2, sox ) " & _
    "SELECT p.CY, p.Veh2011, 'HHDT' as Veh2007, p.gai, p.MdlYr, p.season, p.pop_ca, p.pop_ca*p.HC*h.hr/907184.74 HC,  " & _
    "p.pop_ca*p.CO*h.hr/907184.74 CO, p.pop_ca*p.NOx*h.hr/907184.74 NOx, " & _
    "p.pop_ca*p.pm*h.hr/907184.74 PM, p.pop_ca*p.CO2*h.hr/907184.74 CO2, p.CO2 as sox " & _
```

```
"FROM (" & er_table & " p INNER JOIN idle_time_port h ON " & _
"(p.Veh2011 = h.Inventory_Category)) INNER JOIN year_range y " & _
"ON (y.idle_port_cy = h.CY) AND (p.CY = y.Year);"

sql_db.MySQL_Docmd(sql)

'T7 average idling time

'add columns
sql = "ALTER TABLE " & er_table & " ADD COLUMN `fleet_idle` VARCHAR(45) " & _
"NULL  AFTER `CO2` , ADD COLUMN `my_idle` VARCHAR(45) NULL  AFTER `fleet_idle` , " & _
"ADD COLUMN `cy_idle` VARCHAR(45) NULL  AFTER `my_idle` ;"

sql_db.MySQL_Docmd(sql)

'add index

sql = "ALTER TABLE " & er_table & " " & _
"ADD INDEX `cy` (`CY` ASC) , ADD INDEX `my` (`MdlYr` ASC) ;"

sql_db.MySQL_Docmd(sql)

sql = "UPDATE " & er_table & " INNER JOIN vehicle_fleet_category ON " & _
"" & er_table & ".Veh2011 = vehicle_fleet_category.Inventory_Category " & _
"SET " & er_table & ".fleet_idle = vehicle_fleet_category.Inventory_Category_idle;"

sql_db.MySQL_Docmd(sql)

sql = "UPDATE " & er_table & " INNER JOIN year_range ON " & _
"" & er_table & ".CY = year_range.Year " & _
"SET " & er_table & ".cy_idle = year_range.idle_t7_cy;"

sql_db.MySQL_Docmd(sql)

sql = "UPDATE " & er_table & " INNER JOIN year_range ON " & _
"" & er_table & ".MdlYr = year_range.Year " & _
"SET " & er_table & ".my_idle = year_range.idle_t7_my;"

sql_db.MySQL_Docmd(sql)
```

```
    'Add index
    sql = "ALTER TABLE " & er_table & " " & _
    "ADD INDEX `idle_t7` (`fleet_idle` ASC, `CY_idle` ASC, `MY_idle` ASC) ;"
    sql_db.MySQL_Docmd(sql)

    'append

    sql = "INSERT INTO " & ei_table & " ( CY, Veh2011, Veh2007, gai, MdlYr, season, pop_ca, HC, " & _
        "CO, NOx, PM, CO2, sox ) " & _
    "SELECT p.CY, Veh2011, Veh2007, p.gai, p.MdlYr, p.season, p.pop_ca, p.pop_ca*p.HC*h.hr_avg/907184.74 HC,  " & _
    "p.pop_ca*p.CO*h.hr_avg/907184.74 CO, p.pop_ca*p.NOx*h.hr_avg/907184.74 NOx, " & _
    "p.pop_ca*p.pm*h.hr_avg/907184.74 PM, p.pop_ca*p.CO2*h.hr_avg/907184.74 CO2, p.CO2 as sox " & _
    "FROM " & er_table & " p INNER JOIN idle_t7_hr_avg h ON " & _
    "(p.cy_idle = h.Calendar_year_range) AND (p.my_idle = h.Model_year_range) " & _
    "AND (p.fleet_idle = h.Inventory_Category_idle);"

    sql_db.MySQL_Docmd(sql)

    'add SOx emissions

    sql = "select * from information_schema.columns " & _
   "where table_name = 'ei_idle' and " & _
   "column_name = 'sox' and table_schema = '" & database_outputs & "';"

    If sql_db.component_exist(sql) = False Then

        sql = "ALTER TABLE " & ei_table & " ADD COLUMN `sox` DOUBLE NULL  AFTER `CO2` ;"
        sql_db.MySQL_Docmd(sql)

    End If

    sql = "select * from information_schema.statistics " & _
"where table_name = 'ei_idle' and index_name = 'sox_index' " & _
"and table_schema = '" & database_outputs & "';"

    If sql_db.component_exist(sql) = False Then

        sql = "ALTER TABLE " & ei_table & " ADD INDEX `sox_index` (`CY` ASC, `gai` ASC) ;"
        sql_db.MySQL_Docmd(sql)
```

```
    End If

    sql = "select * from information_schema.statistics " & _
"where table_name = 'sulfur_content' and index_name = 'index' " & _
"and table_schema = '" & database_inputs & "';"

    If sql_db.index_exist("sulfur_content", "index") = False Then

        sql = "ALTER TABLE `sulfur_content` ADD INDEX `index` (`CY_sulfur` ASC, `fcf_area` ASC) ;"
        sql_db.MySQL_Docmd(sql)

    End If

    'Update sox
    sql = "UPDATE ((sulfur_content s INNER JOIN geographic_regions g ON " & _
    "s.fcf_area = g.FCF_area) INNER JOIN " & ei_table & " e ON g.gai = e.gai) INNER JOIN year_range ON " & _
    "(year_range.Year = e.CY) AND (s.CY_sulfur = year_range.cy_sulfur) " & _
    "SET e.sox = e.CO2/3.1445*s.sulfur_ppm/1000000*2;"

    sql_db.MySQL_Docmd(sql)

    '*************************************
    'aps emissions

    If i >= 2008 Then    'calculate aps emissions only when calculate annual

        'add pop_sleeper column

        sql = "ALTER TABLE " & er_table & " ADD COLUMN `sleeper_pop` " & _
        "DOUBLE NULL  AFTER `cy_idle` ;"

        sql_db.MySQL_Docmd(sql)

        'add index to update sleeper pop

        sql = "ALTER TABLE " & er_table & " " & _
        "ADD INDEX `aps` (`fleet_idle` ASC, `cy_idle` ASC) ;"

        sql_db.MySQL_Docmd(sql)
```

```
'update sleeper pop
'Assume a sleeper does not need an APS before 2008

sql = "UPDATE " & er_table & " p INNER JOIN idle_t7_sleeper s ON " & _
"(p.cy_idle = s.Calendar_year_range) AND (p.fleet_idle = s.Inventory_Category_idle) " & _
"SET p.sleeper_pop = p.pop_ca*s.sleeper*s.sleeper_compliance where cy_idle = '2008+';"

sql_db.MySQL_Docmd(sql)

'add aps my

sql = "ALTER TABLE " & er_table & " ADD COLUMN `my_aps` " & _
"VARCHAR(45) NULL  AFTER `sleeper_pop` ;"

sql_db.MySQL_Docmd(sql)

'update my_aps

sql = "UPDATE " & er_table & " p INNER JOIN year_range y ON " & _
"p.MdlYr = y.Year " & _
"SET p.my_aps = y.idle_aps_my WHERE p.sleeper_pop >0;;"

sql_db.MySQL_Docmd(sql)

'aps emissions
'6 hrs per day

sql = "DROP Table IF EXISTS " & ei_table_aps & ";"
sql_db.MySQL_Docmd(sql)

sql = "Create Table " & ei_table_aps & " ENGINE = MYISAM as  " & _
"SELECT p.CY, Veh2011, Veh2007, p.fleet_idle, p.gai, p.MdlYr, p.season, p.pop_ca, p.sleeper_pop, " & _
"pm_control, p.sleeper_pop*e.HC*6/907184.74*pm_control HC,  " & _
"p.sleeper_pop*e.CO*6/907184.74*pm_control CO, p.sleeper_pop*e.NOx*6/907184.74 NOx, " & _
"p.sleeper_pop*e.pm*6/907184.74*pm_control PM, p.sleeper_pop*e.CO2*6/907184.74 CO2, e.CO2 as sox " & _
"FROM " & er_table & " p INNER JOIN idle_aps_emission_rates e ON p.my_aps = e.MY " & _
"WHERE p.sleeper_pop >0 and my_idle = 'pre 2008';"

sql_db.MySQL_Docmd(sql)
```

```
'2% increase on CO2 with PM control
sql = "UPDATE " & ei_table_aps & "  " & _
"SET CO2= CO2*1.02 where pm_control = 0.15 and cy = " & i & ";"

sql_db.MySQL_Docmd(sql)

'Aps SOx emissions
ei_table_aps = database_outputs & ".ei_idle_aps"

'add SOx emissions

sql = "select * from information_schema.columns " & _
"where table_name = 'ei_idle_aps' and " & _
"column_name = 'sox' and table_schema = '" & database_outputs & "';"

If sql_db.component_exist(sql) = False Then

    sql = "ALTER TABLE " & ei_table_aps & " ADD COLUMN `sox` DOUBLE NULL  AFTER `CO2` ;"
    sql_db.MySQL_Docmd(sql)

End If

sql = "select * from information_schema.statistics " & _
"where table_name = 'ei_idle_aps' and index_name = 'sox_index' " & _
"and table_schema = '" & database_outputs & "';"

If sql_db.component_exist(sql) = False Then

    sql = "ALTER TABLE " & ei_table_aps & " ADD INDEX `sox_index` (`CY` ASC, `gai` ASC) ;"
    sql_db.MySQL_Docmd(sql)

End If

sql = "select * from information_schema.statistics " & _
"where table_name = 'sulfur_content' and index_name = 'index' " & _
"and table_schema = '" & database_inputs & "';"

If sql_db.component_exist(sql) = False Then
```

```
            sql = "ALTER TABLE `sulfur_content` ADD INDEX `index` (`CY_sulfur` ASC, `fcf_area` ASC) ;"
            sql_db.MySQL_Docmd(sql)

        End If

        'Update sox
        sql = "UPDATE ((sulfur_content s INNER JOIN geographic_regions g ON " & _
        "s.fcf_area = g.FCF_area) INNER JOIN " & ei_table_aps & " e ON g.gai = e.gai) INNER JOIN year_range ON " & _
        "(year_range.Year = e.CY) AND (s.CY_sulfur = year_range.cy_sulfur) " & _
        "SET e.sox = e.CO2/3.1445*s.sulfur_ppm/1000000*2;"

        sql_db.MySQL_Docmd(sql)

        'Insert Aps into idle table

        sql = "INSERT INTO " & ei_table & " ( CY, Veh2011, Veh2007, gai, MdlYr, season, pop_ca, " & _
            "HC, CO, NOx, PM, CO2, sox ) " & _
            "SELECT CY, Veh2011, Veh2007, gai, MdlYr, season, 0 as pop_ca, HC, CO, NOx, PM, CO2, sox " & _
            "from " & ei_table_aps & ";"

        sql_db.MySQL_Docmd(sql)


    End If  'APS offset emissions

    'Summarize idle and aps emissions

    Dim by_process As String = ""
    Dim by_my As String = ""
    Dim group_by_my As String = ""

    If ei_by_process = True Then
        by_process = "'IDLEX'"
    Else
        by_process = "'All'"
    End If

    If ei_by_my = True Then
        by_my = "e.MdlYr"
        group_by_my = ", e.MdlYr"
```

```
        Else
            by_my = "'All' as MdlYr"
            group_by_my = ""
        End If

        tablename = database_outputs & ".emfac_hd_ei_output_summary"

        'Append to export table

        sql = "Insert into " & tablename & " (Process, Area, GAI, CalYr, Season, Fuel, Veh, MdlYr, " & _
        "pop_CA, VMT, NOx, PM10, PM2_5, TOG, ROG, CO, CO2, SOx, Fuel_DSL) " & _
       "SELECT " & by_process & " as Process, g.area, e.gai, e.CY, e.season, 'DSL' as fuel, " & _
       "" & emfac_cat & " as Veh, " & by_my & ", sum(e.pop_ca) pop_CA, 0 as vmt, " & _
       "sum(e.NOx) NOx, sum(e.PM) PM10, sum(e.PM)*0.92 PM2_5, " & _
       "sum(e.HC)*1.4417 TOG, sum(e.HC)*1.2664 ROG, sum(e.CO) CO, sum(e.CO2) CO2, sum(e.SOx) SOx, " & _
       "sum(e.CO2)*0.09 Fuel_DSL from " & ei_table & " e INNER JOIN geographic_regions g ON e.gai = g.gai " & _
       "where" & season_sql_annual & season_sql_summer & season_sql_winter & " " & _
       "group by g.area, e.gai, e.cy, e.season, fuel, Veh" & group_by_my & ";"

        sql_db.MySQL_Docmd(sql)

    End Sub

    Sub brake_wear_ei(ByVal scenario As String, ByVal cy As Integer, ByVal gai As Integer)

        'Procedure to calculate brake wear emissions

        Dim sql_db As New procedure_gui
        Dim vmt_table, ei_table As String

        vmt_table = database_outputs & ".vmt_pop_" & scenario & "_gai"

        sql = "select * from information_schema.statistics " & _
        "where table_name = 'vmt_pop_" & scenario & "_gai' and index_name = 'fleet' " & _
        "and table_schema = '" & database_outputs & "';"

        If sql_db.component_exist(sql) = False Then
            sql = "ALTER TABLE " & vmt_table & " ADD INDEX `fleet` (`fleet` ASC) ;"
            sql_db.MySQL_Docmd(sql)
        End If
```

```
'Brakewear emissions

ei_table = database_outputs & ".ei_bw"

sql = "DROP table IF EXISTS " & ei_table & ";"
sql_db.MySQL_Docmd(sql)

sql = "Create Table " & ei_table & " ENGINE = MYISAM  as  " & _
"SELECT v.CY, v.Fleet Veh2011, f.EMFAC2007_category veh2007, v.gai, " & _
"v.er_my MdlYr, sum(vmt) vmt, sum(pop_ca) pop_ca, " & _
"sum(v.vmt*e.PM/907184.74) as PM, sum(v.vmt*e.PM/907184.74)*" & bwpm10ratio & " PM10, " & _
"sum(v.vmt*e.PM/907184.74)*" & bwpm25ratio & " PM2_5 " & _
"FROM emission_rates_brakewear e INNER JOIN (" & vmt_table & " v " & _
"INNER JOIN vehicle_fleet_category f ON v.Fleet = f.Inventory_Category) " & _
"ON e.Vehicle_Class = f.Vehicle_class_brakewear " & _
"Where v.cy = " & cy & " and v.gai = " & gai & " " & _
"group by cy, Veh2011, veh2007, gai, v.er_my;"

sql_db.MySQL_Docmd(sql)

sql = "select * from information_schema.statistics " & _
"where table_name = 'ei_bw' and index_name = 'fleet' " & _
"and table_schema = '" & database_outputs & "';"

If sql_db.component_exist(sql) = False Then
    sql = "ALTER TABLE " & ei_table & " ADD INDEX `fleet` (`Veh2011` ASC) ;"
    sql_db.MySQL_Docmd(sql)
End If

sql = "select * from information_schema.statistics " & _
"where table_name = 'operation_day' and index_name = 'fleet' " & _
"and table_schema = '" & database_inputs & "';"

If sql_db.component_exist(sql) = False Then

    sql = "ALTER TABLE operation_day ADD INDEX `fleet` (`Fleet` ASC) ;"
    sql_db.MySQL_Docmd(sql)
End If
```

```
sql = "UPDATE " & ei_table & " e INNER JOIN " & _
"operation_day O ON e.Veh2011 = O.fleet SET vmt = vmt/operation_day, " & _
"PM = PM/operation_day, PM10 = PM10/operation_day, PM2_5 = PM2_5/operation_day;"

sql_db.MySQL_Docmd(sql)

'Insert Emissions to summary table

Dim by_process As String = ""
Dim by_my As String = ""
Dim group_by_my As String = ""

If ei_by_process = True Then
    by_process = "'PMBW'"
Else
    by_process = "'All'"
End If

If ei_by_my = True Then
    by_my = "e.MdlYr"
    group_by_my = ", e.MdlYr"
Else
    by_my = "'All' as MdlYr"
    group_by_my = ""
End If

'Table to hold emissions estimation for all processes by CY by GAI
tablename = "emfac_hd_ei_output_summary"

'Annual

If season_sql_annual <> "" Then

    sql = "Insert into " & database_outputs & "." & tablename & " (Process, Area, GAI, CalYr, Season, Fuel, Veh,
    MdlYr, " & _
    "pop_ca, VMT, NOx, PM10, PM2_5, TOG, ROG, CO, CO2, SOx) " & _
   "SELECT " & by_process & " as Process, g.area, e.gai, e.CY, 'a' as season, 'DSL' as fuel, " & _
   "" & emfac_cat & " as Veh, " & by_my & ", sum(pop_ca) pop_ca, sum(vmt) as vmt, " & _
   "0 NOx, sum(e.PM10) PM10, sum(e.PM2_5) PM2_5, " & _
   "0 TOG, 0 ROG, 0 CO, 0 CO2, 0 SOx " & _
```

```
    "from " & ei_table & " e INNER JOIN geographic_regions g ON e.gai = g.gai " & _
    "group by g.area, e.gai, e.cy, season, fuel, Veh" & group_by_my & ";"

     sql_db.MySQL_Docmd(sql)

End If

'winter

If season_sql_winter <> "" Then

     sql = "Insert into " & database_outputs & "." & tablename & " (Process, Area, GAI, CalYr, Season, Fuel, Veh,
     MdlYr, " & _
     "pop_ca, VMT, NOx, PM10, PM2_5, TOG, ROG, CO, CO2, SOx) " & _
    "SELECT " & by_process & " as Process, g.area, e.gai, e.CY, 'w' as season, 'DSL' as fuel, " & _
    "" & emfac_cat & " as Veh, " & by_my & ", sum(pop_ca) pop_ca, sum(vmt) as vmt, " & _
    "0 NOx, sum(e.PM10) PM10, sum(e.PM2_5) PM2_5, " & _
    "0 TOG, 0 ROG, 0 CO, 0 CO2, 0 SOx " & _
    "from " & ei_table & " e INNER JOIN geographic_regions g ON e.gai = g.gai " & _
    "group by g.area, e.gai, e.cy, season, fuel, Veh" & group_by_my & ";"

     sql_db.MySQL_Docmd(sql)

End If

'summer

If season_sql_summer <> "" Then

     sql = "Insert into " & database_outputs & "." & tablename & " (Process, Area, GAI, CalYr, Season, Fuel, Veh,
     MdlYr, " & _
     "pop_ca, VMT, NOx, PM10, PM2_5, TOG, ROG, CO, CO2, SOx) " & _
    "SELECT " & by_process & " as Process, g.area, e.gai, e.CY, 's' as season, 'DSL' as fuel, " & _
    "" & emfac_cat & " as Veh, " & by_my & ", sum(pop_ca) pop_ca, sum(vmt) as vmt, " & _
    "0 NOx, sum(e.PM10) PM10, sum(e.PM2_5) PM2_5, " & _
    "0 TOG, 0 ROG, 0 CO, 0 CO2, 0 SOx " & _
    "from " & ei_table & " e INNER JOIN geographic_regions g ON e.gai = g.gai " & _
    "group by g.area, e.gai, e.cy, season, fuel, Veh" & group_by_my & ";"

     sql_db.MySQL_Docmd(sql)
```

```
    End If

End Sub

Sub tire_wear_ei(ByVal scenario As String, ByVal cy As Integer, ByVal gai As Integer)

    'Procedure to calculate tire wear emissions

    Dim sql_db As New procedure_gui
    Dim vmt_table, ei_table As String

    vmt_table = database_outputs & ".vmt_pop_" & scenario & "_gai"

    'Add fleet index, and it will be much faster

    sql = "select * from information_schema.statistics " & _
    "where table_name = 'vmt_pop_" & scenario & "_gai' and index_name = 'fleet' " & _
    "and table_schema = '" & database_outputs & "';"

    If sql_db.component_exist(sql) = False Then
        sql = "ALTER TABLE " & vmt_table & " ADD INDEX `fleet` (`fleet` ASC) ;"
        sql_db.MySQL_Docmd(sql)
    End If

    'Tirewear emissions

    ei_table = database_outputs & ".ei_tw"

    sql = "DROP table IF EXISTS " & ei_table & ";"
    sql_db.MySQL_Docmd(sql)


    sql = "Create Table " & ei_table & " ENGINE = MYISAM  as  " & _
    "SELECT v.CY, v.Fleet Veh2011, f.EMFAC2007_category veh2007, v.gai, v.er_my MdlYr, " & _
    "sum(vmt) vmt, sum(pop_ca) pop_ca, " & _
    "sum(v.vmt*e.PM/907184.74) as PM, sum(v.vmt*e.PM/907184.74)*" & twpm10ratio & " PM10, " & _
    "sum(v.vmt*e.PM/907184.74)*" & twpm25ratio & " PM2_5 " & _
    "FROM emission_rates_tirewear e INNER JOIN (" & vmt_table & " v " & _
    "INNER JOIN vehicle_fleet_category f ON v.Fleet = f.Inventory_Category) " & _
```

```
"ON e.Vehicle_Class = f.Vehicle_class_tirewear " & _
"Where v.cy = " & cy & " and v.gai = " & gai & " " & _
"group by cy, Veh2011, veh2007, gai, v.er_my;"

sql_db.MySQL_Docmd(sql)

sql = "select * from information_schema.statistics " & _
"where table_name = 'ei_tw' and index_name = 'fleet' " & _
"and table_schema = '" & database_outputs & "';"

If sql_db.component_exist(sql) = False Then
    sql = "ALTER TABLE " & ei_table & " ADD INDEX `fleet` (`Veh2011` ASC) ;"
    sql_db.MySQL_Docmd(sql)
End If

sql = "select * from information_schema.statistics " & _
"where table_name = 'operation_day' and index_name = 'fleet' " & _
"and table_schema = '" & database_inputs & "';"

If sql_db.component_exist(sql) = False Then

    sql = "ALTER TABLE operation_day ADD INDEX `fleet` (`Fleet` ASC) ;"
    sql_db.MySQL_Docmd(sql)
End If

'convert to tons per day

sql = "UPDATE " & ei_table & " e INNER JOIN " & _
"operation_day O ON e.Veh2011 = O.fleet SET vmt = vmt/operation_day, " & _
"PM = PM/operation_day, PM10 = PM10/operation_day, PM2_5 = PM2_5/operation_day;"

sql_db.MySQL_Docmd(sql)

'Insert Emissions to summary table

Dim by_process As String = ""
Dim by_my As String = ""
Dim group_by_my As String = ""

If ei_by_process = True Then
```

```
        by_process = "'PMTW'"
    Else
        by_process = "'All'"
    End If

    If ei_by_my = True Then
        by_my = "e.MdlYr"
        group_by_my = ", e.MdlYr"
    Else
        by_my = "'All' as MdlYr"
        group_by_my = ""
    End If

    'Table to hold emissions estimation for all processes by CY by GAI
    tablename = "emfac_hd_ei_output_summary"

    'Annual

    If season_sql_annual <> "" Then

        sql = "Insert into " & database_outputs & "." & tablename & " (Process, Area, GAI, CalYr, Season, Fuel, Veh,
        MdlYr, " & _
        "pop_ca, VMT, NOx, PM10, PM2_5, TOG, ROG, CO, CO2, SOx) " & _
       "SELECT " & by_process & " as Process, g.area, e.gai, e.CY, 'a' as season, 'DSL' as fuel, " & _
       "" & emfac_cat & " as Veh, " & by_my & ", sum(pop_ca) pop_ca, sum(vmt) as vmt, " & _
       "0 NOx, sum(e.PM10) PM10, sum(e.PM2_5) PM2_5, " & _
       "0 TOG, 0 ROG, 0 CO, 0 CO2, 0 SOx " & _
       "from " & ei_table & " e INNER JOIN geographic_regions g ON e.gai = g.gai " & _
       "group by g.area, e.gai, e.cy, season, fuel, Veh" & group_by_my & ";"

        sql_db.MySQL_Docmd(sql)

    End If

    'winter

    If season_sql_winter <> "" Then

        sql = "Insert into " & database_outputs & "." & tablename & " (Process, Area, GAI, CalYr, Season, Fuel, Veh,
        MdlYr, " & _
```

```
            "pop_ca, VMT, NOx, PM10, PM2_5, TOG, ROG, CO, CO2, SOx) " & _
            "SELECT " & by_process & " as Process, g.area, e.gai, e.CY, 'w' as season, 'DSL' as fuel, " & _
            "" & emfac_cat & " as Veh, " & by_my & ", sum(pop_ca) pop_ca, sum(vmt) as vmt, " & _
            "0 NOx, sum(e.PM10) PM10, sum(e.PM2_5) PM2_5, " & _
            "0 TOG, 0 ROG, 0 CO, 0 CO2, 0 SOx " & _
            "from " & ei_table & " e INNER JOIN geographic_regions g ON e.gai = g.gai " & _
            "group by g.area, e.gai, e.cy, season, fuel, Veh" & group_by_my & ";"

            sql_db.MySQL_Docmd(sql)

        End If

        'summer

        If season_sql_summer <> "" Then

            sql = "Insert into " & database_outputs & "." & tablename & " (Process, Area, GAI, CalYr, Season, Fuel, Veh,
            MdlYr, " & _
            "pop_ca, VMT, NOx, PM10, PM2_5, TOG, ROG, CO, CO2, SOx) " & _
            "SELECT " & by_process & " as Process, g.area, e.gai, e.CY, 's' as season, 'DSL' as fuel, " & _
            "" & emfac_cat & " as Veh, " & by_my & ", sum(pop_ca) pop_ca, sum(vmt) as vmt, " & _
            "0 NOx, sum(e.PM10) PM10, sum(e.PM2_5) PM2_5, " & _
            "0 TOG, 0 ROG, 0 CO, 0 CO2, 0 SOx " & _
            "from " & ei_table & " e INNER JOIN geographic_regions g ON e.gai = g.gai " & _
            "group by g.area, e.gai, e.cy, season, fuel, Veh" & group_by_my & ";"

            sql_db.MySQL_Docmd(sql)

        End If

    End Sub

    Sub ei_sum_export_gai_cy(ByVal cy As Integer, ByVal gai As Integer)

        Dim tablename_input As String = "emfac_hd_ei_output_summary"
        Dim ei_ex As String = "emfac_hd_ei_output_summary_export"

        sql = "DROP Table IF EXISTS " & database_outputs & "." & ei_ex & ";"
        sql_db.MySQL_Docmd(sql)
```

```
    Dim pop_divider, vmt_divider As Integer

    If ei_by_process = True Then
        pop_divider = 1
        vmt_divider = 1

    Else
        pop_divider = 4
        vmt_divider = 3

    End If

    'pop/4 because it was added 4 times, vmt/3 added 3 times

    sql = "Create Table " & database_outputs & "." & ei_ex & " ENGINE = MYISAM as " & _
        "SELECT Process, Area, GAI, CalYr, Season, Veh, Fuel, MdlYr, 'All' Speed_bin, " & _
        "sum(pop_ca)/" & pop_divider & " as pop_ca, sum(e.vmt)/" & vmt_divider & " as vmt, " & _
        "sum(e.NOx) NOx, sum(e.PM10) PM10, sum(e.PM2_5) PM2_5, " & _
        "sum(e.TOG) TOG, sum(e.ROG) ROG, sum(e.CO) CO, sum(e.CO2) CO2, sum(e.SOx) SOx, " & _
        "sum(Fuel_DSL) Fuel_DSL from " & database_outputs & "." & tablename_input & " e " & _
        "group by Process, Area, GAI, CalYr, Season, Fuel, Veh, MdlYr;"

    sql_db.MySQL_Docmd(sql)

    If save2server = True Then

        tablename = database_outputs & ".emfac_hd_ei_output_summary_gai_" & gai.ToString

        sql = "Insert into " & tablename & "  " & _
        "SELECT * from " & database_outputs & "." & ei_ex & " ;"

        sql_db.MySQL_Docmd(sql)

    End If

    'export running emissions by speed file

    If save2folder = True Then

        'Export table with header name
```

```
            Dim filename As String = "emfac_hd_output_gai_" & gai.ToString & "_cy_" & cy.ToString & "_summary.csv"
            filename = savepath & "\" & filename
            filename = Replace(filename, "\", "\\")

            sql = "SELECT 'Process', 'Area', 'GAI', 'CalYr', 'Season', 'Veh', 'Fuel', " & _
           "'MdlYr', 'Speed_bin', 'Pop_CA', 'VMT', 'NOx', 'PM10', 'PM2_5', 'TOG', 'ROG', 'CO', 'CO2', 'SOx', 'Fuel_DSL' " & _
          "UNION SELECT * INTO OUTFILE '" & filename & "' " & _
          "FIELDS TERMINATED BY ',' FROM " & database_outputs & "." & ei_ex & ";"

            sql_db.MySQL_Docmd(sql)

        End If    'whether to save file to folder


    End Sub


End Class
```