Work Package 5 EU-CEG data and enhanced laboratory capacity for regulatory purposes

Regulatory compliance and discrepancies in EU-CEG data

Authors: ANSES
September 2024

Doc. Ref. N°: D5.8
Type: Document (R)
Dissemination: Public (P)

JATC 2 JOINT ACTION ON TOBACCO CONTROL

Co-funded by the European Union's Health Programme under Grant Agreement n°: 101035968 - JA-01-2020 - HP-JA-2020 / HP-JA-2020-2

Version	Date	Authors	Comments
1	30 Sept. 2024	ANSES	First version

Table of contents

Introduction

Manufacturers and importers of tobacco products are required to submit key information about the products they intend to market to the competent authorities of the EU Member States.

Manufacturers' declarations may contain errors, discrepancies, or even regulatory non-compliances with respect to the obligations of the TPD.

In this report, we propose criteria for automatically detecting discrepancies and non-compliances in the declarations.

These criteria were applied to a set of public data shared by 12 European Member States.

1 Data sources

The joint dataset was compiled from the publication files shared through MS-Rep according to the process developed in this project (see JATC2 D5.1).

The competent authorities of 12 Member States shared their public data in June 2023. These were the following countries: Belgium (BE), Bulgaria (BG), Czech Republic (CZ), Estonia (EE), France (FR), Italy (IT), Lithuania (LT), Luxembourg (LU), Latvia (LV), Netherlands (NL), Sweden (SE), and Slovenia (SI).

Each national dataset contained two files for both e-cigarette and tobacco products: one CSV file listing the products (and their presentations) and one CSV file containing composition data (list of ingredients and their quantities).

2 Criteria

Based on an analysis of the TPD provisions cross-referenced with the available data in the respective declarative models for electronic cigarettes and tobacco products, criteria were defined to identify these errors in the declarations.

For electronic cigarettes, we defined 4 inconsistencies and 12 non-compliances.

Table. Non-compliance errors and discrepancies within e-cigarette notifications.

	Error Code	Product Type	Error Name	Legal Basis	Test
Discrepancy	EC-IN-01	E-liquid	Discrepancy in mass composition	-	the sum of ingredient quantities and product weight differ
	EC-IN-02	E-liquid	Discrepancy in e-liquid density	-	density (product weight / volume) oustside [1.1 ; 1.3]
	EC-IN-03	E-liquid	Discrepancy in declared ingredient composition	-	at least one ingredient is not properly identified (CAS, name)
	EC-IN-04	E-liquid w/ nicotine	Discrepancy in nicotine level	-	nicotine ingredient quantity and product nicotine strength differ more than ± 5%
Non-compliance	EC-NC-01	E-cigarette	Non-compliant 6 months notification delay	TPD 20.2	first launch date is less than 6 months from first submission date
	EC-NC-02	E-liquid	Non-compliant composition - Presence of CMR substances	TPD 20.2	at least one ingredient is classified as CMR (1272/2008/EC - CLP Regulation)
	EC-NC-03	E-liquid	Non-compliant composition - Presence of prohibited additives	TPD20.3.c	at least one ingredient is a stimulant or a vitamin (substances negative list)
	EC-NC-04	E-cigarette	Non-compliant sales volumes – No data	TPD 20.7	no sales data provided year N+1 while product was placed on the market year N
	EC-NC-06	E-liquid	Non-compliant nicotine level	TPD 20.3.b	nicotine > 20mg/mL
	EC-NC-07	E-liquid	Non-compliant e-liquid volume	TPD 20.3.a	volume > 10mL (containers) or > 2mL (cartridges)
	EC-NC-08	E-cigarette	Production Conformity not checked	TPD 20.2.f	ProductConformity is not TRUE
	EC-NC-09	E-cigarette	Quality Safety not checked	TPD 20.2.g	QualitySafety is not TRUE
	EC-NC-10	E-cigarette	Child Tamper Proof not checked	TPD 20.3.g	ChildTamperProof is not TRUE
	EC-NC-11	E-liquid	High Purity not checked	TPD 20.3.d	HighPurity is not TRUE
	EC-NC-12	E-cigarette	Non Risk not checked	TPD 20.3.e	NonRisk is not TRUE
	EC-NC-13	E-cigarette	Consistent Dosing not checked	TPD 20.3.f	ConsistentDosing is not TRUE

The inconsistencies concern the quantities of ingredients relative to the total mass of the e-liquid, the density of the e-liquid calculated from its mass and volume, the consistency of the nicotine content with the added ingredient, and the presence of ingredients that are not identified by a clear name or CAS number.

The non-compliances are systematically related to TPD provisions. They include the presence of prohibited ingredients, nicotine concentration or e-liquid volume exceeding the maximum limits, missing sales data, or the absence of the manufacturer's attestation for a number of criteria such as product compliance, quality and safety, ingredient purity, etc. These criteria are indicated through

checkboxes (or a TRUE value) in the manufacturer's declaration.

For tobacco products, we defined 3 inconsistencies and 14 non-compliances.

The inconsistencies concern the quantities of ingredients relative to the total mass of the product or the tobacco, and the presence of ingredients not identified by a clear name or CAS number.

The non-compliances are systematically related to TPD provisions. These include the presence of prohibited ingredients, TNCO values exceeding the maximum limits or missing (for cigarettes), non-compliant total weights (for cigars, cigarillos, and rolling tobacco), among others.

Table. Non-compliance errors and discrepancies within tobacco declarations.

	Error Code	Product Type	Error Name	Legal Basis	Test
Discrepancy	TP-IN-01	Tobacco product	Discrepancy in mass composition - Tobacco weight	-	the sum of tobacco ingredient quantities and tobacco weight differ
	TP-IN-02	Tobacco product	Discrepancy in mass composition - Total weight	-	the sum of all ingredient quantities and tobacco weight differ
	TP-IN-03	Tobacco product	Discrepancy in declared ingredient composition	-	at least one ingredient is not properly identified (CAS, name)
Non-compliance	TP-NC-01	Novel tobacco product	Non-compliant 6 months notification delay	TPD 19.1	first launch date is less than 6 months from first submission date
	TP-NC-02	Tobacco product	Non-compliant composition - Presence of CMR substances	TPD 7.6.e	at least one ingredient is classified as CMR (1272/2008/EC - CLP Regulation)
	TP-NC-03	Tobacco product	Non-compliant composition - Presence of prohibited additives	TPD 7.6.a-d	at least one ingredient is a stimulant or a vitamin (substances negative list)
	TP-NC-04	Tobacco product	Non-compliant sales volumes – No data	TPD 5.6	no sales data provided year N+1 while product was placed on the market year N
	TP-NC-05	Cigarette	Non-compliant TNCO value - Tar	TPD 3.1.a	Tar > 10
	TP-NC-06	Cigarette	Non-compliant TNCO value - Nicotine	TPD 3.1.b	Nicotine > 1
	TP-NC-07	Cigarette	Non-compliant TNCO value - Co	TPD 3.1.c	CO > 10
	TP-NC-08	Cigarette	Non-compliant TNCO value - No data	TPD 3.1	at least one TNCO value is missing
	TP-NC-09	Cigarillo	Non-compliant cigarillo weight	2007/74/CE Art. 8	product weight > 3g
	TP-NC-10	Cigar	Non-compliant cigar weight	2007/74/CE Art. 8	product weight ≤ 3g
	TP-NC-11	Cigarette	Non-compliant cigarette units per packaging	TPD 14.1	units < 20 or not a multiple of 5
	TP-NC-12	Roll your own tobacco	Non-compliant rolling tobacco weight per packaging	TPD 14.1	package net weight < 30g
	TP-NC-13	Oral tobacco	Tobacco for oral use is prohibited in most EEA countries	TPD 17	oral tobacco product type selected (except for Austria, Finland and Sweden)
	TP-NC-14	Tobacco product	Flavor added to non-tobacco part	TPD 7.7	ingredient (flavor) with function in (12;22) added to a non-tobacco ingredient categ

To test these criteria on a concrete use case, we applied those that were feasible to the public EU-CEG dataset shared by 12 Member States as part of this JATC-2 project.

The CSV files were imported into an SQLite database. They underwent a curation process with the addition of information concerning generic ingredients according to a method developed in this project (see JATC2 D5.9).

Eventually, the criteria for discrepancies and regulatory non-compliance were translated into SQL queries to generate the results for both e-cigarette and tobacco products.

Due to the specific nature ou EU-CEG public datasets, it was not possible to test some criteria which need the whole non-confidential dataset: this particularly the case for sales volumes data, manufacturer's attestation or notification delays. Nevertheless, those criteria as easy to implement on a complete national dataset.

3 E-cigarette

The following table shows the non-compliances detected in e-cigarette declarations by type and by Member State. They are ranked from the most frequent to the least frequent.

In general, the non-compliance rates are extremely low, typically affecting less than 1% of products. A non-compliant e-liquid volume is the most commonly encountered issue, followed by the presence of ingredients with CMR properties, then non-compliant nicotine levels, and finally the presence of prohibited ingredients such as vitamins or caffeine.

Table. Non-compliance errors from the e-cigarette EU dataset.

Non-compliance	National Market	Error Rate
Non-compliant e-liquid volume	CZ	3.009%
	LU	0.431%
	FR	0.321%
	BE	0.220%
	IT	0.141%
	NL	0.074%
	EE	0.009%
Non-compliant composition - Presence of CMR substances	CZ	1.011%
	BE	0.595%
	EE	0.398%
	NL	0.376%
	FR	0.275%
	IT	0.148%
	SE	0.115%
	LV	0.095%
	SI	0.018%
Non-compliant nicotine level	BE	0.212%
	CZ	0.078%
	FR	0.050%
	IT	0.023%
	SE	0.016%
	NL	0.009%
Non-compliant composition - Presence of prohibited additives	CZ	0.030%
	FR	0.013%
	IT	0.005%

In contrast, declarative inconsistencies occur at higher frequencies. While it is not surprising to see inconsistencies related to ingredient quantities (since those representing less than 0.1% are not included in public files), the inconsistency in nicotine content is significant, particularly for certain countries. A more detailed analysis is needed to determine the cause of these high frequencies (e.g., use of nicotine salts, unit errors).

Overall, it is reassuring that the majority of ingredients are correctly identified (again, focusing only on those declared at more than 0.1%).

Table. Discrepancies from the e-cigarette EU dataset.

Discrepancy	National Market	Error Rate
Discrepancy in nicotine level	BE	41.554%
	CZ	30.094%
	FR	12.485%
	NL	4.714%
	SE	3.141%
	IT	3.084%
	EE	2.285%
	LV	1.493%
	LT	1.480%
	LU	0.588%
	SI	0.294%
Discrepancy in mass composition	BE	18.620%
	CZ	16.899%
	FR	7.587%
	NL	4.900%
	EE	2.808%
	IT	2.452%
	SE	1.963%
	LU	1.278%
	LV	0.768%
	SI	0.240%
Discrepancy in e-liquid density	BE	16.939%
	CZ	14.654%
	FR	4.127%
	IT	3.389%
	NL	2.055%
	SE	2.050%
	EE	1.353%
	LV	0.604%
	LT	0.444%
	LU	0.339%
	SI	0.215%
Discrepancy in declared ingredient composition	NL	2.500%
	SE	0.053%
	BE	0.032%
	CZ	0.019%
	FR	0.002%

4 Tobacco products

For tobacco products, we calculated the non-compliance rates by nature, by Member State, and by product type.

Table. Non-compliance errors from the tobacco products EU dataset.

Non-compliance	National Market	Product Type	Error Rate
Non-compliant rolling tobacco weight per packaging	BE	Roll your own tobacco	13.1%
	NL	Roll your own tobacco	9.2%
	CZ	Roll your own tobacco	5.9%
	FR	Roll your own tobacco	2.6%
	EE	Roll your own tobacco	2.8%
	LU	Roll your own tobacco	2.7%
	IT	Roll your own tobacco	0.9%
Non-compliant cigar weight	CZ	Cigar	8.7%
	IT	Cigar	5.5%
	BE	Cigar	4.9%
	LV	Cigar	4.4%
	FR	Cigar	3.3%
	SE	Cigar	2.8%
	NL	Cigar	2.2%
	LT	Cigar	2.6%
	BG	Cigar	2.5%
	EE	Cigar	1.7%
	LU	Cigar	0.6%
Non-compliant composition - Presence of prohibited additives	BE	Nasal tobacco	16.7%
	BG	Cigar	0.1%
	NL	Cigarillo	0.8%
	FR	Cigarillo	0.4%
	BG	Waterpipe tobacco	0.3%
	SE	Oral tobacco	0.2%
Non-compliant cigarette units per packaging	IT	Cigarette	4.2%
	BG	Cigarette	2.3%
	BE	Cigarette	1.1%
	CZ	Cigarette	1.4%
	NL	Cigarette	0.9%
	FR	Cigarette	0.1%
Non-compliant cigarillo weight	BG	Cigarillo	3.8%
	CZ	Cigarillo	2.4%
	BE	Cigarillo	0.5%
	NL	Cigarillo	0.4%
Non-compliant composition - Presence of CMR substances	NL	Cigarillo	4.1%
	BG	Cigar	0.1%
	BG	Cigarillo	3.8%
	IT	Cigarette	0.1%
	CZ	Cigar	0.1%
	CZ	Cigarillo	1.2%
	IT	Cigarillo	1.4%
	FR	Cigarillo	1.6%
	EE	Waterpipe tobacco	0.1%
Non-compliant TNCO value - Tar	SI	Cigarette	2.1%
	IT	Cigarette	0.1%
	LU	Cigarette	0.3%
Non-compliant TNCO value - Nicotine	SI	Cigarette	1.5%
	LU	Cigarette	0.3%
Non-compliant TNCO value - Co	CZ	Cigarette	0.1%
	SI	Cigarette	0.3%

The highest non-compliance rates are related to the unit weights of rolling tobacco and cigars. Other non-compliances remain low, mostly below 5%, particularly for TNCO values, which are highly standardized.

Regarding inconsistencies, and as observed for electronic cigarettes, it is not surprising to see high error rates for masses: this is explained by the fact that the public EU-CEG datasets only include ingredients above a certain threshold. Therefore, this criterion is more appropriate for application on complete datasets at the Member State level.

Table. Total weight error from the tobacco products EU dataset.

Product Type	BE	BG	CZ	EE	FR	IT	LT	LU	LV	NL	SE	SI
Chewing tobacco			16%		13%	25%				14%	14%	13%
Cigar	9%	9%	15%	4%	4%	6%	1%	1%	4%	1%	2%	0%
Cigarette	10%	8%	7%		9%	6%	4%	7%		3%	1%	6%
Cigarillo	16%	15%	25%		7%	8%	8%	4%	14%	5%	1%	4%
Nasal tobacco			35%									
Novel tobacco product	43%	25%	16%	3%	37%	1%		2%		6%		6%
Oral tobacco											22%	
Pipe tobacco	18%	16%	5%	5%	10%	7%		3%	2%	3%	1%	
Roll your own tobacco	37%	30%	22%	44%	35%	31%	3%	25%	7%	22%	24%	21%
Waterpipe tobacco	30%	37%	33%	39%	14%	3%	34%	1%	13%	0%	14%	11%

The error rates related to unidentifiable ingredients are even lower than for electronic cigarettes, with the exception of chewing tobacco in two countries. This high rate is due to the small number of products of this type.

Table. Ingredient error from the tobacco products EU dataset.

Product Type	BE	BG	CZ	EE	FR	IT	LT	LU	LV	NL	SE	SI
Chewing tobacco			4.7%			12.5%						
Cigar		0.1%	0.0%			0.0%						
Cigarette		0.6%	0.3%					0.3%				0.3%
Cigarillo	1.5%					0.5%			1.3%			
Nasal tobacco			1.7%									
Novel tobacco product		0.4%	0.3%	0.6%	0.7%	0.3%	0.5%	1.5%	0.7%	1.4%	1.5%	0.7%
Oral tobacco											0.2%	
Pipe tobacco	2.5%										1.2%	
Roll your own tobacco	0.6%				0.7%	0.9%						
Waterpipe tobacco		0.3%					0.2%	0.3%	0.4%			

5 Conclusion

Manufacturers of tobacco products and electronic cigarettes are required to provide reliable information in their declarations to the competent authorities of the EU Member States.

By leveraging the data available in the declarative data models, we were able to define, for each product category, automated tests to verify regulatory compliance or detect inconsistencies in the data.

This report presents some general statistics on the application of these criteria to a dataset. However, this is primarily intended as a tool for the competent authorities to follow up with the declarants concerned by these non-compliances or inconsistencies. They should update and correct their declarations accordingly.

Indeed, a declarative error does not necessarily mean a non-compliant product, and a thorough analysis is still essential to assess the product's compliance with the applicable regulations.

Finally, now that manufacturers are accustomed to the EU-CEG declarative format, it would be worthwhile to consider introducing these regulatory compliance and consistency criteria as business rules during the declaration process itself, in order to further enhance the quality of the data submitted in EU-CEG.

6 Annex: SQL code to apply conformity and consisteny criteria

```
--
-- IN-NC.sql version 2024-09-30
--
-- Copyright (c) euceg@anses.fr 2022-2024 (JATC2-WP5)
-- Except specific files which bear a different mention, this programme is licensed under the EUPL-
1.2 or later
-- You may obtain a copy of the license at https://joinup.ec.europa.eu/collection/eupl/eupl-text-
eupl-12
--
-- This activity has received funding from the European Union's Health Program (2014-2020) under
grant agreement N°101035968 (JA-01-2020 - HP-JA-2020 - HP-JA-2020-2).
-- The content of this document represents the views of the author only and is his/her sole
responsibility; it cannot be considered to reflect the views of the European Commission and/or the
European Health and Digital Executive
-- Agency (HaDEA) or any other body of the European Union. The European Commission and the Agency do
not accept any responsibility for use that may be made of the information it contains.
--
-- USE: apply these script on the join EU-CEG public dataset to generate NC and IN tables
--

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- EC
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- -- Table : EC_Recipe
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
DROP TABLE IF EXISTS EC_Recipe;

CREATE TABLE EC_Recipe AS

SELECT
p.NationalMarket,
p.ProductTypeEN,
EC_Comp.ProductID,
EC_Comp.IdentificationRefillContainerCartridge,
EC_Comp.ProductID || '-' || COALESCE(EC_Comp.IdentificationRefillContainerCartridge, '') AS
RecipeID,
EC_Comp.FunctionEN,
EC_Comp.CasNumber,
EC_Comp.IngredientName,
EC_Comp.ConcentrationPPM,
EC_Comp.Weight,
EC_Comp.RecipeWeight,
EC_Comp.RecipeQuantity,
MapIngSubst.SubstCas,
MapIngSubst.SubstName,
MapIngSubst.GenericSubstName,
MapIngSubst.SubstType,
MapIngSubst.PubChemCID,
MapIngSubst.MolecularFormula,
MapIngSubst.MolecularWeight,
MapIngSubst.CanonicalSmile,
MapIngSubst.IsomericSmile,
MapIngSubst.InChiKey,
MapIngSubst.IupacName,
MapIngSubst.Title,
MapIngSubst.ExtractTaxonomy,
MapIngSubst.FlavorWheel,
MapIngSubst.IngID,
MapIngSubst.SubstID,
ParticularitéMAP.Particularité

FROM
EC_Comp
INNER JOIN (SELECT DISTINCT ProductID, NationalMarket, ProductTypeEN FROM EC_List) AS p ON
p.ProductID=EC_Comp.ProductID
LEFT JOIN MapIngSubst ON MapIngSubst.IngID=EC_Comp.IngID

LEFT JOIN ParticularitéMAP
ON    MapIngSubst.SubstCAS = ParticularitéMAP.CasNumberRef
and    MapIngSubst.SubstName = ParticularitéMAP.NameRef

ORDER BY p.NationalMarket, RecipeID, ConcentrationPPM DESC;
```

```
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- -- Table : EC_IN_01
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS EC_IN_01;

CREATE TABLE EC_IN_01 AS

Select
   EC_Recipe.NationalMarket,
   EC_Recipe.ProductTypeEN,
   EC_Recipe.ProductID,
   EC_Recipe.Weight,
   EC_Recipe.RecipeWeight

From
    EC_Recipe
Where
    EC_Recipe.RecipeWeight > (EC_Recipe.Weight)*1.1
    OR     EC_Recipe.RecipeWeight < (EC_Recipe.Weight)*0.9

Group By
   EC_Recipe.NationalMarket,
   EC_Recipe.ProductTypeEN,
   EC_Recipe.ProductID,
    EC_Recipe.Weight,
    EC_Recipe.RecipeWeight;


-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- -- Table : EC_IN_02
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS EC_IN_02;

CREATE TABLE EC_IN_02 AS

Select
    EC_List.NationalMarket,
    EC_List.ProductTypeEN,
    EC_List.ProductID,
    EC_Recipe.Weight/1000/EC_List.LiquidVolumeCapacity,
    EC_Recipe.Weight,
    EC_List.LiquidVolumeCapacity
From
    EC_List

    INNER JOIN EC_Recipe
    ON    EC_List.NationalMarket = EC_Recipe.NationalMarket
    and     EC_List.ProductID = EC_Recipe.ProductID

Where
        EC_List.ProductTypeEN IN ('Refill container/cartridge containing e-liquid.','Electronic
cigarette – Disposable.',
        'Electronic cigarette – Rechargeable, placed on the market with one type of e-liquid (fixed
combination). ')
and
        EC_List.LiquidVolumeCapacity is not null
and
        EC_List.LiquidVolumeCapacity>0
and
        EC_Recipe.Weight/1000/EC_List.LiquidVolumeCapacity>1
and
         EC_Recipe.Weight/1000/EC_List.LiquidVolumeCapacity<1.3

Group By
    EC_List.NationalMarket,
    EC_List.ProductTypeEN,
    EC_List.ProductID;

 -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- -- Table : EC_IN_03
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS EC_IN_03;

CREATE TABLE EC_IN_03 AS
```

```
Select
   EC_Recipe.NationalMarket,
   EC_Recipe.ProductTypeEN,
   EC_Recipe.ProductID,
   EC_Recipe.SubstType
From
    EC_Recipe
Where
    EC_Recipe.SubstType = 'NA'

Group By
   EC_Recipe.NationalMarket,
   EC_Recipe.ProductTypeEN,
   EC_Recipe.SubstType;


 -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- -- Table : EC_IN_04
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS EC_IN_04a;

CREATE TABLE EC_IN_04a AS

Select
   EC_Recipe.NationalMarket,
   EC_Recipe.ProductTypeEN,
   EC_Recipe.ProductID,
   EC_Recipe.RecipeID,
   max(EC_Recipe.GenericSubstName) AS NICOTINE,
   sum(EC_Recipe.RecipeQuantity)  AS QUANTITY

From
    EC_Recipe
INNER JOIN GENICOTINE ON GENICOTINE.GenericSubstName=EC_Recipe.GenericSubstName

Group By
   EC_Recipe.NationalMarket,
   EC_Recipe.ProductTypeEN,
   EC_Recipe.ProductID,
   EC_Recipe.RecipeID;

DROP TABLE IF EXISTS EC_IN_04b;

CREATE TABLE EC_IN_04b AS

Select
   EC_IN_04a.NationalMarket,
   EC_IN_04a.ProductTypeEN,
   EC_IN_04a.ProductID,
   EC_IN_04a.RecipeID,
   EC_IN_04a.QUANTITY,
   EC_List.Volume,
   EC_IN_04a.QUANTITY/EC_List.Volume AS NICO_CONCENTR_DECL,
   EC_List.NicotineConcentration,
   ABS(100*((EC_IN_04a.QUANTITY/EC_List.Volume)-
EC_List.NicotineConcentration)/EC_List.NicotineConcentration) AS TAUX_ECART_NIC

From
    EC_IN_04a
INNER JOIN EC_List ON EC_List.ProductID=EC_IN_04a.ProductID
and EC_List.NationalMarket=EC_IN_04a.NationalMarket

where       EC_List.NicotineConcentration is not null

Group By
   EC_IN_04a.NationalMarket,
   EC_IN_04a.ProductTypeEN,
   EC_IN_04a.ProductID,
   EC_IN_04a.RecipeID;

 DROP TABLE IF EXISTS EC_IN_04;

CREATE TABLE EC_IN_04 AS

Select
   EC_IN_04b.NationalMarket,
```

```
   EC_IN_04b.ProductTypeEN,
   EC_IN_04b.ProductID,
   EC_IN_04b.QUANTITY,
   EC_IN_04b.Volume,
   EC_IN_04b.NICO_CONCENTR_DECL,
   EC_IN_04b.NicotineConcentration,
   EC_IN_04b.TAUX_ECART_NIC
From
    EC_IN_04b

WHERE    EC_IN_04b.TAUX_ECART_NIC > 5

Group By
   EC_IN_04b.NationalMarket,
   EC_IN_04b.ProductTypeEN,
   EC_IN_04b.ProductID;



 -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- -- Table : EC_NC_02
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS EC_NC_02;

CREATE TABLE EC_NC_02 AS

Select
   EC_Recipe.NationalMarket,
   EC_Recipe.ProductTypeEN,
   EC_Recipe.ProductID,
   EC_Recipe.Particularité

From
    EC_Recipe
Where
   EC_Recipe.Particularité ='CMR'

Group By
   EC_Recipe.NationalMarket,
   EC_Recipe.ProductTypeEN,
   EC_Recipe.ProductID;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- -- Table : EC_NC_03
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS EC_NC_03;

CREATE TABLE EC_NC_03 AS

Select
   EC_Recipe.NationalMarket,
   EC_Recipe.ProductTypeEN,
   EC_Recipe.ProductID,
   EC_Recipe.Particularité

From
    EC_Recipe
Where
   EC_Recipe.Particularité ='ADDITIF_INTERDIT'

Group By
   EC_Recipe.NationalMarket,
   EC_Recipe.ProductTypeEN,
   EC_Recipe.ProductID;



-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- -- Table : EC_NC_06
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS EC_NC_06;

CREATE TABLE EC_NC_06 AS

Select
```

```
    EC_List.NationalMarket,
    EC_List.ProductTypeEN,
    EC_List.ProductID,
    EC_List.NicotineConcentration
From
    EC_List
Where
        cast(EC_List.NicotineConcentration as real) > 20
Group By
    EC_List.NationalMarket,
    EC_List.ProductTypeEN,
    EC_List.ProductID;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- -- Table : EC_NC_07
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS EC_NC_07;

CREATE TABLE EC_NC_07 AS

Select
    EC_List.NationalMarket,
    EC_List.ProductTypeEN,
    EC_List.ProductID AS ProductID,
    EC_List.LiquidVolumeCapacity,
    EC_List.ProductTypeEN
From
    EC_List
Where
        EC_List.ProductTypeEN='Refill container/cartridge containing e-liquid.'
        and
        cast(EC_List.LiquidVolumeCapacity as real) > 10
Group By
    EC_List.NationalMarket,
    EC_List.ProductTypeEN,
    EC_List.ProductID;




-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
--   TP
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- -- Table : TP_Recipe
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
DROP TABLE IF EXISTS TP_Recipe;

CREATE TABLE TP_Recipe AS


SELECT
p.NationalMarket,
p.ProductTypeEN,
TP_comp.ProductID,
TP_comp.FunctionEN,
TP_comp.CategoryEN,
TP_comp.CasNumber,
TP_comp.IngredientName,
TP_comp.ConcentrationPPM,
TP_comp.Weight,
TP_comp.TobaccoWeight,
TP_comp.RecipeQuantity,
MapIngSubst.SubstCas,
MapIngSubst.SubstName,
MapIngSubst.GenericSubstName,
MapIngSubst.SubstType,
MapIngSubst.PubChemCID,
MapIngSubst.MolecularFormula,
MapIngSubst.MolecularWeight,
MapIngSubst.CanonicalSmile,
MapIngSubst.IsomericSmile,
MapIngSubst.InChiKey,
MapIngSubst.IupacName,
MapIngSubst.Title,
MapIngSubst.ExtractTaxonomy,
```

```
MapIngSubst.FlavorWheel,
MapIngSubst.IngID,
MapIngSubst.SubstID,
ParticularitéMAP.Particularité


FROM
TP_comp
INNER JOIN (SELECT DISTINCT ProductID, NationalMarket, ProductTypeEN FROM TP_List) AS p ON
p.ProductID=TP_comp.ProductID
LEFT JOIN MapIngSubst ON MapIngSubst.IngID=TP_comp.IngID


LEFT JOIN ParticularitéMAP
ON    MapIngSubst.SubstCAS = ParticularitéMAP.CasNumberRef
and    MapIngSubst.SubstName = ParticularitéMAP.NameRef

ORDER BY NationalMarket, TP_comp.ProductID, ConcentrationPPM DESC;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- Table : TP_IN_01
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

-- Table : TP_IN_01_A

DROP TABLE IF EXISTS TP_IN_01A;

CREATE TABLE TP_IN_01A AS

SELECT
TP_Recipe.NationalMarket,
TP_Recipe.ProductTypeEN,
TP_Recipe.ProductID,
TP_Recipe.CategoryEN,
TP_Recipe.FunctionEN,
sum(TP_Recipe.RecipeQuantity),
TP_Recipe.TobaccoWeight
From
TP_Recipe


where TP_Recipe.CategoryEN is null and TP_Recipe.FunctionEN is null

      GROUP BY
TP_Recipe.NationalMarket,
TP_Recipe.ProductTypeEN,
TP_Recipe.ProductID,
TP_Recipe.TobaccoWeight;


-- Table : TP_IN_01

DROP TABLE IF EXISTS TP_IN_01;

CREATE TABLE TP_IN_01 AS

Select
    TP_IN_01A.NationalMarket,
    TP_IN_01A.ProductTypeEN,
    TP_IN_01A.ProductID,
   TP_IN_01A.'sum(TP_Recipe.RecipeQuantity)',
   TP_IN_01A.TobaccoWeight

From
    TP_IN_01A
Where
    TP_IN_01A.'sum(TP_Recipe.RecipeQuantity)' > (TP_IN_01A.TobaccoWeight)*1.1
    OR     TP_IN_01A.'sum(TP_Recipe.RecipeQuantity)' < (TP_IN_01A.TobaccoWeight)*0.9

Group By
    TP_IN_01A.NationalMarket,
    TP_IN_01A.ProductTypeEN,
    TP_IN_01A.ProductID,
    TP_IN_01A.TobaccoWeight,
    TP_IN_01A.'sum(TP_Recipe.RecipeQuantity)';


-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
```

```
-- Table : TP_IN_02
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

-- Table : TP_IN_02_A

DROP TABLE IF EXISTS TP_IN_02A;

CREATE TABLE TP_IN_02A AS

SELECT
TP_Recipe.NationalMarket,
TP_Recipe.ProductTypeEN,
TP_Recipe.ProductID,
sum(TP_Recipe.RecipeQuantity),
TP_Recipe.Weight
From
TP_Recipe


      GROUP BY
TP_Recipe.NationalMarket,
TP_Recipe.ProductTypeEN,
TP_Recipe.ProductID,
TP_Recipe.TobaccoWeight;


-- Table : TP_IN_02

DROP TABLE IF EXISTS TP_IN_02;

CREATE TABLE TP_IN_02 AS

Select
TP_IN_02A.NationalMarket,
TP_IN_02A.ProductTypeEN,
TP_IN_02A.ProductID,
   TP_IN_02A.'sum(TP_Recipe.RecipeQuantity)',
   TP_IN_02A.Weight

From
    TP_IN_02A
Where
    TP_IN_02A.'sum(TP_Recipe.RecipeQuantity)' > (TP_IN_02A.Weight)*1.1
    OR     TP_IN_02A.'sum(TP_Recipe.RecipeQuantity)' < (TP_IN_02A.Weight)*0.9

Group By
TP_IN_02A.NationalMarket,
TP_IN_02A.ProductTypeEN,
TP_IN_02A.ProductID,
    TP_IN_02A.Weight,
    TP_IN_02A.'sum(TP_Recipe.RecipeQuantity)';

 -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- -- Table : TP_IN_03
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS TP_IN_03;

CREATE TABLE TP_IN_03 AS

Select
   TP_Recipe.NationalMarket,
   TP_Recipe.ProductTypeEN,
   TP_Recipe.ProductID,
   TP_Recipe.SubstType
From
    TP_Recipe
Where
    TP_Recipe.SubstType = 'NA'

Group By
   TP Recipe.NationalMarket,
   TP_Recipe.ProductTypeEN,
   TP_Recipe.SubstType;



 -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
```

```
-- -- Table : TP_NC_02
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS TP_NC_02;


CREATE TABLE TP_NC_02 AS

Select
   TP_Recipe.NationalMarket,
   TP_Recipe.ProductTypeEN,
   TP_Recipe.ProductID,
   TP_Recipe.Particularité

From
    TP_Recipe
Where
   TP_Recipe.Particularité ='CMR'

Group By
   TP_Recipe.NationalMarket,
   TP_Recipe.ProductTypeEN,
   TP_Recipe.ProductID;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- -- Table : TP_NC_03
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS TP_NC_03;

CREATE TABLE TP_NC_03 AS

Select
   TP_Recipe.NationalMarket,
   TP_Recipe.ProductTypeEN,
   TP_Recipe.ProductID,
   TP_Recipe.Particularité

From
    TP_Recipe
Where
   TP_Recipe.Particularité ='ADDITIF_INTERDIT'

Group By
   TP_Recipe.NationalMarket,
   TP_Recipe.ProductTypeEN,
   TP_Recipe.ProductID;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- Table : TP_NC_05
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS TP_NC_05;

CREATE TABLE TP_NC_05 AS

Select
TP_List.NationalMarket,
TP_List.ProductTypeEN,
TP_List.ProductID,
   TP_List.Tar,
   TP_List.ProductTypeEN

From
    TP_List
Where
     TP_List.ProductTypeEN='Cigarette'
     AND TP_List.Tar>10


Group By
TP List.NationalMarket,
TP_List.ProductTypeEN,
TP_List.ProductID,
   TP_List.Tar,
   TP_List.ProductTypeEN;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
```

```
-- Table : TP_NC_06
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS TP_NC_06;

CREATE TABLE TP_NC_06 AS

Select
TP_List.NationalMarket,
TP_List.ProductTypeEN,
TP_List.ProductID,
   TP_List.Co

From
    TP_List
Where
     TP_List.ProductTypeEN='Cigarette'
     AND
    TP_List.Co>10

Group By
TP_List.NationalMarket,
TP_List.ProductTypeEN,
TP_List.ProductID,
   TP_List.Co;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- Table : TP_NC_07
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS TP_NC_07;

CREATE TABLE TP_NC_07 AS

Select
TP_List.NationalMarket,
TP_List.ProductTypeEN,
TP_List.ProductID,
   TP_List.Nicotine,
   TP_List.ProductTypeEN

From
    TP_List
Where
     TP_List.ProductTypeEN='Cigarette'
     AND TP_List.Nicotine>1


Group By
TP_List.NationalMarket,
TP_List.ProductTypeEN,
TP_List.ProductID,
   TP_List.Nicotine,
   TP_List.ProductTypeEN;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- Table : TP_NC_08
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS TP_NC_08;

CREATE TABLE TP_NC_08 AS

Select
TP_List.NationalMarket,
TP_List.ProductTypeEN,
TP_List.ProductID,
   TP_List.Co,
   TP_List.Nicotine,
   TP_List.Tar,
   TP_List.ProductTypeEN

From
    TP_List

Where TP_List.ProductTypeEN='Cigarette'
 AND
 (TP_List.Tar=='-'
```

```
 or TP_List.Nicotine=='-'
 or TP_List.Co=='-')


Group By
TP_List.NationalMarket,
TP_List.ProductTypeEN,
TP_List.ProductID,
   TP_List.Tar,
   TP_List.ProductTypeEN;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- Table : TP_NC_09
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS TP_NC_09;

CREATE TABLE TP_NC_09 AS

Select
TP_List.NationalMarket,
TP_List.ProductTypeEN,
TP_List.ProductID,
   TP_List.ProductTypeEN,
   TP_List.Weight

From
    TP_List
Where
    TP_List.ProductTypeEN='Cigarillo'
    AND
   TP_List.Weight>3000
      Group By
TP_List.NationalMarket,
TP_List.ProductTypeEN,
TP_List.ProductID,
   TP_List.ProductTypeEN,
   TP_List.Weight;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- Table : TP_NC_10
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS TP_NC_10;

CREATE TABLE TP_NC_10 AS

Select
TP_List.NationalMarket,
TP_List.ProductTypeEN,
TP_List.ProductID,
   TP_List.ProductTypeEN,
   TP_List.Weight

From
    TP_List
Where
    TP_List.ProductTypeEN='Cigar'
    AND
   TP_List.Weight<=3000
      Group By
TP_List.NationalMarket,
TP_List.ProductTypeEN,
TP_List.ProductID,
   TP_List.ProductTypeEN,
   TP_List.Weight;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- Table : TP_NC_11
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS TP NC 11;

CREATE TABLE TP_NC_11 AS

Select
TP List.NationalMarket,
TP_List.ProductTypeEN,
```

```
TP_List.ProductID,
   TP_List.ProductTypeEN,
   TP_List.PackageUnits

From
    TP_List
Where
    TP_List.ProductTypeEN='Cigarette'
    AND
    (TP_List.PackageUnits<20
    OR
    ROUND(TP_List.PackageUnits/5,0)<>TP_List.PackageUnits/5)

Group By
TP_List.NationalMarket,
TP_List.ProductTypeEN,
TP_List.ProductID,
   TP_List.ProductTypeEN,
   TP_List.PackageUnits;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- Table : TP_NC_12
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS TP_NC_12;

CREATE TABLE TP_NC_12 AS

Select
TP_List.NationalMarket,
TP_List.ProductTypeEN,
TP_List.ProductID,
   TP_List.ProductTypeEN,
   TP_List.PackageNetWeight

From
    TP_List
Where
    TP_List.ProductTypeEN='Roll your own tobacco'
    AND
    (TP_List.PackageNetWeight<30
    OR
    ROUND(TP_List.PackageNetWeight/5,0)<>TP_List.PackageNetWeight/5)

Group By
TP_List.NationalMarket,
TP_List.ProductTypeEN,
TP_List.ProductID,
   TP_List.ProductTypeEN,
   TP_List.PackageNetWeight


 ------------------------------------------------------------------------------------------------------------
-----------------------------------------
 --- ADDING INNC
  ------------------------------------------------------------------------------------------------------------
-----------------------------------------

 -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- EC
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS Publi_EC_INNC_A;

CREATE TABLE Publi_EC_INNC_A (
    NationalMarket                      TEXT,
    LastUpdate  DATE,
    ProductTypeEN                         TEXT,
    ProductID                           TEXT,
    Type_Ecart                           TEXT,
    Name_INNC                    TEXT,
    Code_INNC                            TEXT
);

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : EC_IN_01
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
```

```
INSERT INTO Publi_EC_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)

SELECT
    EC_List.NationalMarket AS NationalMarket,
    EC_List.LastUpdate AS LastUpdate,
    EC_List.ProductTypeEN AS ProductTypeEN,
    EC_List.ProductID AS ProductID,
    'EC-IN-01' AS Code_INNC
FROM
    EC_List
    INNER JOIN EC_IN_01 On EC_List.ProductID = EC_IN_01.ProductID;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : EC IN 02
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --


INSERT INTO Publi_EC_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)

SELECT
    EC_List.NationalMarket AS NationalMarket,
    EC_List.LastUpdate AS LastUpdate,
    EC_List.ProductTypeEN AS ProductTypeEN,
    EC_List.ProductID AS ProductID,
    'EC-IN-02' AS Code_INNC

FROM
    EC_List
    INNER JOIN EC_IN_02 On EC_List.ProductID = EC_IN_02.ProductID;


-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : EC_IN_03
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --


INSERT INTO Publi_EC_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)

SELECT
    EC_List.NationalMarket AS NationalMarket,
    EC_List.LastUpdate AS LastUpdate,
    EC_List.ProductTypeEN AS ProductTypeEN,
    EC_List.ProductID AS ProductID,
    'EC-IN-03' AS Code_INNC

FROM
    EC_List
    INNER JOIN EC_IN_03 On EC_List.ProductID = EC_IN_03.ProductID;



-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : EC_IN_04
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --


INSERT INTO Publi_EC_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN.
```

```
    ProductID,
    Code_INNC
)

SELECT
    EC_List.NationalMarket AS NationalMarket,
    EC_List.LastUpdate AS LastUpdate,
    EC_List.ProductTypeEN AS ProductTypeEN,
    EC_List.ProductID AS ProductID,
    'EC-IN-04' AS Code_INNC

FROM
    EC_List
    INNER JOIN EC_IN_04 On EC_List.ProductID = EC_IN_04.ProductID;



-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : EC_NC_02
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --


INSERT INTO Publi_EC_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)

SELECT
    EC_List.NationalMarket AS NationalMarket,
    EC_List.LastUpdate AS LastUpdate,
    EC_List.ProductTypeEN AS ProductTypeEN,
    EC_List.ProductID AS ProductID,
    'EC-NC-02' AS Code_INNC

FROM
    EC_List
    INNER JOIN EC_NC_02 On EC_List.ProductID = EC_NC_02.ProductID;


    -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : EC_NC_03
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --


INSERT INTO Publi_EC_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)

SELECT
    EC_List.NationalMarket AS NationalMarket,
    EC_List.LastUpdate AS LastUpdate,
    EC_List.ProductTypeEN AS ProductTypeEN,
    EC_List.ProductID AS ProductID,
    'EC-NC-03' AS Code_INNC

FROM
    EC_List
    INNER JOIN EC_NC_03 On EC_List.ProductID = EC_NC_03.ProductID;


-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : EC_NC_06
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --


INSERT INTO Publi_EC_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
```

```
)

SELECT
    EC_List.NationalMarket AS NationalMarket,
    EC_List.LastUpdate AS LastUpdate,
    EC_List.ProductTypeEN AS ProductTypeEN,
    EC_List.ProductID AS ProductID,
    'EC-NC-06' AS Code_INNC

FROM
    EC_List
    INNER JOIN EC_NC_06 On EC_List.ProductID = EC_NC_06.ProductID;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : EC_NC_07
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --


INSERT INTO Publi_EC_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)

SELECT
    EC_List.NationalMarket AS NationalMarket,
    EC_List.LastUpdate AS LastUpdate,
    EC_List.ProductTypeEN AS ProductTypeEN,
    EC_List.ProductID AS ProductID,
    'EC-NC-07' AS Code_INNC

FROM
    EC_List
    INNER JOIN EC_NC_07 On EC_List.ProductID = EC_NC_07.ProductID;



-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADJUST : Publi_EC_INNC_A
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS Publi_EC_INNC;

CREATE TABLE Publi_EC_INNC AS


SELECT
    Publi_EC_INNC_A.NationalMarket,
    Publi_EC_INNC_A.LastUpdate,
    Publi_EC_INNC_A.ProductID,
    Publi_EC_INNC_A.ProductTypeEN,
    INNC_Description.Type_Ecart,
    INNC_Description.Name_INNC,
    INNC_Description.Code_INNC

from Publi_EC_INNC_A
    INNER JOIN INNC_Description On Publi_EC_INNC_A.Code_INNC = INNC_Description.Code_INNC
    INNER JOIN Dataset_selection On Publi_EC_INNC_A.ProductTypeEN = Dataset_selection.ProductTypeEN
    WHERE Dataset_selection.D58_S = 'true'

GROUP BY Publi_EC_INNC_A.ProductID,
         INNC_Description.Code_INNC;


-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- EC_INNC_OVERVIEW
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS EC_INNC_OVERVIEW_a;

CREATE TABLE EC_INNC_OVERVIEW_a AS


Select
    Publi EC INNC.Code INNC,
    Publi_EC_INNC.Name_INNC,
```

```
    Publi_EC_INNC.NationalMarket,
    Publi_EC_INNC.ProductTypeEN,
    count(Publi_EC_INNC.ProductID) as NB_INNC,
    EC_LIST_NB_TOT.NB_TOT


From
   Publi_EC_INNC

LEFT JOIN EC_LIST_NB_TOT
ON    Publi_EC_INNC.NationalMarket = EC_LIST_NB_TOT.NationalMarket
and   Publi_EC_INNC.ProductTypeEN = EC_LIST_NB_TOT.ProductTypeEN

Group by
    Publi_EC_INNC.Code_INNC,
    Publi_EC_INNC.Name_INNC,
    Publi_EC_INNC.NationalMarket,
    Publi_EC_INNC.ProductTypeEN,
    EC_LIST_NB_TOT.NB_TOT;



DROP TABLE IF EXISTS EC_INNC_OVERVIEW_b;

CREATE TABLE EC_INNC_OVERVIEW_b AS


Select
    EC_INNC_OVERVIEW_a.Code_INNC,
    EC_INNC_OVERVIEW_a.Name_INNC,
    EC_INNC_OVERVIEW_a.NationalMarket,
    sum(EC_INNC_OVERVIEW_a.NB_INNC) as NB_INNC,
    sum(EC_INNC_OVERVIEW_a.NB_TOT) AS NB_TOT

From
   EC_INNC_OVERVIEW_a

Group by
    EC_INNC_OVERVIEW_a.Code_INNC,
    EC_INNC_OVERVIEW_a.Name_INNC,
    EC_INNC_OVERVIEW_a.NationalMarket;


DROP TABLE IF EXISTS EC_INNC_OVERVIEW;

CREATE TABLE EC_INNC_OVERVIEW AS


Select
    EC_INNC_OVERVIEW_b.Code_INNC,
    EC_INNC_OVERVIEW_b.Name_INNC,
    EC_INNC_OVERVIEW_b.NationalMarket,
    EC_INNC_OVERVIEW_b.NB_INNC,
    EC_INNC_OVERVIEW_b.NB_TOT,
    100*EC_INNC_OVERVIEW_b.NB_INNC/EC_INNC_OVERVIEW_b.NB_TOT as TAUX_INNC

From
   EC_INNC_OVERVIEW_b

Group by
    EC_INNC_OVERVIEW_b.Code_INNC,
    EC_INNC_OVERVIEW_b.Name_INNC,
    EC_INNC_OVERVIEW_b.NationalMarket,
    EC_INNC_OVERVIEW_b.NB_TOT;


-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
--   TP
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS Publi_TP_INNC_A;

CREATE TABLE Publi_TP_INNC_A (
    NationalMarket                    TEXT,
    LastUpdate  DATE,
    ProductTypeEN                       TEXT,
    ProductID                         TEXT,
    Type_Ecart                         TEXT,
```

```
    Name_INNC                           TEXT,
    Code_INNC                             TEXT
);

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : TP_IN_01
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
INSERT INTO Publi_TP_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)
SELECT
    TP_List.NationalMarket AS NationalMarket,
    TP_List.LastUpdate AS LastUpdate,
    TP_List.ProductTypeEN AS ProductTypeEN,
    TP_List.ProductID AS ProductID,
    'TP-IN-01' AS Code_INNC
FROM
    TP_List
    INNER JOIN TP_IN_01 On TP_List.ProductID = TP_IN_01.ProductID;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : TP_IN_02
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
INSERT INTO Publi_TP_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)
SELECT
    TP_List.NationalMarket AS NationalMarket,
    TP_List.LastUpdate AS LastUpdate,
    TP_List.ProductTypeEN AS ProductTypeEN,
    TP_List.ProductID AS ProductID,
    'TP-IN-02' AS Code_INNC
FROM
    TP_List
    INNER JOIN TP_IN_02 On TP_List.ProductID = TP_IN_02.ProductID;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : TP_IN_03
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
INSERT INTO Publi_TP_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)
SELECT
    TP_List.NationalMarket AS NationalMarket,
    TP_List.LastUpdate AS LastUpdate,
    TP_List.ProductTypeEN AS ProductTypeEN,
    TP_List.ProductID AS ProductID,
    'TP-IN-03' AS Code_INNC
FROM
    TP_List
    INNER JOIN TP_IN_03 On TP_List.ProductID = TP_IN_03.ProductID;


-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : TP_NC_02
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
INSERT INTO Publi_TP_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)
SELECT
    TP List.NationalMarket AS NationalMarket,
    TP_List.LastUpdate AS LastUpdate,
```

```
    TP_List.ProductTypeEN AS ProductTypeEN,
    TP_List.ProductID AS ProductID,
    'TP-NC-02' AS Code_INNC
FROM
    TP_List
    INNER JOIN TP_NC_02 On TP_List.ProductID = TP_NC_02.ProductID;

    -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : TP_NC_03
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
INSERT INTO Publi_TP_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)
SELECT
    TP_List.NationalMarket AS NationalMarket,
    TP_List.LastUpdate AS LastUpdate,
    TP_List.ProductTypeEN AS ProductTypeEN,
    TP_List.ProductID AS ProductID,
    'TP-NC-03' AS Code_INNC
FROM
    TP_List
    INNER JOIN TP_NC_03 On TP_List.ProductID = TP_NC_03.ProductID;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : TP_NC_05
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
INSERT INTO Publi_TP_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)
SELECT
    TP_List.NationalMarket AS NationalMarket,
    TP_List.LastUpdate AS LastUpdate,
    TP_List.ProductTypeEN AS ProductTypeEN,
    TP_List.ProductID AS ProductID,
    'TP-NC-05' AS Code_INNC
FROM
    TP_List
    INNER JOIN TP_NC_05 On TP_List.ProductID = TP_NC_05.ProductID;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : TP_NC_06
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
INSERT INTO Publi_TP_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)
SELECT
    TP_List.NationalMarket AS NationalMarket,
    TP_List.LastUpdate AS LastUpdate,
    TP_List.ProductTypeEN AS ProductTypeEN,
    TP_List.ProductID AS ProductID,
    'TP-NC-06' AS Code_INNC
FROM
    TP_List
    INNER JOIN TP_NC_06 On TP_List.ProductID = TP_NC_06.ProductID;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : TP_NC_07
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
INSERT INTO Publi_TP_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)
SELECT
```

```
    TP_List.NationalMarket AS NationalMarket,
    TP_List.LastUpdate AS LastUpdate,
    TP_List.ProductTypeEN AS ProductTypeEN,
    TP_List.ProductID AS ProductID,
    'TP-NC-07' AS Code_INNC
FROM
    TP_List
    INNER JOIN TP_NC_07 On TP_List.ProductID = TP_NC_07.ProductID;


-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : TP_NC_08
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
INSERT INTO Publi_TP_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)
SELECT
    TP_List.NationalMarket AS NationalMarket,
    TP_List.LastUpdate AS LastUpdate,
    TP_List.ProductTypeEN AS ProductTypeEN,
    TP_List.ProductID AS ProductID,
    'TP-NC-08' AS Code_INNC
FROM
    TP_List
    INNER JOIN TP_NC_08 On TP_List.ProductID = TP_NC_08.ProductID;


-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : TP_NC_09
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
INSERT INTO Publi_TP_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)
SELECT
    TP_List.NationalMarket AS NationalMarket,
    TP_List.LastUpdate AS LastUpdate,
    TP_List.ProductTypeEN AS ProductTypeEN,
    TP_List.ProductID AS ProductID,
    'TP-NC-09' AS Code_INNC
FROM
    TP_List
    INNER JOIN TP_NC_09 On TP_List.ProductID = TP_NC_09.ProductID;


-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : TP_NC_10
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
INSERT INTO Publi_TP_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)
SELECT
    TP_List.NationalMarket AS NationalMarket,
    TP_List.LastUpdate AS LastUpdate,
    TP_List.ProductTypeEN AS ProductTypeEN,
    TP_List.ProductID AS ProductID,
    'TP-NC-10' AS Code_INNC
FROM
    TP_List
    INNER JOIN TP_NC_10 On TP_List.ProductID = TP_NC_10.ProductID;


-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : TP_NC_11
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
INSERT INTO Publi_TP_INNC_A (
```

```
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)
SELECT
    TP_List.NationalMarket AS NationalMarket,
    TP_List.LastUpdate AS LastUpdate,
    TP_List.ProductTypeEN AS ProductTypeEN,
    TP_List.ProductID AS ProductID,
    'TP-NC-11' AS Code_INNC
FROM
    TP_List
    INNER JOIN TP_NC_11 On TP_List.ProductID = TP_NC_11.ProductID;

-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : TP_NC_12
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
INSERT INTO Publi_TP_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)
SELECT
    TP_List.NationalMarket AS NationalMarket,
    TP_List.LastUpdate AS LastUpdate,
    TP_List.ProductTypeEN AS ProductTypeEN,
    TP_List.ProductID AS ProductID,
    'TP-NC-12' AS Code_INNC
FROM
    TP_List
    INNER JOIN TP_NC_12 On TP_List.ProductID = TP_NC_12.ProductID;


-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADD : TP_NC_13
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
INSERT INTO Publi_TP_INNC_A (
    NationalMarket,
    LastUpdate,
    ProductTypeEN,
    ProductID,
    Code_INNC
)
SELECT
    TP_List.NationalMarket AS NationalMarket,
    TP_List.LastUpdate AS LastUpdate,
    TP_List.ProductTypeEN AS ProductTypeEN,
    TP_List.ProductID AS ProductID,
    'TP-NC-13' AS Code_INNC
FROM
    TP_List
    INNER JOIN TP_NC_13 On TP_List.ProductID = TP_NC_13.ProductID;



-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- ADJUST : Publi_TP_INNC_A
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS Publi_TP_INNC;

CREATE TABLE Publi_TP_INNC AS


SELECT
    Publi_TP_INNC_A.NationalMarket,
    Publi_TP_INNC_A.LastUpdate,
    Publi_TP_INNC_A.ProductTypeEN,
    Publi_TP_INNC_A.ProductID,
    INNC_Description.Type_Ecart,
    INNC_Description.Name_INNC,
    INNC_Description.Code_INNC

from Publi_TP_INNC_A
```

```
    INNER JOIN INNC_Description On Publi_TP_INNC_A.Code_INNC = INNC_Description.Code_INNC
GROUP BY Publi_TP_INNC_A.ProductID,
         INNC_Description.Code_INNC;


-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
-- TP_INNC_OVERVIEW
-- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

DROP TABLE IF EXISTS TP_INNC_OVERVIEW;

CREATE TABLE TP_INNC_OVERVIEW AS

Select
    Publi_TP_INNC.Code_INNC,
    Publi_TP_INNC.Name_INNC,
    Publi_TP_INNC.NationalMarket,
    Publi_TP_INNC.ProductTypeEN,
    count(Publi_TP_INNC.ProductID) as NB_INNC,
    TP_LIST_NB_TOT.NB_TOT,
    100*count(Publi_TP_INNC.ProductID)/TP_LIST_NB_TOT.NB_TOT as TAUX_INNC

From
   Publi_TP_INNC

LEFT JOIN TP_LIST_NB_TOT
ON    Publi_TP_INNC.NationalMarket = TP_LIST_NB_TOT.NationalMarket
and   Publi_TP_INNC.ProductTypeEN = TP_LIST_NB_TOT.ProductTypeEN

Group by
    Publi_TP_INNC.Code_INNC,
    Publi_TP_INNC.Name_INNC,
    Publi_TP_INNC.NationalMarket,
    Publi_TP_INNC.ProductTypeEN,
    TP_LIST_NB_TOT.NB_TOT
```

***