Rattrapage Analyste Programmeur

Une société souhaite alimenter sa base de données produit à partir d'un fichier Excel.

La table produit existante est la suivante :

	Туре	Interclassement	Attributs	Null	Défaut	Extra	Action						
pr_cd_pr	bigint(16)			Non	0			1	×		U	3	T
pr_desi	varchar(100)	latin1_swedish_ci		Non	aucune			1	X	T	Ü	1	T
pr_stre	int(11)			Non	0			1	×		U	3	T
pr_douane	varchar(14)	latin1_swedish_ci		Non	0			1	X	T	Ü	1	T
pr_prac	decimal(8,2)			Non	0.00			1	X	T	U	3	T
pr_deg	int(11)			Non	0			1	×		U	3	T
pr_pdn	int(11)			Non	0			1	X	T	U	3	T
pr_pdb	int(11)			Non	0			1	×		U	3	T
pr_four	int(11)			Non	0			1	×		U	3	T
pr_refour	varchar(40)	latin1_swedish_ci		Non	aucune			1	X		U	3	T
pr_codebarre	bigint(14)			Oui	NULL			1	×		U	3	T
pr_modele	varchar(30)	latin1_swedish_ci		Non	aucune			1	X		U	1	T
pr_prix	decimal(8,2)			Non	aucune			<i>></i>	X	T	U	3	T
pr_pack	int(11)			Non	aucune			1	×		U	3	T
ppppppppppppppppppppppppppppppppppppppp	or_desi or_stre or_douane or_prac or_pdg or_pdh or_pdb or_four or_refour or_codebarre or_modele	or_desi varchar(100) or_stre int(11) or_douane varchar(14) or_prac decimal(8,2) or_deg int(11) or_pdn int(11) or_pdb int(11) or_four int(11) or_refour varchar(40) or_codebarre bigint(14) or_modele varchar(30) or_prix decimal(8,2)	varchar(100) latin1_swedish_ci int_stre int(11) in_douane varchar(14) latin1_swedish_ci int_prac decimal(8,2) int_deg int(11) in_pdh int(11) in_four int(11) in_refour varchar(40) latin1_swedish_ci int_codebarre bigint(14) in_modele varchar(30) latin1_swedish_ci in_prix decimal(8,2)	or_desi varchar(100) latin1_swedish_ci or_stre int(11) or_douane varchar(14) latin1_swedish_ci or_prac decimal(8,2) or_deg int(11) or_pdn int(11) or_pdb int(11) or_four int(11) or_refour varchar(40) latin1_swedish_ci or_codebarre bigint(14) or_modele varchar(30) latin1_swedish_ci or_prix decimal(8,2)	or_desi varchar(100) latin1_swedish_ci Non or_stre int(11) Non or_douane varchar(14) latin1_swedish_ci Non or_prac decimal(8,2) Non or_deg int(11) Non or_pdn int(11) Non or_pdb int(11) Non or_four int(11) Non or_refour varchar(40) latin1_swedish_ci Non or_codebarre bigint(14) Oui or_modele varchar(30) latin1_swedish_ci Non or_prix decimal(8,2) Non	or_desi varchar(100) latin1_swedish_ci Non aucune or_stre int(11) Non 0 or_douane varchar(14) latin1_swedish_ci Non 0 or_prac decimal(8,2) Non 0.00 or_deg int(11) Non 0 or_pdn int(11) Non 0 or_pdb int(11) Non 0 or_four int(11) Non 0 or_four int(11) Non aucune or_codebarre bigint(14) Oui NULL or_modele varchar(30) latin1_swedish_ci Non aucune or_prix decimal(8,2) Non aucune	or_desi varchar(100) latin1_swedish_ci Non aucune or_stre int(11) Non 0 or_douane varchar(14) latin1_swedish_ci Non 0 or_prac decimal(8,2) Non 0.00 or_deg int(11) Non 0 or_pdn int(11) Non 0 or_pdb int(11) Non 0 or_four int(11) Non 0 or_four int(11) Non aucune or_codebarre bigint(14) Oui NULL or_modele varchar(30) latin1_swedish_ci Non aucune or_prix decimal(8,2) Non aucune	or_desi varchar(100) latin1_swedish_ci Non aucune or_stre int(11) Non 0 or_douane varchar(14) latin1_swedish_ci Non 0 or_prac decimal(8,2) Non 0.00 III or_deg int(11) Non 0 III or_pdh int(11) Non 0 III or_pdb int(11) Non 0 III or_four int(11) Non 0 III or_refour varchar(40) latin1_swedish_ci Non aucune III or_modele varchar(30) latin1_swedish_ci Non aucune III or_prix decimal(8,2) Non aucune III	or_desi varchar(100) latin1_swedish_ci Non aucune latin1_swedish_ci Non 0 latin1_swedish_ci Non 0 latin1_swedish_ci Non 0 latin1_swedish_ci Non 0 latin1_swedish_ci Non 0.00 latin1_swedish_ci Non 0.00 latin1_swedish_ci Non 0 latin1_swedish_ci Non 0 latin1_swedish_ci Non 0 latin1_swedish_ci Non 0 latin1_swedish_ci Non aucune latin1_swe	or_desi varchar(100) latin1_swedish_ci Non aucune Image: Comparity of the comparity o	or_desi varchar(100) latin1_swedish_ci Non aucune	or_desi varchar(100) latin1_swedish_ci Non aucune	varchar(100) latin1_swedish_ci Non aucune int(11) Non 0 int_douane varchar(14) latin1_swedish_ci Non 0 int_douane varchar(14) latin1_swedish_ci Non 0 int_deg int(11) Non 0 int_deg int(11) Non 0 int(11) int_douane int(11) Non 0 int_douane int(11) Non 0 int_douane int(11) int_douane int(11) Non 0 int_douane int(11) int_douane int(11) int_douane int_

Les correspondances entre le fichier Excel et la base de données sont les suivantes :

(À **gauche** les champs de la base de données → à **droite** les colonnes du fichier Excel)

<u>Référence fournisseur</u>: **pr_refour** → Colonne A du fichier <u>Désignation</u>: **pr_desi** → Colonne C + Colonne D du fichier

- Exemple ligne 8 : BOSS JOUR LUMINEUSE EDP VAPO 30 ML Nombre de produit par carton : pr_pack → Colonne E du fichier

Code barre : pr codebarre → Colonne H

<u>Prix achat</u>: **pr_prac** → Colonne F <u>Prix de vente</u>: **pr_prix** → Colonne G

<u>Modèle</u>: **pr_modele** → ligne bleue, rose, orange (qui est en fait une désignation d'un modèle, comprenant la marque)

 Les marques (exemple : Hugo Boss), seront intégrées dans une autre table de la base de données, qui est aussi à créer par vos soins

<u>Les codes produits</u> (*champ* **pr_cd_pr** *de la table ci-dessus*) seront générés et auto-incrémentés dans l'ordre chronologique à partir de 1 000 000.

ATTENTION: Dans le cadre de cet exercice certains champs de la table (BDD) ne sont pas exploités.

Il vous est demandé de passer par un fichier **CSV** pour le traitement en **Python ou Langage C** (selon modalité de rattrapage, voir avec votre pilote).

De plus, les erreurs devront être affichées / stockées avant la création du fichier SQL.

Le script SQL à produire sera sous la forme suivante :

>> Ceci est un exemple, les données sont dynamiques et dépendent du contenu du fichier Excel.

Naturellement, vous ne devez pas ajouter les lignes vides (du fichier Excel) dans la base de données!

Déroulé des actions à réaliser

- 1. Prendre connaissance du fichier Excel et se l'approprier
 - Il peut être nécessaire de le modifier s'il comporte des erreurs
- 2. Créer la base de données avec le SGBD de votre choix
- 3. Créer la table en vous appuyant sur la table d'exemple (image) en début de ce document
- 4. Créer une table « modele_marque » permettant de lier les modèles (ligne Bleue, Orange, Rouge fichier Excel) à la marque (ligne noire du fichier Excel)
 - Le but étant de pouvoir lier facilement les modèles à leur marque
- 5. Convertir le fichier Excel joint en fichier CSV « exploitable »
- 6. Récupérer, lire et traiter le fichier CSV avec le langage identifié (Python ou Langage C)
- 7. Insérer les données dans la table « produit »
- 8. Logger les erreurs dans un fichier text

Le pilote se tient à votre disposition pour vous aider dans la compréhension de cet exercice.

Naturellement, l'exercice est à réaliser SEUL.