# ----- Projet BDD Fast-Food ------

#

# 1. Extraire le nombre d’employés dans le département du 64

# (Colonne 1 : nombre d’employés)

SELECT COUNT(Postes.ind\_id)

FROM Postes

JOIN Restaurants r ON r.id = Postes.resto\_id

WHERE r.code\_postal LIKE '64%';

# 2. Extraire l’ingrédient le plus utilisé dans chaque département.

# (Colonne 1 : département, Colonne 2 : ingrédient, Colonne 3 : Utilisation ingrédient)

SELECT results.dept, results.ingredient, MAX(total\_count)

FROM

(SELECT SUBSTRING(r.code\_postal , 1, 2) AS dept, ing.nom as ingredient, COUNT(\*) AS total\_count

FROM Restaurants r

JOIN Orders o ON o.resto\_id = r.id

JOIN Orders\_items oi ON o.id = oi.order\_id AND ((oi.item\_id > 13) OR (oi.item\_id < 8)) #Select plats only

LEFT JOIN Menus m ON m.id = oi.item\_id

LEFT JOIN Recettes rec ON rec.item\_id = oi.item\_id  OR rec.item\_id = m.plat

JOIN Ingredients ing ON rec.ing\_id = ing.id

GROUP BY dept, ing.nom

ORDER BY dept, total\_count DESC) AS results

GROUP BY results.dept;

# 3. Extraire le restaurant ayant la masse salariale la moins importante

# (somme des salaires) (Colonne 1 : Restaurant, Colonne 2 : masse salariale)

SELECT postes.resto\_id, SUM(salaire)

FROM salairesnotes, postes

GROUP BY resto\_id

ORDER BY SUM(salaire) ASC

LIMIT 1;

# 4. Extraire le plat le moins vendu entre 00h00 et 11h59 pour chaque restaurant.

# (Colonne 1 : Restaurant, colonne 2 : plat, colonne 3 : quantité)

**select** orders.resto\_id, items.nom **as** plat, Orders\_items.quantite

**FROM** orders

**JOIN** orders\_items **ON** orders.id = orders\_items.order\_id

**join** items **on** items.id = orders\_items.item\_id

**WHERE** **extract**(**hour** **from** Orders.date\_heure) **BETWEEN** 0 **AND** 12 **and** items.type\_item='plat'

**GROUP** **BY** orders.resto\_id

**ORDER** **BY** resto\_id;

# 5. Extraire les restaurants n’ayant pas encore ouvert

# (sans employés). (Colonne 1 : Restaurant)

SELECT resto\_id, COUNT(ind\_id) as emp

FROM postes

HAVING emp < 1;

# 6. Extraire les stocks du restaurant du 64200 avec une colonne intitulée « à commander »

# qui aura comme valeur (Colonne 1 : Ingredient, Colonne 2 : A commander) :

# • Si l’ingrédient a plus de 1000 unités en stock, le statut est « OK ».

# • Si l’ingrédient a entre 750 et 999 unités en stock, le statut est « en stock ».

# • Si l’ingrédient a entre 500 et 747 unités en stock, le statut est « à commander : Pas d’urgence ».

# • Si l’ingrédient a entre 250 et 499 unités en stock, le statut est « à commander : urgence ».

# • Si l’ingrédient a entre 1 et 249 unités en stock, le statut est « à commander : prioritaire ».

# • Si l’ingrédient n’a pas de stock, le statut est : « à commander : Rupture de stock ».

—-- pas besoin d'ajouter à la table, peut simplement le faire à partir de la déclaration SELECT finale—--

alter table stocks

add status varchar(300);

update stocks

set status  = (case

when Stocks.unites < 1 then 'à commander : Rupture de stock'

when Stocks.unites < 249 then 'à commander : prioritaire'

when Stocks.unites < 499 then 'à commander : urgence'

when Stocks.unites < 747 then 'à commander : Pas d’urgence'

when Stocks.unites < 999 then 'en stock'

ELSE 'OK'

end );

SELECT ingredients.nom as Ingredient, stocks.status as 'à commander'

FROM stocks

JOIN restaurants ON Restaurants.id = Stocks.rest\_id

join ingredients on ingredients.id = stocks.ing\_id

WHERE Restaurants.code\_postal = '64200' and stocks.unites<748

GROUP BY Stocks.ing\_id;

# 7. Extraire le bénéfice moyen de chaque pays sur l’ensemble des produits vendus

# (bénéfice = prix de vente d’un item – prix d’achat). (Colonne 1 : Pays, Colonne 2 : Benefmoyen)

SELECT Pays.pays,AVG((Orders\_items.quantite\*Items.prix)-(Recettes.unites\*Ingredients.prix)) as Benefmoyen

FROM Orders\_items

JOIN Items ON Orders\_items.item\_id = Items.id

join recettes on recettes.item\_id = Items.id

join ingredients on ingredients.id = recettes.ing\_id

join cartes on cartes.item\_id = items.id

join pays on pays.id = cartes.pays\_id

GROUP BY Pays.pays;

# 8. Extraire le top 10 des restaurants payant le mieux leurs employés ayant plus de deux ans d’expérience.

# (Colonne 1 : restaurant, colonne 2 : salaireMoyen)

**SELECT** Restaurants.id **as** restaurant, **AVG**(salairesnotes.salaire) **as** salaireMoyen

**FROM** personnel

**join** salairesnotes **on** salairesnotes.ind\_id = personnel.id

**join** postes **on** postes.ind\_id = personnel.id

**JOIN** Restaurants **ON** postes.resto\_id = Restaurants.id

**WHERE** Personnel.experience > 2

**GROUP** **BY** Restaurants.id

**ORDER** **BY** **AVG**(salairesnotes.salaire) **DESC**

**LIMIT** 10;

# 9.Extraire le nombre moyen de ventes réalisé pour chaque heure par un employé ayant moins d’1 an

# d’expérience (Colonne 1 : Heure, Colonne 2 : nbventes)

**SELECT**  **extract**(**hour** **from** Orders.date\_heure) **as** heure, **count**(Orders.id) **as** nb\_vente

**FROM** Orders

**join** Personnel **on** personnel.id = orders.vendeur\_id

**WHERE** Personnel.experience < 1.1

**GROUP** **BY** **extract**(**hour** **from** Orders.date\_heure) **ASC**

# 10. Extraire le chiffre d’affaire moyen des restaurants ayant un espace pour enfant et celui des

# restaurants n’en ayant pas. (Colonne 1 : typeRestaurant, Colonne 2 : CAmoyen)

**SELECT** r.espace\_enfant **as** espace\_enfant, **SUM**(o.somme)/r.id **AS** CAmoyen **FROM** Orders o **LEFT** **JOIN** Restaurants r

**ON** o.resto\_id = r.id

**GROUP** **BY** r.espace\_enfant ;

# 11. Extraire la moyenne des notes attribuées pour chaque manager aux employés sous leurs ordres

# (Colonne 1 : manager, Colonne 2 : notemoyenne).

**SELECT**  p2.id **as** manager, **AVG**(sn.note) **as** notemoyenne

**FROM** Postes p

**INNER** **JOIN** Personnel p2 **ON** p.id = p2.id

**INNER** **JOIN** SalairesNotes sn **ON** p2.id = sn.ind\_id

**WHERE** p.poste ='manager'

**GROUP** **BY** p2.id ;

# 12. Lister l’ensemble des menus pour la France et l’Espagne

# (Colonne 1 : Pays, Colonne 2 : Nom menu, Colonne 3 : plat, Colonne 4 : boisson, Colonne 5 : dessert)

**SELECT** mm.pays, mm.menu, i2.nom **AS** plat, i3.nom **AS** boisson, i4.nom **AS** dessert

**FROM**

(**SELECT** p.pays **as** pays, i.nom **AS** menu, m.plat **AS** plat, m.boisson **AS** boisson, m.dessert **AS** dessert

**FROM** Cartes c

**JOIN** Pays p **ON** c.pays\_id = p.id

**JOIN** Items i **ON** i.id = c.item\_id

**JOIN** Menus m **ON** c.item\_id = m.id

**WHERE** (c.pays\_id = 1 **OR** c.pays\_id = 2) **AND** i.type\_item = 'menu') **AS** mm

**JOIN** Items i2 **ON** i2.id = mm.plat

**JOIN** Items i3 **ON** i3.id = mm.dessert

**JOIN** Items i4 **ON** i4.id = mm.boisson;