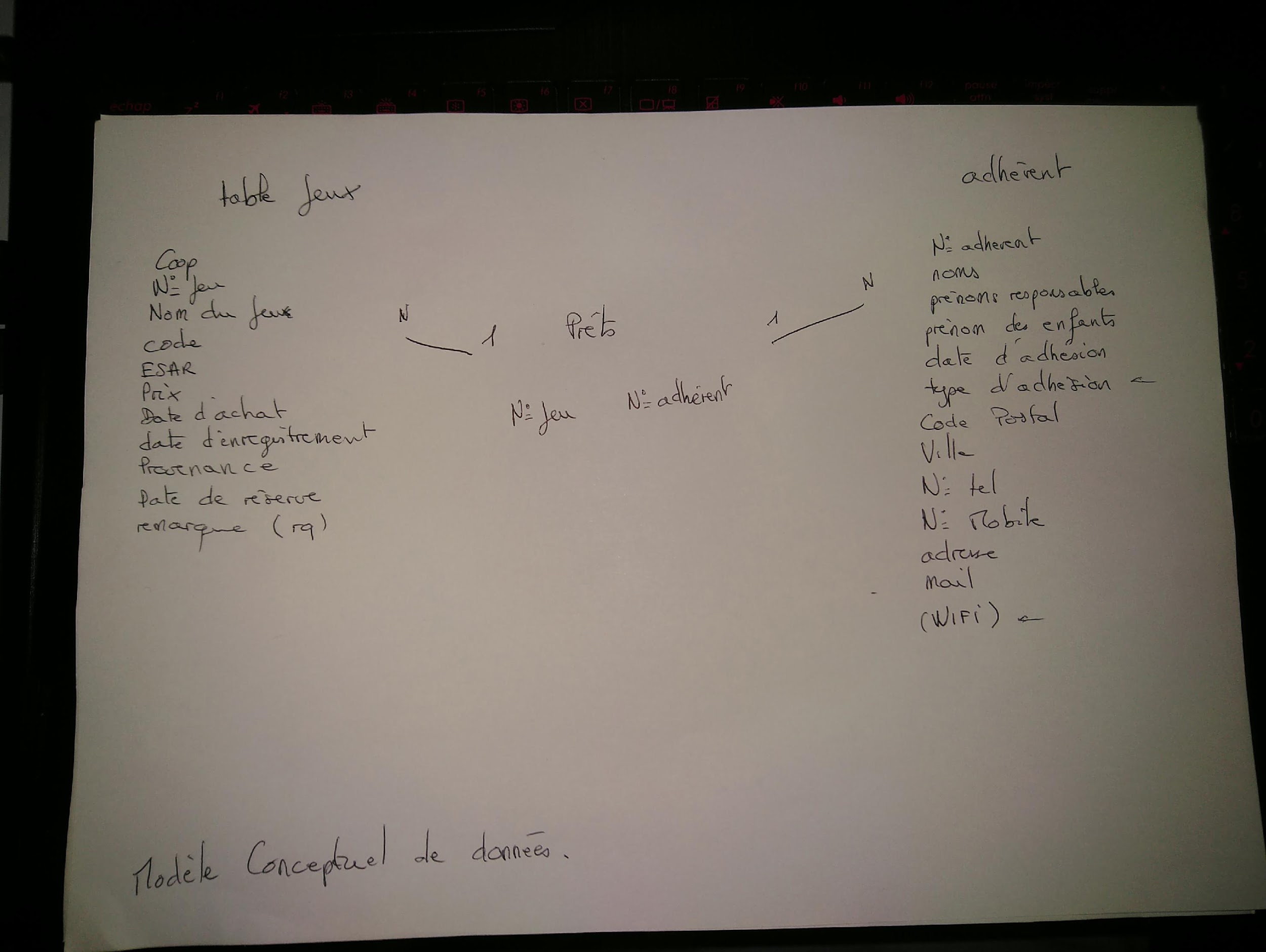
**Modèle Conceptuel de Données MCD** (les requêtes sql ne sont pas vérifiées et peuvent contenir des erreurs à cette étape du projet)

J'ai récolté, auprès de la Cité des Jeux, les données pour réaliser le modèle conceptuel de données. La gestion actuelle est réalisée avec Windows Excel mais la stabilité du logiciel rompt ça table de liaison pour des problèmes de mise à jour selon les dires. La cité des Jeux a en projet la commande d'un logiciel sur mesure pour 600 euros par ans. Je leur ai conseillé d'attendre la fin de mon stage pour envisager prendre une décision car l'applicatif que je réalise prendra la place du logiciel actuel.

Première étape, le mcd Excel



Après lecture des éléments, il y a de nombreuses lacunes. Je dois y ajouté la date de prêt, le prix de la location et identifier les champs à choix multiples comme par exemple : le type d'adhésion.

Pour optimiser le MCD, il semblerait qu'il soit opportun de séparer la gestion de stocks des jeux (coop, code, date d'achat, prix, date d'enregistrement, provenance, date de réserve, remarque) de la gestion de prêts des jeux. Cette optimisation s'appliquera aussi sur la gestion des adhérents dont leurs infos personnelles se doivent de rester sous contrôle (prénoms responsables, prénom des enfants, date d'adhésion, type d'adhésion, CP, Ville, tel, adresse, wifi. Je rajoute un champ alias pour une connexion plus privative.

Deux tables indispensables se profilent, la table ESAR (catégorie du jeu) et la table des prix(1,30€, 1,50€, 5€). La formule de comptabilité est la suivante.

Pour chaque numéro de prêt où la date prêt se situe entre la date de début de comptabilité et la date de fin de comptabilité.

Additionner le prix au prix total.

**CALCUL DE COMPTABILITE POUR UNE PERIODE DONNEES**

$rows = SELECT prix FROM prets

WHERE date\_deb\_compta<date\_pret AND date\_pret<date\_fin\_compta

Example : **select prix from prets where '2018-5-11' < date\_pret AND date\_pret <= '2018-5-21'**

$total\_compta=0

for each $rows {

$total\_compta+=prix

}

{{total\_compta}} euros pour la période du date\_deb\_compta à date\_fin\_compta

ou mieux :

$tot\_prix13=0

$tot\_prix15=0

$tot\_prix5=0

$total\_compta=0

for each $rows {

$total\_compta+=prix

if prix===1.3

$tot\_prix13+=prix

else if prix===1.5

$tot\_prix15+=prix

elseif prix===5

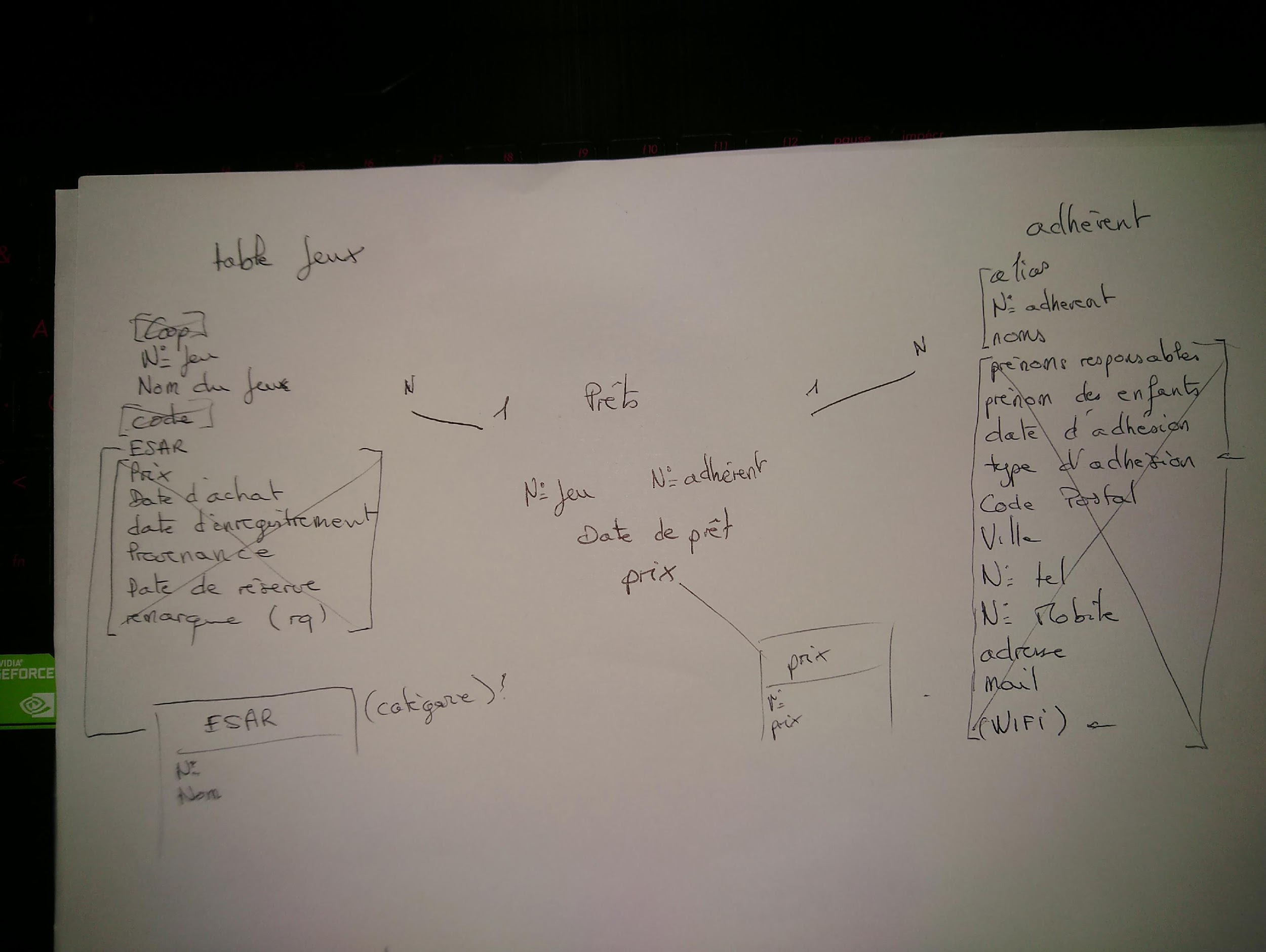
$tot\_prix5+=prix

}

{{tot\_prix1.3}} euros de prêts à 1,30 euros

{{tot\_prix1.5}} euros de prêts à 1,50 euros

{{tot\_prix5}} euros de prêts à 5,00 euros

{{total\_compta}} euros pour la période du date\_deb\_compta à date\_fin\_compta

Une discussion avec le staff s'impose pour valider si la comptabilité se fait sur la date de prêts ou la date de fin de prêts soit la date de prêt plus trois semaines. Soit :

$rows = SELECT prix, date\_deb\_pret FROM Prets

WHERE date\_deb\_compta<date\_deb\_pret+3semaines<date\_fin\_compta

Pour la partie emprunts, la formule est la suivante :

SELECT jeux.num\_jeu, nom\_du\_jeu, ESAR, prets.num\_jeu, date pret FROM jeux, prets

WHERE jeux.num\_jeu != prets.num\_jeu && date\_pret+3semaines<=date\_du\_jour

SELECT jeux.num\_jeu, nom\_du\_jeu, ESAR FROM jeux INNER JOIN prets

WHERE date\_pret+3semaines<=date\_du\_jour

Cette requète ne gère pas les retards

Je rajoute donc un champ retour dans la TABLE prets.

Et j'obtiens :

SELECT jeux.num\_jeu, nom\_du\_jeu, ESAR, prets.num\_jeu, date prêt, retour

FROM jeux, prets

WHERE jeux.num\_jeu != prets.num\_jeu && retour==false

Ce qui oblige cependant une manipulation lors du retour du produits. Cet élément fait partie d'une question à poser au staff.

Si un tableau de retour papier est utilisé il est nécessaire de concevoir aussi un listing des prêts pour une période données dont la formule est la suivante :

**LISTINGS DE TOUS LES PRETS POUR UNE PERIODE DONNEES**

SELECT jeux.num\_jeu, nom\_du\_jeu, ESAR, prets.num\_jeu, date pret FROM jeux, prets

WHERE jeux.num\_jeu != prets.num\_jeu

&& date\_deb<date\_pret+3semaines<date\_fin

On a donc la date de retour pour chaque jeu sur la période données.

La solution la plus optimale me sempble être de ne pas utilser un champ retour mais l'affichage des retours par période pour vérifier ainsi si un jeu a été sorti.

Pour la partie adhérent, la formule à utiliser est la suivante :

**LISTE DES PRETS EN COURS POUR 1 ADHERENT**

SELECT jeux.num\_jeu, nom\_du\_jeu, ESAR, prets.num\_jeu, date prêt, prets.num\_adherent, adherents.num\_adherent

FROM jeux, prets, adherents

WHERE prets.num\_adherent===$num\_adherent

&& prets.num\_jeu===jeux.num\_jeu

Les autres requètes SQL sont :

* + l'ajout de surcatégories

INSERT INTO `surcategorie` (`nom\_surcat`) VALUES ('Jeux d\'Exercice'), ('Jeux Symbolyques');

* + l'ajout d'une catégorie

INSERT INTO `categorie` (`nom\_categorie`, `nom\_surcat`) VALUES ('Jeu sensoriel sonore', 'Jeux d\'Exercice'), ('Jeu sensoriel visuel', 'Jeux d\'Exercice');

* + l'ajout/suppression d'un jeu

INSERT INTO `jeux` (`num\_jeu`, `nom\_jeu`, `nom\_categorie`) VALUES ('5', 'cinq', 'Jeu sensoriel sonore'), ('6', 'six', 'Jeu sensoriel visuel'), ('7', 'sept', 'Jeu sensoriel visuel'), ('8', 'huit', 'Jeu sensoriel sonore'), ('9', 'neuf', 'Jeu sensoriel visuel'), ('10', 'dix', 'Jeu sensoriel visuel'), ('11', 'onze', 'Jeu sensoriel visuel'), ('12', 'douze', 'Jeu sensoriel sonore'), ('13', 'treize', 'Jeu sensoriel visuel'), ('14', 'quatorze', 'Jeu sensoriel visuel');

DELETE FROM jeux WHERE num\_jeu===$num\_jeu

* + l'ajout/suppression d'un prix
  + l'ajout/suppression d'un adhérent

INSERT INTO `adherents` (`num\_adherent`, `noms\_adherent`, `alias`)

VALUES ('$num\_adherent','$noms\_adherent','$alias')

DELETE FROM adherents WHERE num\_adherent===$num\_adherent

* + la création d'un prêt

INSERT INTO prets VALUES ('$num\_jeu','$num\_adherent','$date\_pret','$prix')

Après discussion avec l'équipe, le choix de la gestion des prêts est une remise en base de données automatique ; la gestionnaire des prêts (Cécilia) gère les prêts avec les adhérents (rentrée en base cf création d'un prêt), la recherche de disponibilité du produit se fera donc sur la liste de la TABLE prêts pour tous les jeux dont la date actuelle n'appartient pas à la période de prêt de ceux prêtés.

Pour la liste des jeux disponibles, la formule est donc la suivante :

SELECT jeux.num\_jeu, nom\_du\_jeu, ESAR, prets.num\_jeu, date pret FROM jeux, prets

WHERE jeux.num\_jeu != prets.num\_jeu && date\_pret+3semaines>=date\_du\_jour

SELECT \* FROM jeux OUTER JOIN prets WHERE date\_pret+3semaines<=date\_du\_jour

SELECT jeux.num\_jeu, nom\_du\_jeu, ESAR, prets.num\_jeu, date pret FROM jeux

SELECT id, prenom, nom, utilisateur\_id FROM utilisateur LEFT JOIN commande ON utilisateur.id = commande.utilisateur\_id WHERE utilisateur\_id IS NULL

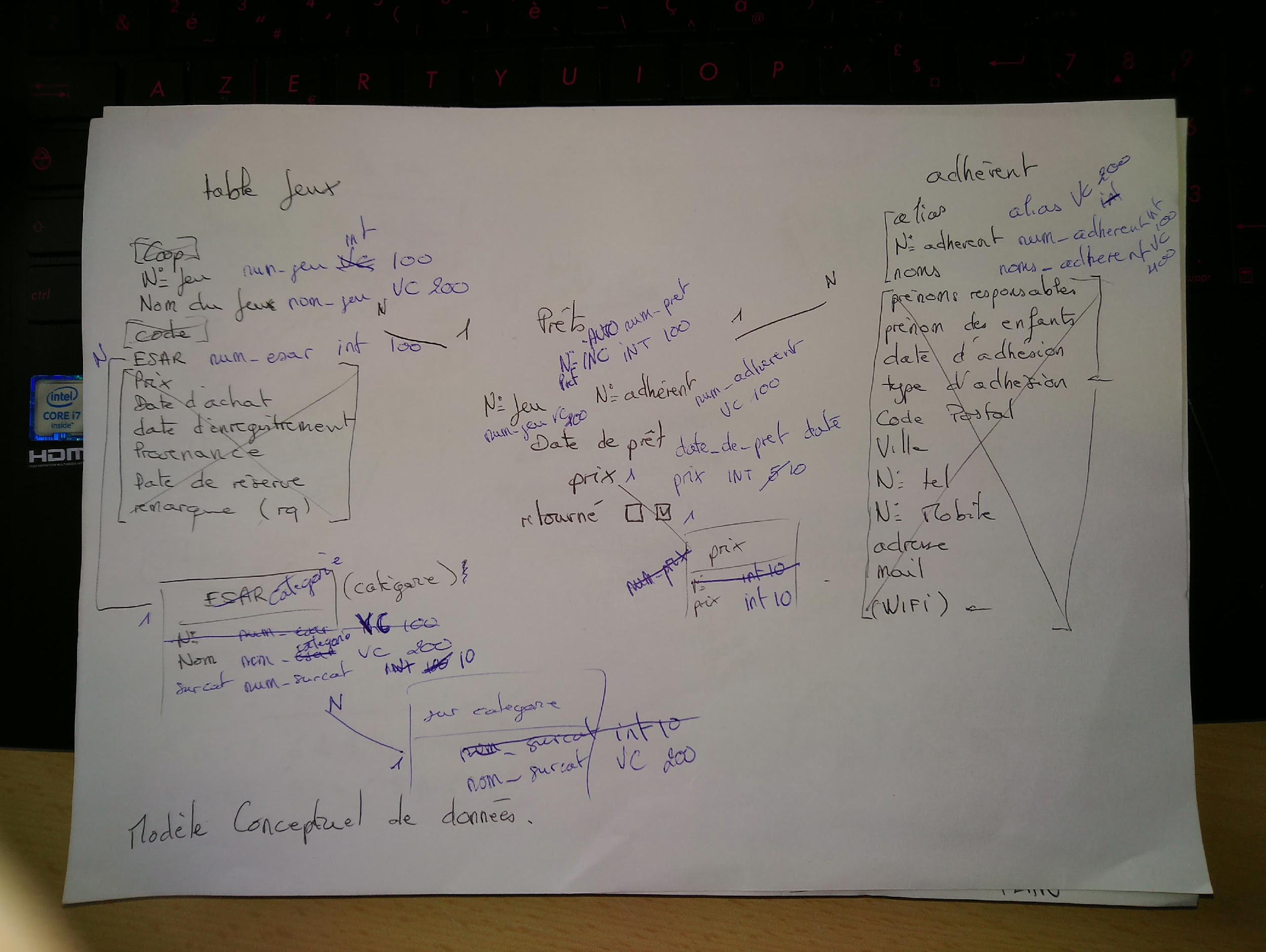
SELECT start\_date, end\_date, CASE WHEN start\_date <= '2015-08-31' THEN 'true' ELSE 'false' END AS started\_before\_end\_of\_month, CASE WHEN NOT end\_date <= '2015-08-01' THEN 'true' ELSE 'false' END AS did\_not\_end\_before\_begining\_of\_month FROM product\_sales WHERE start\_date <= '2015-08-31' AND end\_date >= '2015-08-01' ORDER BY start\_date;

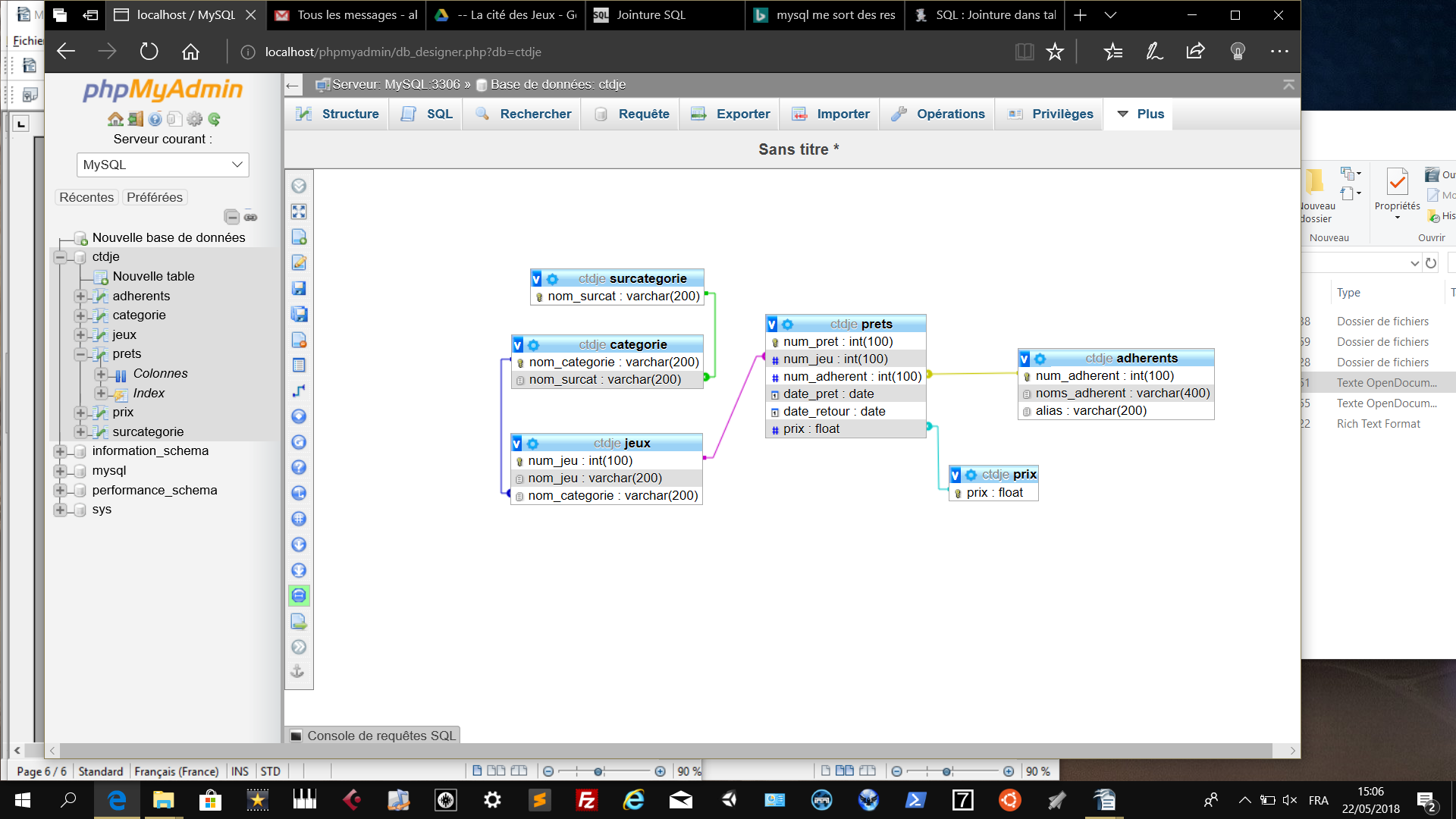
<https://stackoverflow.com/questions/14208958/select-data-from-date-range-between-two-dates>

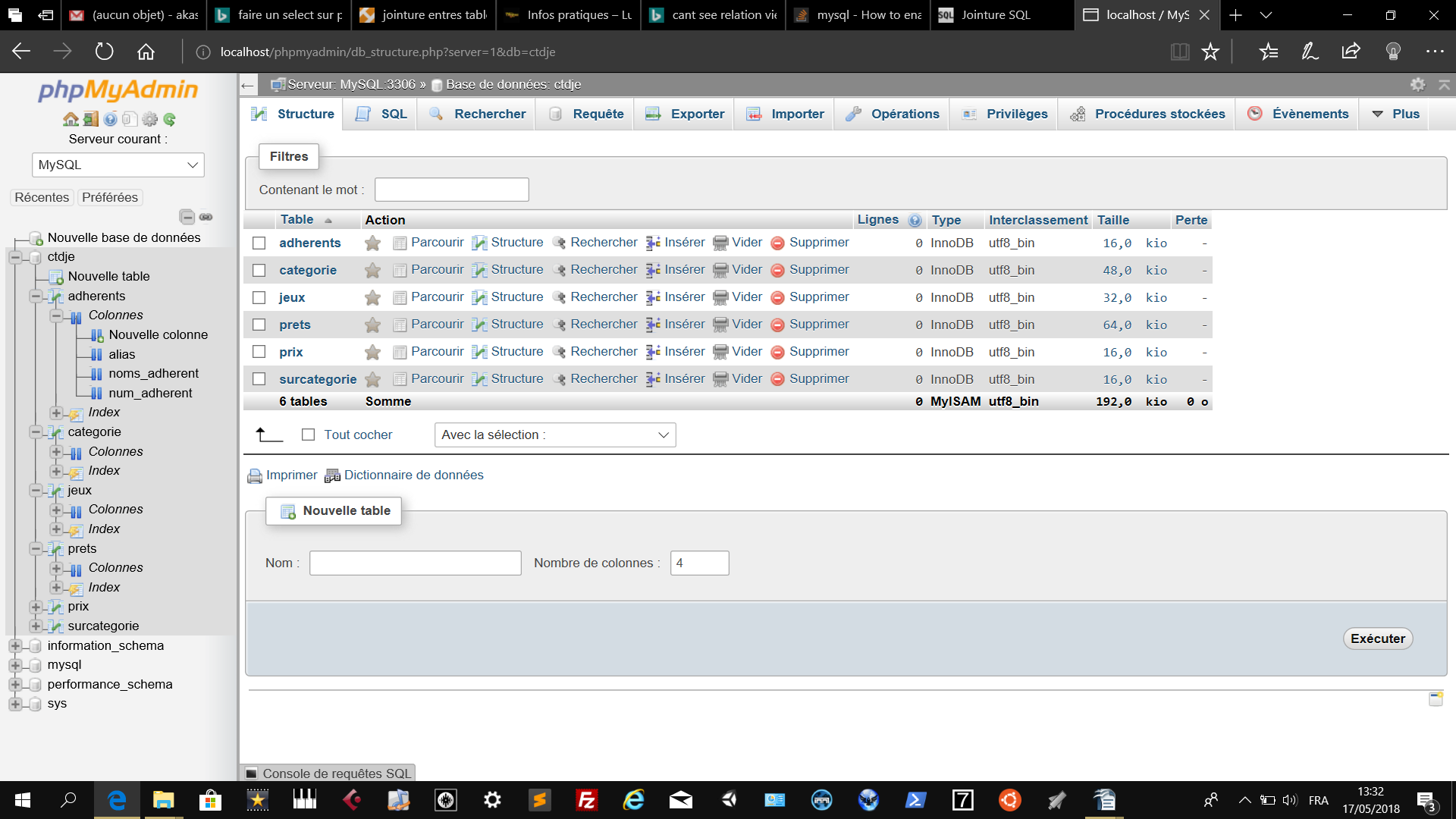
INSERT INTO tbl(fromdate,todate) values (now(), DATE\_ADD(now(),INTERVAL 2 WEEK))

https://stackoverflow.com/questions/19925400/add-2-weeks-to-a-date-sql

Après quelques heures de modifications (~2 heures)

Je peux enfin vous présenter le MCD de test auprès duquel je vais valider mes requêtes SQL ci-dessus. Mais j'ai besoin de rajouter un champ date\_retour dans la TABLE prets pour prendre en compte les retours décalés, prolongés pour des raisons de congés, jours fériés ou dates exceptionnels.





cf. <http://sql.sh/>

**Les technologies utilisées**

Vous avez sûrement remarqué la présence des technologies depuis le début de ma rédaction, pour les nommer maintenant :

1- Le SQL (SELECT)

2- Le PHP ($rows)

3- Angular ( {{tot\_compta}} )