

Projet SQL

Présenté par Nicolas TORO & Guillaume JACQUEMIN en TG09

Créateur de BASIQ-FAT

The logo for BASIQ-FAT is displayed on a solid orange rectangular background. The text 'BASIQ-FAT' is rendered in a bold, sans-serif font. 'BASIQ' is white with a small starburst graphic above the 'A', and '-FAT' is dark grey. The entire logo is slightly tilted to the right.

***Rejoignez BASIQ-FAT et vous ne
ressemblerez plus à une patate !***

Présentation

Nous sommes BASIQ-FAT, une jeune start-up de salle de sport dans le but rendre maigre les personnes en surpoids. Nous avons donc fait une base de données contenant nos clients, nos employés, nos abonnements, etc.

Sommaire

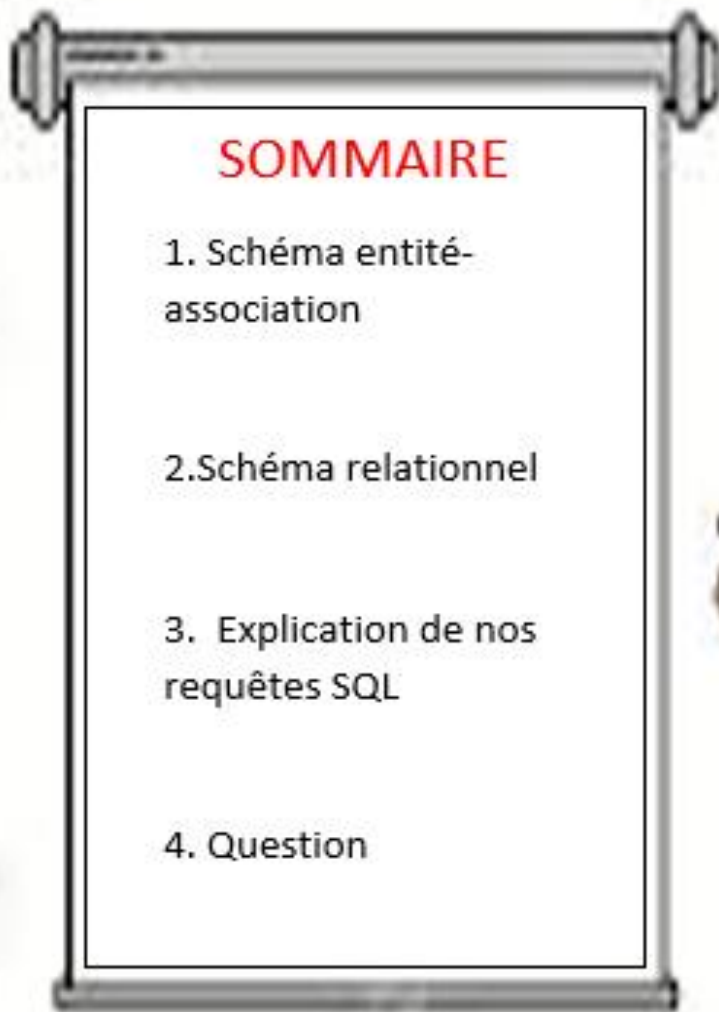


Schéma Entité-association

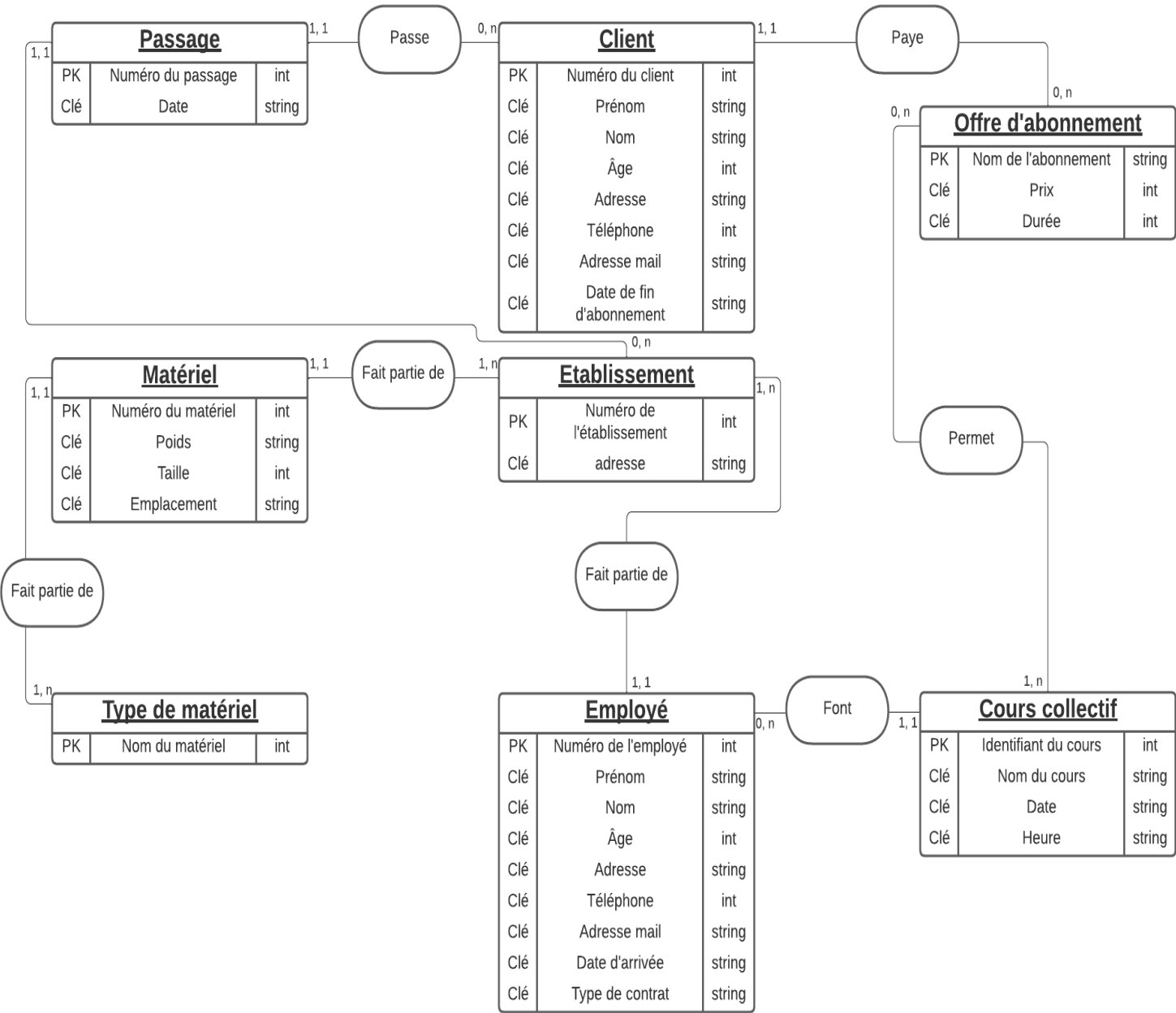
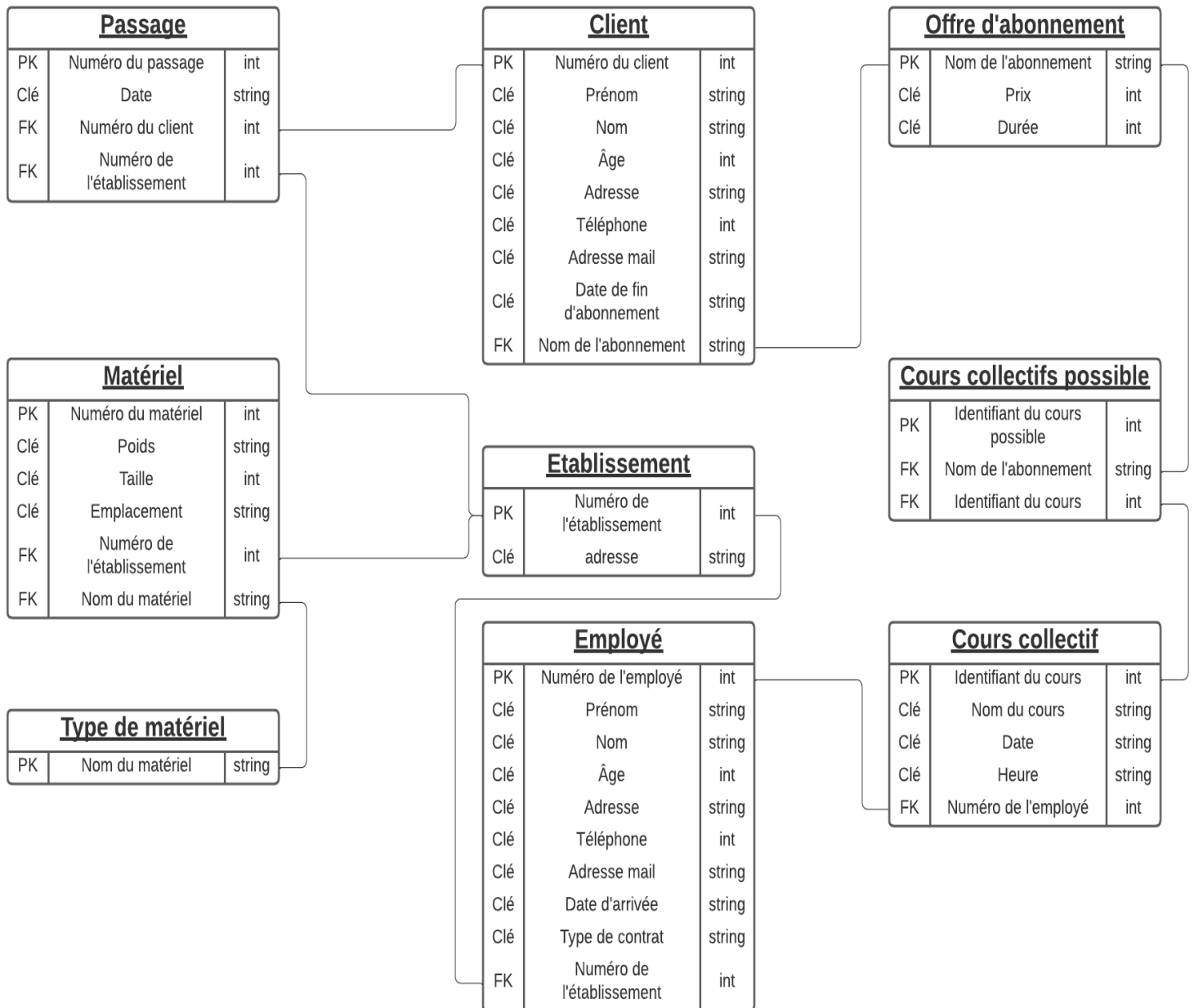


Schéma relationnel



Requêtes SQL

Mise en situation :

Grâce à plusieurs étapes, vous allez voir de nombreux exemples dans le but d'expliquer certaines commandes liées à notre base de données. Les commentaires pour détailler le code seront en italique et débuterons par des « -- ». Ici, vous allez vous glissez dans la peau du directeur des abonnements de BASIQ-FAT :

Etape 1 :

Nous souhaitons créer un nouvel abonnement pour le black Friday. Voici notre commande :

```
INSERT INTO "offre_abonnement" --Insérer dans « offre_abonnement » Les valeurs
VALUES (
    "abonnement black friday", --Nom_de L'abonnement
    10, --Prix
    1 --Durée
);
```

Ensuite, une idée nous vient en tête. Le mois de décembre et la période de Noël arrive, c'est donc le moment de faire un abonnement pour Noël. Il aura les mêmes spécificités que l'abonnement « BASIQ » mais sera moins chère et ne se sera achetable que pendant le mois de décembre 2022. Voici notre commande :

```
INSERT INTO "offre_abonnement" --Insérer dans « offre_abonnement » Les valeurs
VALUES (
    "NOEL 2022 1", --Nom_de L'abonnement
    15, --Prix
    1 --Durée
);
```

```
INSERT INTO "offre_abonnement" --Insérer dans « offre_abonnement » Les valeurs
VALUES (
    "NOEL 2022 2", --Nom_de L'abonnement
    40, --Prix
    3 --Durée
);
```

```
INSERT INTO "offre_abonnement" --Insérer dans « offre_abonnement » Les valeurs
VALUES (
    "NOEL 2022 3", --Nom_de L'abonnement
    160, --Prix
    12 --Durée
```

);

Etape 2 :

Par inadvertance, nous nous rendons compte que l'abonnement du black Friday ne nous plait pas, nous voulons donc le modifier (le nom, le prix et la durée). Voici notre commande :

```
UPDATE offre_abonnement --Mettre à jour « offre abonnement »
SET
    nom_abonnement = "BLACK FRIDAY 2021", --Nouveau nom
    prix = 30, --Nouveau prix
    duree = 3 --Nouvelle durée
WHERE
    nom_abonnement = "abonnement black friday"; --Faire nos changement sur l'abonnement «
abonnement black Friday »
```

Etape 3 :

L'année dernière, en début décembre 2021, nous avons créé un abonnement spécial Noël 2021 valable jusqu'au 1er décembre 2022 pour toutes personnes. Nous souhaitons donc le supprimer aujourd'hui. Voici notre commande :

```
DELETE FROM "offre_abonnement" --Supprimer quelque chose dans « offre_abonnement »
WHERE nom_abonnement = "NOEL 2021"; --Supprimer l'abonnement « NOEL 2021 »
```

Etape 4 :

Maintenant que nous avons fait nos ajouts et modifications dans notre base de données, nous allons afficher certains choses.

Premièrement, nous désirons afficher tous les abonnements du plus petit au plus grand prix. Voici notre commande :

```
SELECT * FROM offre_abonnement ORDER BY "prix" --Sélectionner tout dans « offre_abonnement »
dans l'ordre du prix
```

Puis, nous allons afficher tous les abonnements des clients. Voici notre commande :

```
SELECT "nom_abonnement" FROM client --Afficher la colonne « nom_abonnement » de la table
« client »
```

Etape 5 :

Nous aimerions maintenant avoir des résultats plus précis en faisant des recherches plus précises.

Tout d'abords, nous souhaitons savoir quels abonnements dure 1 mois. Voici notre commande :

```
SELECT "nom_abonnement" --Sélectionner les éléments de la colonne « nom_abonnement »  
FROM "offre_abonnement" --Dans la table « offre_abonnement »  
WHERE duree = 1; --Quand la durée vaut 1
```

Maintenant, nous allons faire une recherche pour notre ami Mathieu. Il veut devenir client mais ne dispose que de 150€ pour payer son abonnement. Il souhaite donc savoir quelle sont les abonnements (avec leur prix et leur durée) qui sont en dessous de 150€. Voici la commande :

```
SELECT * --Sélectionner tous les éléments de la colonne « nom_abonnement »  
FROM "offre_abonnement" --Dans la table « offre_abonnement »  
WHERE prix < 150; --Quand le « prix » est en dessous de 150€
```

Etape 6 :

A BASIQ-FAT, il y a de nombreux cours et ce ne sont pas les mêmes pour chaque établissement ainsi que pour les employés. On souhaite afficher les cours en fonction des employés et de leurs adresses. Cela nous permettra de voir la proportion de chaque cours en fonction du lieu. Voici notre commande :

```
SELECT cours_collectif.nom_du_cours, employe.prenom, employe.nom, employe.adresse --  
Sélectionner tous les éléments de la colonne « nom_du_cours » (dans « cours_collectif ») et  
« prenom », « nom » et « adresse » (dans « employe »)  
FROM employe --Dans la table « employe »  
INNER JOIN cours_collectif --Rejoindre la table « cours_collectif »  
ON employe.numero_employe = cours_collectif.numero_employe --« numero_employe » (dans  
« employe ») vaut « numero_employe » (dans « cours_collectif »)  
ORDER BY employe.adresse --Dans l'ordre de l'adresse des employés
```

Notre amis Mathieu cherche un matériel précis qui ne se trouve pas dans toutes les salles de sport. Il nous a donc demander de lui montrer la liste du matériel de chaque établissement dans l'ordre des établissements, des emplacements (salles) et du nom du matériel. Voici notre commande :

```
SELECT materiel.numero_etablissement, materiel.emplacement, materiel.numero_du_materiel,  
type_de_materiel.nom_du_materiel --Sélectionner tous les éléments de la colonne  
« nom_du_materiel » (dans « type_de_materiel ») et « numéro_etablissement », « emplacement »  
et « numero_du_materiel » (dans « materiel »)  
FROM materiel --Dans la table « materiel »  
INNER JOIN type_de_materiel --Rejoindre la table « type_de_materiel »  
ON materiel.nom_du_materiel = type_de_materiel.nom_du_materiel --« nom_du_materiel » (dans  
« materiel ») vaut « nom_du_materiel » (dans type_de_materiel »)  
ORDER BY materiel.numero_etablissement, materiel.emplacement, materiel.nom_du_materiel --Dans  
l'ordre du numéro de l'établissement, de l'emplacement du matériel et du nom du matériel
```

Etape 7 :

Nous nous apercevons que certains BASIQ-FAT enregistre plus de passage que les autres. Dans le but de savoir quelle est la ville qui obtient le plus de passage, nous allons faire une liste des 20 premiers passages en fonctions des établissements. Voici notre commande :

```
SELECT etablisement.adresse, passage.numero_du_client --Sélectionner Les éléments de la
colonne « adresse » (dans « etablisement ») et « numero_du_client » (dans « passage »)

FROM passage --Dans La table « passage »

INNER JOIN etablisement --Rejoindre la table « etablisement »

ON passage.numero_etablisement = etablisement.numero_etablisement --« numero_etablisement
» (dans « passage ») vaut « numero_etablisement » (dans « etablisement »)

WHERE passage.numero_du_passage <= 20 --Quand Le numéro du passage est inférieur ou égale à
20

ORDER BY etablisement.adresse, passage.numero_du_client --Dans l'ordre de l'adresse des
établissement et des numéros des clients
```

On cherche maintenant à offrir une récompense à notre meilleur client. On souhaite donc savoir quel client à fait le plus de passage avant la rentrée scolaire (début septembre 2022). Voici notre commande :

```
SELECT client.prenom, client.nom, passage.numero_du_passage, passage.date,
passage.numero_etablisement --Sélectionner Les éléments de la colonne « prenom » et « nom »
(dans « client ») et « numero_du_passage », « date » et « numero_etablisement » (dans
« passage »)

FROM passage --Dans La table « passage »

INNER JOIN client --Rejoindre la table « client »

ON passage.numero_du_client = client.numero_du_client --« numero_du_client » (dans
« passage ») vaut « numero_du_client » (dans « client »)

WHERE passage.date < "22/09/01" --Quand La date du passage est avant Le 1 septembre 2022

ORDER BY passage.numero_du_client, passage.numero_du_passage --Dans l'ordre des numéros des
clients et des numéros des passages
```

On s'intéresse enfin au type Contrat des établissements du sud de la France. Selon des études, on donnerait plus de CDD aux jeunes du sud de la France. Pour vérifier cela, nous allons afficher l'âge et le type de contrat des employés appartenant à des établissements du sud de la France. Voici notre commande :

```
SELECT employe.numero_employe, employe.age, employe.type_de_contrat,
employe.numero_etablisement, etablisement.adresse --Sélectionner Les éléments de la colonne
« adresse » (dans « etablisement ») et « numero_employe », « age », « type_de_contrat » et
« numero_etablisement » (dans « employe »)

FROM employe --Dans La table « employe »

INNER JOIN etablisement --Rejoindre la table « etablisement »

ON employe.numero_etablisement = etablisement.numero_etablisement --
« numero_etablisement » (dans « employe ») vaut « numero_etablisement » (dans «
etablisement »)
```



```
WHERE employe.numero_etablissement = 2 OR employe.numero_etablissement = 3 OR  
employe.numero_etablissement = 5 OR employe.numero_etablissement = 7 OR  
employe.numero_etablissement = 8 --Quand l'établissement est dans le sud de la France  
  
ORDER BY employe.type_de_contrat DESC, employe.numero_employe --Dans l'ordre inverse du type  
de contrat et du numéro des employés
```

Pour finir, comme les soldes arrivent bientôt, nous aimerions faire des réductions sur les abonnements pendant quelques jours. Nous choisissons de mettre une grosse réduction sur l'abonnement le moins choisis en fonction de la durée, une petite réduction sur le plus choisis en fonction de la durée et une réduction moyenne sur le dernier. A BASIQ-FAT, chaque abonnement classique existe en 3 fois en fonction de la durée et du prix et on sait que la plupart des gens intéressés par les soldes sont les personnes de moins de 25 ans. Nous allons donc afficher le nom de l'abonnement en fonction de chaque âge des clients inférieur ou égale à 25 ans. Voici notre commande :

```
SELECT offre_abonnement.nom_abonnement, client.prenom, client.nom, client.age --Sélectionner  
les éléments de la colonne « nom_abonnement » (dans « offre_abonnement ») et « prenom »,  
« nom » et « age » (dans « client »)  
  
FROM offre_abonnement --Dans la table « offre_abonnement »  
  
INNER JOIN client --Rejoindre la table « client »  
  
ON offre_abonnement.nom_abonnement = client.nom_abonnement --« nom_abonnement » (dans  
« offre_abonnement ») vaut « nom_abonnement » (dans « client »)  
  
WHERE client.age <= 25 --Quand l'âge des clients est inférieur ou égale à 25  
  
ORDER BY client.age --Dans l'ordre de l'âge des clients
```

Avez-vous des questions ?



(Si vous n'avez pas eu le temps de les poser pendant notre oral, vous pouvez nous en envoyer à help@basiq-fat.com)

**Merci d'avoir suivi notre présentation et
notre PDF sur BASIQ-FAT !**