

# BDD tuto 2 : Prise en compte d'un besoin supplémentaire et extension de la base de donnée

## Diapo 1

Maintenant qu'on en sait plus sur les tables et les relations qu'il peut y avoir entre elles, on va faire évoluer notre exemple de centre aéré pour prendre en compte les activités auxquelles les enfants participent.

Les animateurs prévoient d'organiser des sorties en dehors du centre aéré, par exemple ping-pong, piscine et cinéma, etc. Certaines de ces sorties ont un coût lié à la participation de l'enfant: le cinéma facture 6€ par participation à la séance, la piscine, c'est 6€ et le ping pong, c'est gratuit. Les enfants vont parfois à l'une ou l'autre des activités, et ce n'est pas toujours les mêmes groupes d'enfant, certains préférant parfois rester au centre, le lundi, Nicolas et Marie vont au cinéma, le mardi il y a Manon et Chloé qui vont à la piscine, le mercredi c'est ping-pong, et le jeudi, on refait un petit tour à la piscine

## Diapo2

Le centre aéré paye les entrées, comptant bien refacturer le montant au parent par la suite.

À chaque fin de semaine, il faudrait que le comptable fasse le bilan des factures à envoyer aux parents, en ajoutant les montants de la semaine passée aux coûts des participations aux activités de la semaine courante.

Cette opération est rébarbative et présente un risque d'erreur humaine. Comment faire pour que ce calcul soit automatique ?

## Diapo3

Déjà pour répondre à ce problème, il faut stocker dans notre base de donnée des informations qui n'y sont pas encore. Il faut déjà des informations sur les activités et notamment leur coût, on aura donc besoin d'une table avec un nom d'activité, et un coût. Voilà ce que ça donne avec un échantillon de données imaginé J'ai préféré mettre un matricule au cas où il y aie un doublon de nom (par exemple il peut y avoir Cinéma à 5€ et Cinéma à 6€ )

## Diapo4

On a trouvé qu'il fallait une nouvelle table, cherchons à établir une relation un à plusieurs avec une autre table de notre base de donnée. De toute évidence, c'est à l'enfant qu'on doit associer une activité, donc on va s'intéresser à la relation entre activité et enfant.

Supposons l'existence d'une relation un à un: un enfant doit pouvoir participer à une ou plusieurs activités, par exemple aller au cinéma le lundi, puis au ping pong le mardi, ce n'est déjà pas une relation de un à un

Ensuite, on peut vraisemblablement penser qu'une activité doit pouvoir accueillir plusieurs enfants. Si on combine cette information avec le fait qu'un enfant peut faire plusieurs activités, on voit ici qu'on n'est pas dans le cas d'une relation de un à plusieurs, mais d'une relation de plusieurs à plusieurs.

## Diapo5

Il nous faut donc une table intermédiaire, mais quelles sont les informations qu'on a oublié d'enregistrer?

Revenons au problème de départ, on veut calculer le montant total des activités auxquelles ont participé les enfants. Pour pouvoir faire le calcul de la facture, on a besoin d'enregistrer les informations de chaque participation à une activité dans une table.

## Diapo6

Comme le nombre de participations est indéterminé et ne dépend pas d'une autre donnée qu'on a déjà, il nous faut une table avec une ligne par participation d'un enfant à une activité. On va donner à cette table le nom de l'élément qu'on enregistre, c'est-à-dire la participation.

Quels attributs? Pour chaque participation, on va devoir préciser quel enfant est concerné, et à quelle activité il participe. En bonus, on peut également créer un attribut pour préciser la date de la participation, comme ça on pourra conserver une historique.

## Diapo7

Si on part sur ce principe, est-ce qu'on a bien des relations de un à plusieurs?

On va pouvoir associer un enfant à une ou plusieurs participations, tandis qu'une participation sera associée à un seul enfant.

## Diapo8

Du côté de l'activité: une participation sera associée à une seule activité tandis qu'une activité pourra être associée à plusieurs participations

On a donc ici deux relations un à plusieurs entre les trois tables. Le tour est joué.

## Diapo9

Ajoutons les colonnes clés à nos tables, la table enfant, on la connaît, et on a dit que la table activité devait contenir, un matricule, un nom et un coût.

Pour la table participation, on doit pouvoir associer un enfant, il faut donc une clé étrangère ici qui contiendra des matricules d'enfant.

Pour le lien entre l'activité et la participation, il faut une autre clé étrangère qui contiendra des matricules d'activité.

## Diapo10

Voici un aperçu de la table participation avec un échantillon de donnée, comme une table doit toujours avoir une clé primaire, même si ce ne sera pas forcément utile ici, on peut mettre une colonne matricule.

## Diapo11

En ajoutant les types de données adéquat, c'est-à-dire texte, date et nombre décimaux ici. Comme ça été vu précédemment, les clés primaires sont en nombre auto et les clés étrangères en nombre entier. Après avoir ajouter les relations, le schéma du modèle physique de notre base de donnée ressemble maintenant à ceci.

## Diapo12

On a créé la structure qui fallait, la preuve avec cette animation, imaginons qu'on veut savoir combien doit Mme Moutier cette semaine.

Le logiciel va d'abord chercher dans la table parent son matricule. Une fois trouvé, il suivra le lien d'intégrité référentielle pour rechercher la présence de ce matricule dans les clés étrangères de la table enfant. Dans notre cas Mme Moutier est en parent1 pour 2 enfants, et en parent 2 pour un

autre enfant. Il y a donc les enfants 26, 27,30. En suivant le lien, le logiciel va rechercher ensuite les participations aux activités de ces enfants.

Une fois qu'il aura trouvé le bilan des participations, il lui faudra suivre le lien pour aller chercher le prix de l'activité, ce qui va lui permettre de faire le calcul et de donner le résultat.

La requête présentée ici se résume à une seule question sur un seul parent, mais le logiciel peut aussi faire des requêtes qui vont remplir un tableau comme celui du comptable en une fraction de seconde.

## **Diapo13**

Il reste encore un point de détail à régler: Lorsqu'une telle requête sera créée pour afficher le tableau du comptable, l'attribut du montant dû de chaque parent est une valeur qui pourra être calculée.

Comme toute valeur calculée ne doit pas être considérée comme une donnée, on pourra supprimer la colonne des « montants dû » de la table parent. En effet le tableau du comptable pourra s'appuyer sur la requête qui calcule tout automatiquement. +

## **Diapo14**

On a réglé ici quelques détails, mais il reste des points à aborder, parce que, après tout la colonne montant payé ne peut-elle pas être calculée automatiquement elle aussi. Eh bien cette fois-ci c'est à vous de chercher, ce sera peut-être publié prochainement, mais n'attendez pas trop. Maintenant, place aux manipulations pour rentrer notre base de données dans un vrai logiciel.