

Gestion de Fond Commun de Placement

Soit une base de données Oracle d'informations financières :

Client (numCli, prenomCli, omCli, dateOuvertureCompte)

Action (codeAct, nomAct, valeurCourante, codeRegion)

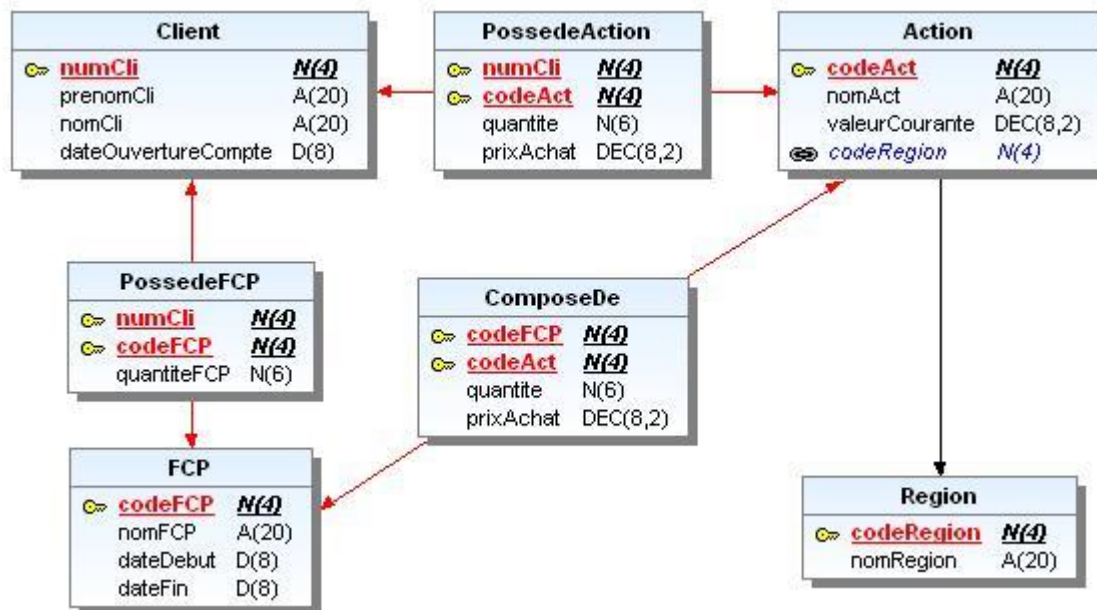
PossedeAction (numCli, codeAct, quantite, prixAchat)

FCP (codeFCP, nomFCP, dateDebut, dateFin)

ComposeDe (codeFCP, codeAct, quantite, prixAchat)

PossedeFcp (numCli, codeFCP, quantiteFCP)

Region (codeRegion, nomRegion)



On suppose que les FCP (Fond Commun de Placement, un peu comme une SICAV) sont achetés lors de leur émission. Une unité de FCP est composée de plusieurs actions (ce que l'on appelle un panier d'action). On suppose que tous les FCP sont en cours (date fin > date courante).

On suppose que le client ne revend rien...

Une implantation de cette base vous est fournie.

Ecrire les requêtes SQL Oracle répondant aux questions suivantes :

- Liste des clients (nom et prenom) classés par ordre alphabétique.
- Liste des actions (code, nom et valeur courante).
- Action la plus chère, la moins chère.
- Liste des actions possédées par Pierre Leloup.
- Gain (ou perte) total potentiel sur les actions de Pierre Leloup (valeur courante - prix achat).
- Gain (ou perte) potentiel par FCP.
- FCP le plus performant, le moins performant.
- Performance moyenne des FCP.
- Gain (ou perte) total potentiel de Pierre Leloup (actions + FCP).
- Client possédant le plus gros gain potentiel absolu (valeur).
- Client possédant le plus gros gain potentiel relatif (par rapport à la somme engagée).
- Client ayant le meilleur rendement annuel (vis-à-vis de la date d'ouverture de compte).
- Liste des FCP commencés en 2000.
- Code et nom du FCP qui se termine le premier.
- FCP éligible au PEA (comportant uniquement des actions de la région "Europe").
- Clients dont toutes valeurs sont éligibles au PEA.

PARTIE PL/SQL

Sous partie 1 : Questions simples

- Q1.** Écrire une procédure qui à partir du code d'une région renvoie le nom des actions disponible dans cette région.
- Q2.** Écrire un script qui vérifie si des combinaisons (codefc,codeact) existent plusieurs fois dans la table composeDe et affiche le nombre d'occurrence unique à chaque combinaison.

Sous partie 2 : Questions intermédiaires

- Q1.** Construire une fonction 'mdpcli' pour mot de passe client qui génère le mot de passe d'un client en concaténant son numéro de client, son nom, la date d'ouverture de son compte et un dièse (hashtag).
- Q2.** Construire dans le même temps une fonction 'email' pour email client qui génère un email client fictif en concaténant le nom et le prénom du client avec gmail comme extension par défaut.
- Q3.** Créer une table mot de passe où sera enregistré les mots de passes et emails générés par les deux fonctions précédentes. La table mot de passe 'mdp' contiendra les attributs identifiant client (idcl), mot de passe client (motpassecli), email client (emailcli) et date du jour (datedujour).

Sous partie 3 : Questions intermédiaires

- Q1.** Écrire une procédure qui calcul le total des actions d'un client (quantite dans possedeAction).

Q2. Créer une procédure qui augmente de 10 pour cent les prix des actions d'une région donnée puis l'utilisée pour augmenter le prix des actions de la région Europe par exemple.

Q3. Écrire une fonction qui retourne la liste des FCP expirées (dont la date de fin est antérieure à la date actuelle).

Sous partie 4 : Questions avancées

Q1. Créer une procédure qui ajoute automatiquement une action non associé à un FCP dans celui possédant le moins d'actions.

En d'autres termes : Si une action n'est pas associée à un FCP, l'associe à celui qui a le moins d'action.

Q2. Créer une procédure qui génère un rapport client détaillé pour chaque client. Elle devra fournir :

- Le nombre total d'actions possédées par le client
- La valeur total des actions détenues ($\text{quantite} \times \text{prix}$),
- Le nombre de FCP dans lesquels il est investi. Les résultats seront stockés dans une table rapportclient, qui servira d'historique.

Q3. Mettre en place un audit des transactions qui ont lieu sur la table possedeAction. Pour cela :

- Ajouter un trigger sur la table possedeAction pour conserver un historique de chaque modification (insert, update, delete).
- Utilisez une nouvelle table auditpossedeAction pour enregistrer les modifications avec la date et le type d'opération inclu.

Q4. Créer une procédure qui supprime les actions sans possesseurs, c'est à dire celle qui ne sont possédées par aucun client (aucune entrée dans possedeAction)

Sous partie 5 : Questions très avancées

Q1. Créer une procédure qui génère un portefeuille optimal pour chaque client, c'est à dire les actions les plus recommandées pour un client.

Q2. Calculer la rentabilité des fcp, en écrivant une fonction qui calcul la rentabilité d'un fcp en fonction des actions qu'il contient.

Q3. Comparaison de portefeuilles entre clients

Créer une procédure qui prend deux identifiants de clients en entrée et compare leurs portefeuilles :

- total des actions
- nombre de fcp communs
- valeur totale des portefeuilles.