

Ci-dessous des échanges avec Perplexity (à regarder en lien avec le diaporama de S Parise et celui de C Viroulaud). Si on ne lui dit pas précisément ce que l'on veut (1^{ère} question volontairement très succincte), le moteur de recherche Perplexity va « inventer » c.-à-d. compléter avec ce qu'il va trouver de plus présent sur le thème, ici BDD des croisières, mais voir comment il a pris en compte la demande précise et commenté la réponse apportée (ordre des villes). Toujours vérifier ce que l'on obtient bien entendu et épurer au besoin. Les scripts SQL (création de la BDD puis peuplement de la BDD) obtenus sont à exécuter dans un SDGB (Capitale par ex) pour pouvoir ensuite utiliser SQL pour consulter la BDD. Une dernière question volontairement fautive (une étoile en trop et table NAVIRES avec un s) est posée à perplexity à la fin de ce document (résultat de select ** from navires). Voir la réponse obtenue. On peut également lui demander ce qui ne va pas dans une requête en lui donnant la base de données ou en s'appuyant sur une qu'il vient de créer. Etc.

L'outil peut être utilisé pour créer des exercices.

Propose-moi une modélisation de la base de données des croisières. Une croisière passe par plusieurs villes. L'ordre des villes a de l'importance.

Sources



Voir 2 de plus

Réponse

Pour modéliser la base de données des croisières en tenant compte du fait qu'une croisière passe par plusieurs villes et que l'ordre des villes est important, voici une proposition de modèle entité-association (E/A) :

Entités principales

Croisière

- ID_Croisiere (clé primaire)
- Nom
- Date_Depart
- Date_Arrivee
- Durée

Ville

- ID_Ville (clé primaire)
- Nom
- Pays
- Coordonnées_GPS

Navire

- ID_Navire (clé primaire)
- Nom
- Capacité_Passagers
- Tonnage

Passager

- ID_Passager (clé primaire)
- Nom
- Prénom
- Date_Naissance

Relations

Itinéraire

- ID_Croisiere (clé étrangère)
- ID_Ville (clé étrangère)
- Ordre_Escale (pour indiquer l'ordre des villes visitées)
- Date_Arrivée
- Date_Départ
- Durée_Escale

Effectue

- ID_Croisiere (clé étrangère)
- ID_Navire (clé étrangère)

Réservation

- ID_Croisiere (clé étrangère)
- ID_Passager (clé étrangère)
- Date_Réservation
- Numéro_Cabine

Explications

1. L'entité **Croisière** représente chaque voyage organisé

⌘

.

⌘

L'entité **Ville** stocke les informations sur les ports visités pendant la croisière

⌘

.

⌘

La relation **Itinéraire** est cruciale car elle lie les croisières aux villes visitées. Le champ "Ordre_Escale" permet de définir l'ordre des escales, répondant ainsi à l'exigence que l'ordre des villes a de l'importance

⌘

.

⌘

L'entité **Navire** permet de gérer les différents bateaux utilisés pour les croisières

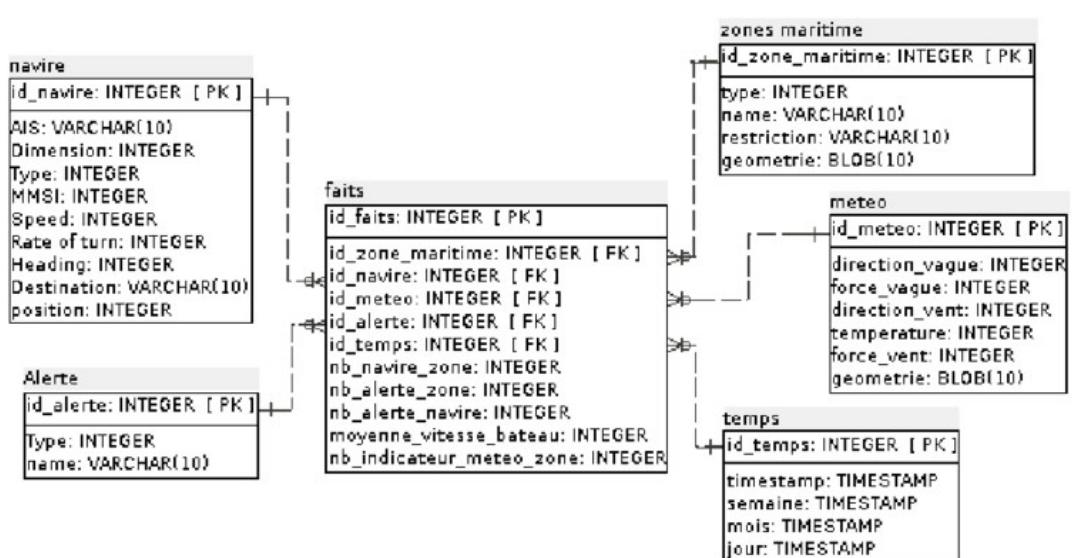
- La relation **Effectue** lie chaque croisière à un navire spécifique.
- L'entité **Passager** et la relation **Réservation** permettent de gérer les réservations des passagers pour chaque croisière

6. .

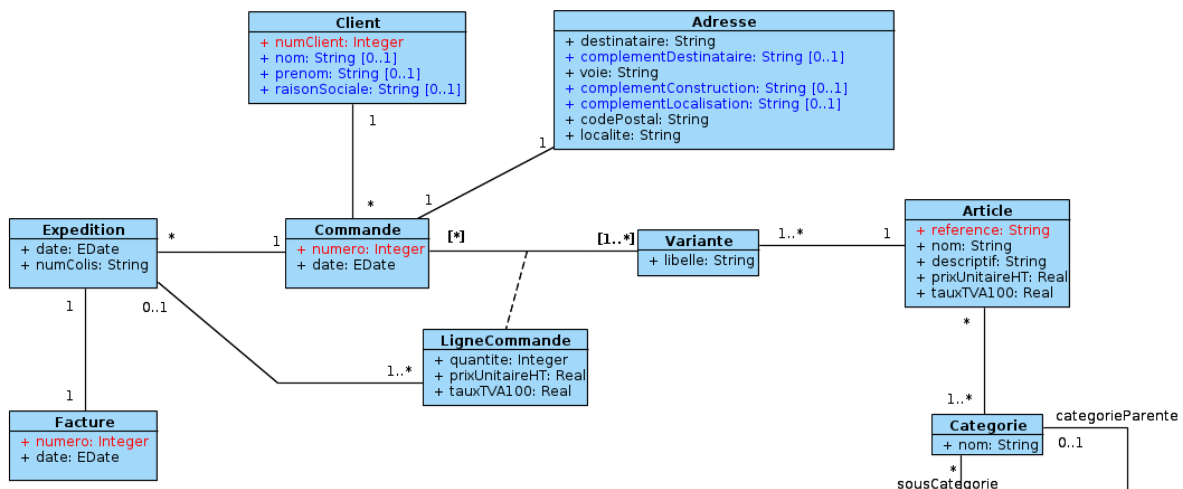
Ce modèle permet de répondre à diverses questions, telles que :

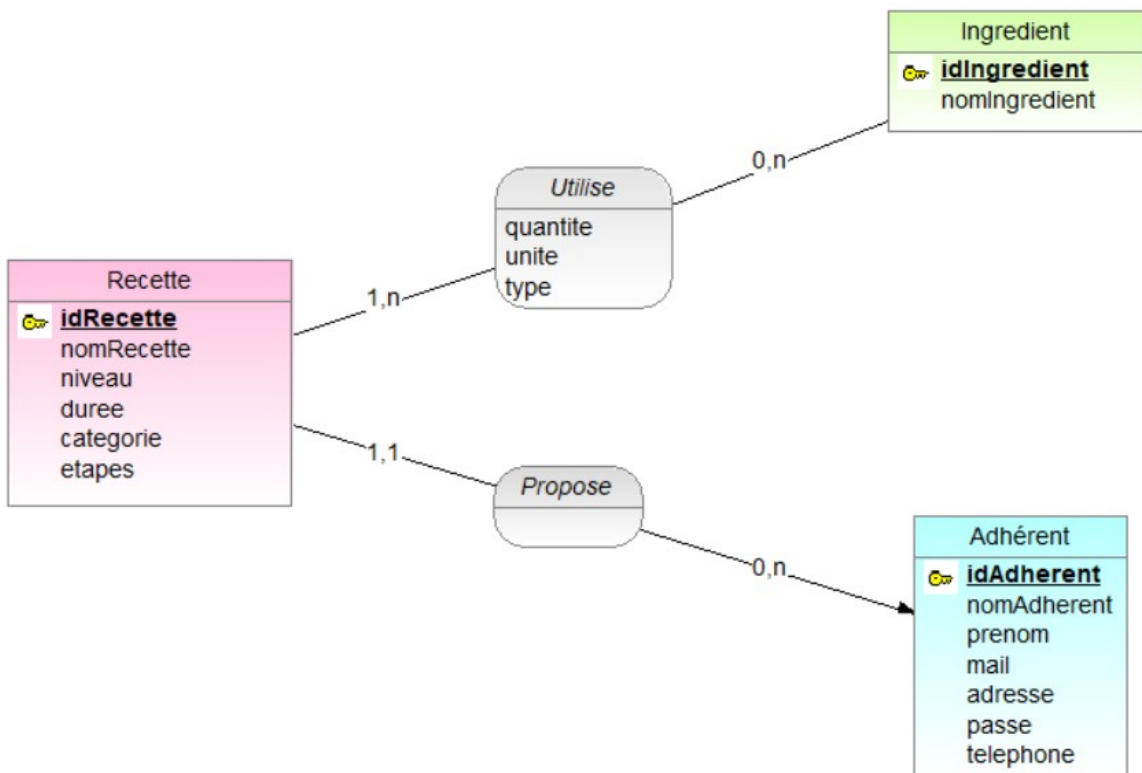
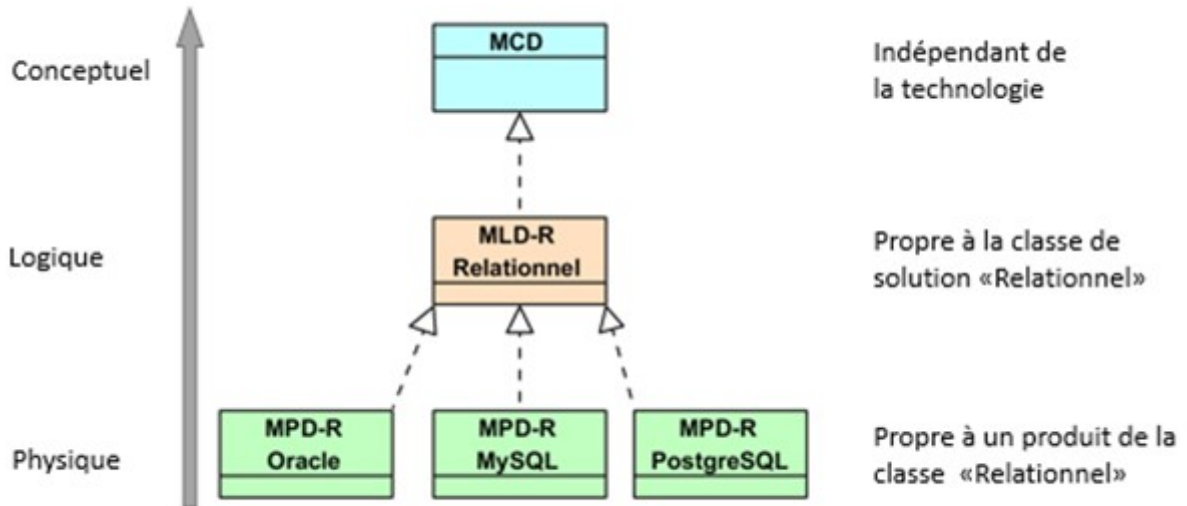
- Quelles sont les villes visitées lors d'une croisière spécifique et dans quel ordre ?
- Quel navire est utilisé pour une croisière donnée ?
- Quels passagers sont inscrits à une croisière particulière ?
- Quelle est la durée de chaque escale dans une ville ?

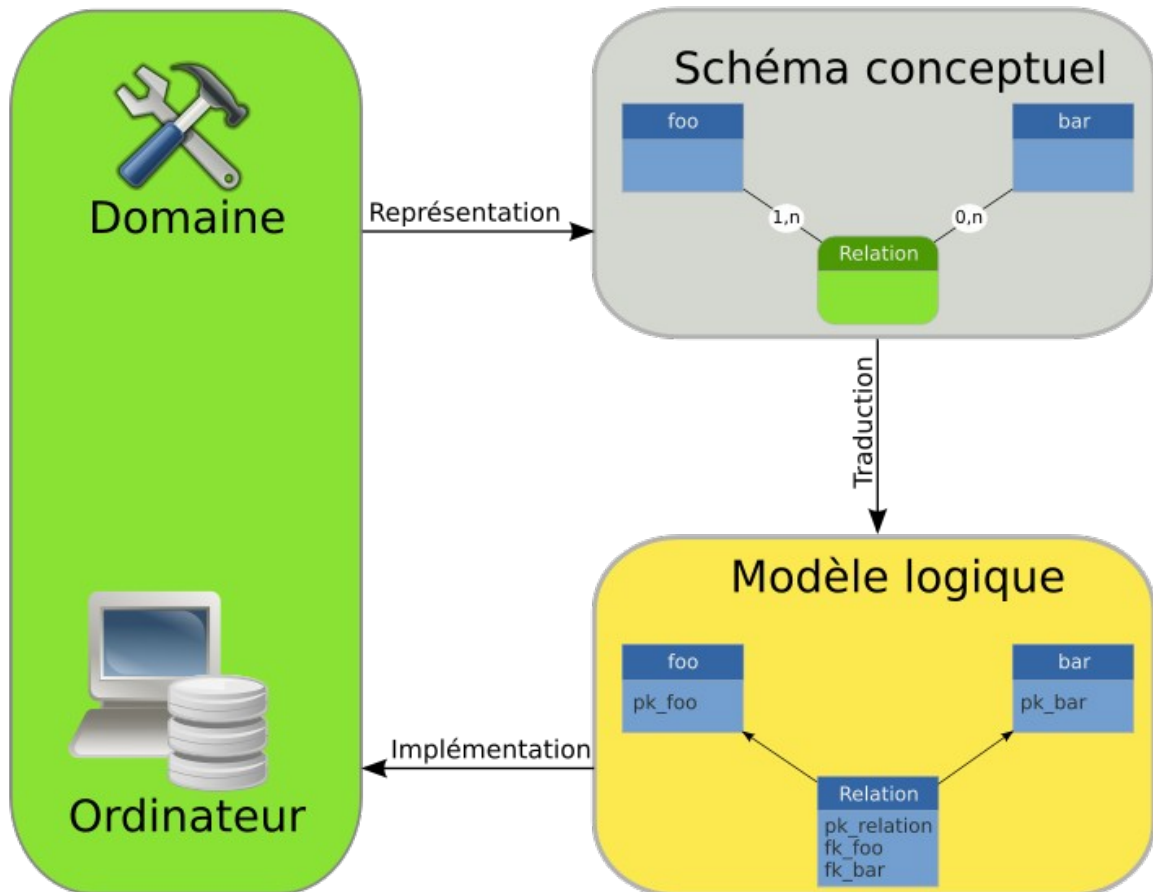
Il offre également la flexibilité nécessaire pour ajouter d'autres informations si besoin, comme les activités proposées dans chaque ville ou les détails sur les cabines des navires



5







Voir plus

Merci. Peux-tu écrire les requêtes sql de création de la base de données précédente

Réponse

Bien sûr ! Voici les requêtes SQL pour créer la base de données basée sur le modèle que nous avons discuté précédemment. Je vais utiliser la syntaxe SQL standard, qui devrait fonctionner avec la plupart des systèmes de gestion de base de données relationnelles (SGBDR).
sql

```
-- Création de la table Croisiere
CREATE TABLE Croisiere (
  ID_Croisiere INT PRIMARY KEY AUTO INCREMENT,
  Nom VARCHAR(100) NOT NULL,
  Date_Depart DATE NOT NULL,
  Date_Arrivee DATE NOT NULL,
  Duree INT NOT NULL
);
```

```
-- Création de la table Ville
CREATE TABLE Ville (
  ID_Ville INT PRIMARY KEY AUTO INCREMENT,
  Nom VARCHAR(100) NOT NULL,
  Pays VARCHAR(100) NOT NULL,
  Coordonnees_GPS VARCHAR(50)
);
```

```
-- Création de la table Navire
CREATE TABLE Navire (
  ID_Navire INT PRIMARY KEY AUTO INCREMENT,
  Nom VARCHAR(100) NOT NULL,
```

```

Capacite_Passagers INT NOT NULL,
Tonnage FLOAT NOT NULL
);

-- Création de la table Passager
CREATE TABLE Passager (
  ID_Passager INT PRIMARY KEY AUTO INCREMENT,
  Nom VARCHAR(100) NOT NULL,
  Prenom VARCHAR(100) NOT NULL,
  Date_Naissance DATE NOT NULL
);

-- Création de la table Itineraire
CREATE TABLE Itineraire (
  ID_Croisiere INT,
  ID_Ville INT,
  Ordre_Escale INT NOT NULL,
  Date_Arrivee DATETIME NOT NULL,
  Date_Depart DATETIME NOT NULL,
  Duree_Escale INT NOT NULL,
  PRIMARY KEY (ID_Croisiere, ID_Ville),
  FOREIGN KEY (ID_Croisiere) REFERENCES Croisiere(ID_Croisiere),
  FOREIGN KEY (ID_Ville) REFERENCES Ville(ID_Ville)
);

-- Création de la table Effectue
CREATE TABLE Effectue (
  ID_Croisiere INT,
  ID_Navire INT,
  PRIMARY KEY (ID_Croisiere, ID_Navire),
  FOREIGN KEY (ID_Croisiere) REFERENCES Croisiere(ID_Croisiere),
  FOREIGN KEY (ID_Navire) REFERENCES Navire(ID_Navire)
);

-- Création de la table Reservation
CREATE TABLE Reservation (
  ID_Croisiere INT,
  ID_Passager INT,
  Date_Reservation DATE NOT NULL,
  Numero_Cabine VARCHAR(20) NOT NULL,
  PRIMARY KEY (ID_Croisiere, ID_Passager),
  FOREIGN KEY (ID_Croisiere) REFERENCES Croisiere(ID_Croisiere),
  FOREIGN KEY (ID_Passager) REFERENCES Passager(ID_Passager)
);

```

Quelques notes sur ces requêtes :

1. J'ai utilisé `AUTO_INCREMENT` pour les clés primaires, ce qui est spécifique à MySQL. Pour d'autres SGBDR, vous pourriez avoir besoin d'utiliser une syntaxe différente (par exemple, `SERIAL` pour PostgreSQL).
2. Les types de données (`VARCHAR`, `INT`, `DATE`, etc.) peuvent être ajustés selon vos besoins spécifiques.
3. J'ai défini des contraintes de clé étrangère pour maintenir l'intégrité référentielle entre les tables.
4. Pour la table `Itineraire`, j'ai utilisé une clé primaire composée (`ID_Croisiere`, `ID_Ville`) pour permettre à une ville d'être visitée plusieurs fois dans différentes croisières, tout en maintenant l'unicité de chaque escale dans une croisière donnée.

- De même, pour les tables Effectue et Reservation, j'ai utilisé des clés primaires composées pour représenter les relations many-to-many.

Ces requêtes créeront la structure de base de données. Vous devrez ensuite insérer les données appropriées dans ces tables pour commencer à utiliser la base de données.

Merci. Peux-tu maintenant peupler cette base de données

Réponse

Bien sûr ! Je vais créer des exemples de données pour peupler chaque table de la base de données. Voici les requêtes SQL pour insérer des données fictives dans chaque table :

```
sql
-- Insertion de données dans la table Croisiere
INSERT INTO Croisiere (Nom, Date_Depart, Date_Arrivee, Duree) VALUES
('Méditerranée Enchantée', '2024-08-01', '2024-08-15', 14),
('Fjords Norvégiens', '2024-09-05', '2024-09-19', 14),
('Caraïbes Paradisiaques', '2024-10-10', '2024-10-24', 14);
```

```
-- Insertion de données dans la table Ville
INSERT INTO Ville (Nom, Pays, Coordonnees_GPS) VALUES
('Barcelone', 'Espagne', '41.3851, 2.1734'),
('Rome', 'Italie', '41.9028, 12.4964'),
('Athènes', 'Grèce', '37.9838, 23.7275'),
('Oslo', 'Norvège', '59.9139, 10.7522'),
('Bergen', 'Norvège', '60.3913, 5.3221'),
('Miami', 'États-Unis', '25.7617, -80.1918'),
('Nassau', 'Bahamas', '25.0480, -77.3554');
```

```
-- Insertion de données dans la table Navire
INSERT INTO Navire (Nom, Capacite_Passagers, Tonnage) VALUES
('Étoile des Mers', 3000, 120000.5),
('Fjord Explorer', 2500, 100000.0),
('Caribbean Dream', 3500, 130000.0);
```

```
-- Insertion de données dans la table Passager
INSERT INTO Passager (Nom, Prenom, Date_Naissance) VALUES
('Dupont', 'Jean', '1980-05-15'),
('Martin', 'Sophie', '1992-11-30'),
('Smith', 'John', '1975-03-22'),
('Garcia', 'Maria', '1988-07-10');
```

```
-- Insertion de données dans la table Itineraire
INSERT INTO Itineraire (ID_Croisiere, ID_Ville, Ordre_Escale, Date_Arrivee, Date_Depart, Duree_Escale) VALUES
(1, 1, 1, '2024-08-01 08:00:00', '2024-08-03 18:00:00', 58),
(1, 2, 2, '2024-08-05 07:00:00', '2024-08-07 19:00:00', 60),
(1, 3, 3, '2024-08-09 06:00:00', '2024-08-11 16:00:00', 58),
(2, 4, 1, '2024-09-05 09:00:00', '2024-09-07 20:00:00', 59),
(2, 5, 2, '2024-09-09 07:00:00', '2024-09-11 18:00:00', 59),
(3, 6, 1, '2024-10-10 08:00:00', '2024-10-12 19:00:00', 59),
(3, 7, 2, '2024-10-14 07:00:00', '2024-10-16 17:00:00', 58);
```

```
-- Insertion de données dans la table Effectue
INSERT INTO Effectue (ID_Croisiere, ID_Navire) VALUES
(1, 1),
(2, 2),
(3, 3);
```

```
-- Insertion de données dans la table Reservation
```

```
INSERT INTO Reservation (ID_Croisiere, ID_Passager, Date_Reservation,
Numero_Cabine) VALUES
(1, 1, '2024-01-15', 'A-101'),
(1, 2, '2024-02-20', 'B-205'),
(2, 3, '2024-03-10', 'C-310'),
(3, 4, '2024-04-05', 'D-415');
```

Ces requêtes vont insérer des données d'exemple dans toutes les tables de votre base de données. Voici un résumé des données insérées :

1. Trois croisières différentes
2. Sept villes (ports d'escale)
3. Trois navires
4. Quatre passagers
5. Des itinéraires pour chaque croisière
6. L'association entre les croisières et les navires
7. Quelques réservations de passagers

Ces données sont fictives et simplifiées, mais elles vous permettront de tester votre base de données et d'effectuer des requêtes pour vérifier son bon fonctionnement. Vous pouvez bien sûr ajouter plus de données ou les modifier selon vos besoins spécifiques.

Peux tu exécuter `select ** from navires`

Réponse

Je suis désolé, mais en tant qu'assistant virtuel, je n'ai pas accès direct à une base de données pour exécuter des requêtes SQL. Cependant, je peux vous montrer ce que serait le résultat attendu de la requête `SELECT * FROM Navire` basé sur les données que nous avons insérées précédemment. Voici ce que vous devriez voir si vous exécutiez cette requête sur la base de données que nous avons créée :

ID_Navire	Nom	Capacite_Passagers	Tonnage
1	Étoile des Mers	3000	120000.5
2	Fjord Explorer	2500	100000.0
3	Caribbean Dream	3500	130000.0

Cette table montre les trois navires que nous avons insérés dans la base de données, avec leurs caractéristiques :

1. L'ID du navire (généralisé automatiquement)
2. Le nom du navire
3. La capacité de passagers
4. Le tonnage du navire

Notez que les ID_Navire peuvent varier si vous avez inséré ou supprimé d'autres données entre-temps, car ils sont générés automatiquement par la base de données. Pour exécuter réellement cette requête, vous devriez la lancer dans votre système de gestion de base de données (comme MySQL, PostgreSQL, etc.) après avoir créé et peuplé la base de données avec les scripts que je vous ai fournis précédemment.