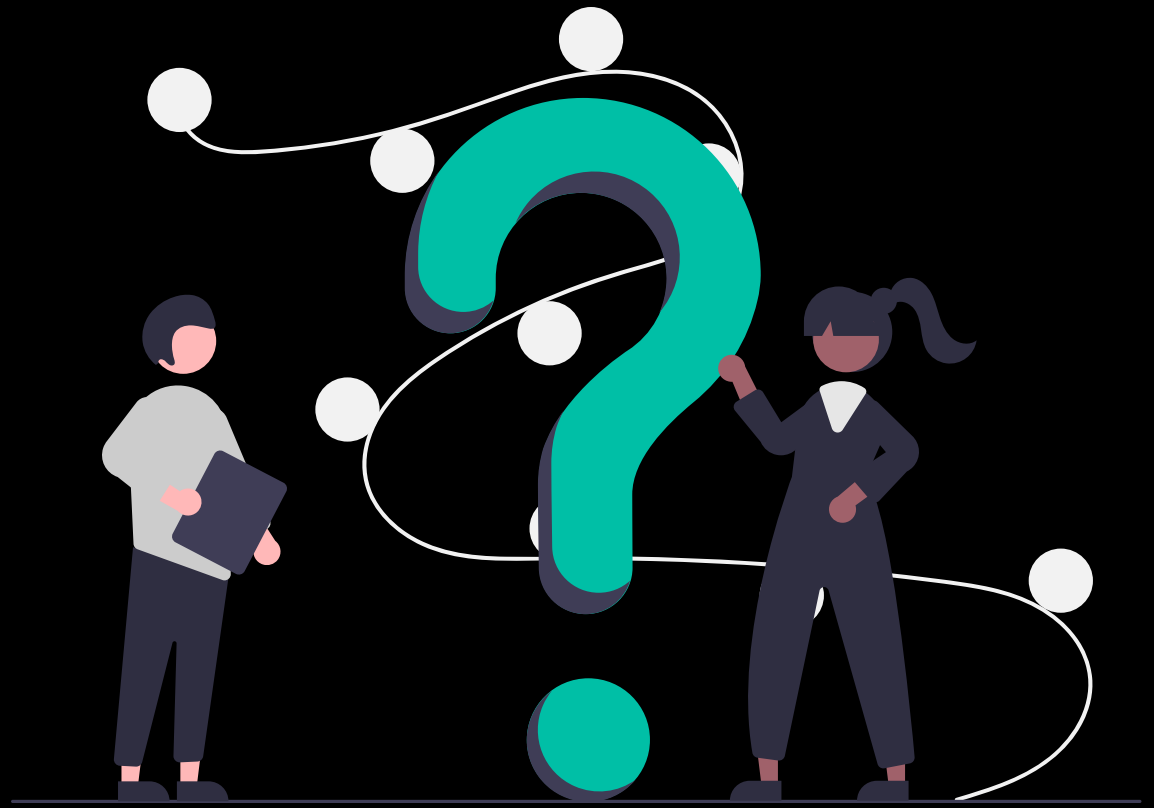
CERTIF ACADEMY

CREER LE MODELE OBJET

CERTIF ACADEMY

1. Paradigme impératif vs orienté objet
2. Pensez objets !
3. Les types et les classes
4. Un entity manager, un repository et les PDS/POD
5. L'agrégation et la composition

CERTIF ACADEMY



CERTIF ACADEMY

Traduire les schémas suivants avec une logique objet

CERTIF ACADEMY

METHODOLOGIE

1. Identifier les entités (noms communs)
2. Identifier les relations (verbes)
3. Identifier la cardinalité (se poser les bonnes questions)
4. Ecrire les classes pour construire les objets

CERTIF ACADEMY

Vous gérez le système de gestion de Carrefour.
On vous demande de faire l‘évolution suivante : il faut stocker les fournisseurs, les produits qu‘ils apportent et les rayons dans lesquels il faut mettre les produits.

Veuillez concevoir l‘architecture objet de la persistence.

CERTIF ACADEMY

SOLUTION

CERTIF ACADEMY

1. Identifier les entités (noms communs)
2. Identifier les relations (verbes)
3. Identifier la cardinalité (se poser les bonnes questions)
4. Ecrire les classes pour construire les objets

CERTIF ACADEMY

Il s‘agit tout d‘abord d‘identifier les entités/tables de la demande :

- Fournisseur
- Produit
- Rayon

CERTIF ACADEMY

1. Identifier les entités (noms communs)
2. Identifier les relations (verbes)
3. Identifier la cardinalité (se poser les bonnes questions)
4. Ecrire les classes pour construire les objets

CERTIF ACADEMY

Regardons les relations possibles en chaque paire de tables :

1. Fournisseur – Produit ?
2. Fournisseur – Rayon ?
3. Produit – Rayon ?

CERTIF ACADEMY

Regardons les relations possibles en chaque paire de tables :

1. Fournisseur – Produit ? (fournir)
2. Fournisseur – Rayon ? (aucun)
3. Produit – Rayon ? (composer)

CERTIF ACADEMY

Regardons les relations possibles en chaque pair de tables :

1. Fournisseur – Produit ? (fournir)
2. ~~Fournisseur – Rayon ? (aucun)~~
3. Produit – Rayon ? (composer)

CERTIF ACADEMY

Nous avons donc 2 relations que l’on souhaite stocker :

1. Fournisseur – Produit ? (fournir)
2. Produit – Rayon ? (composer)

CERTIF ACADEMY

1. Identifier les entités (noms communs)
2. Identifier les relations (verbes)
3. Identifier la cardinalité (se poser les bonnes questions)
4. Ecrire les classes pour construire les objets

CERTIF ACADEMY

On va traiter la 1er relation :

- Fournisseur – Produit ? (fournir)

CERTIF ACADEMY

Par exemple, une bonne réponse peut être :

- Un « Fournisseur » peut « fournir » combien de « Produit » ? Plusieurs produits
- Un «Produit » peut être « fourni » de combien de « Fournisseur » ? Plusieurs fournisseurs

CERTIF ACADEMY

C’est donc une relation ManyToMany (ou n-n) :

- Plusieurs fournisseurs
- Plusieurs produits

Pour une relation ManyToMany, nous allons créer une nouvelle structure.

CERTIF ACADEMY

On va traiter la 2eme relation :

- Produit – Rayon ? (composer)

CERTIF ACADEMY

Par exemple, une bonne réponse peut être :

- Un « Produit » peut « composer/être dans » combien de « Rayon » ? 1 seul rayon
- Un « Rayon » peut être « composé » de combien de « Produit » ? Plusieurs produits

CERTIF ACADEMY

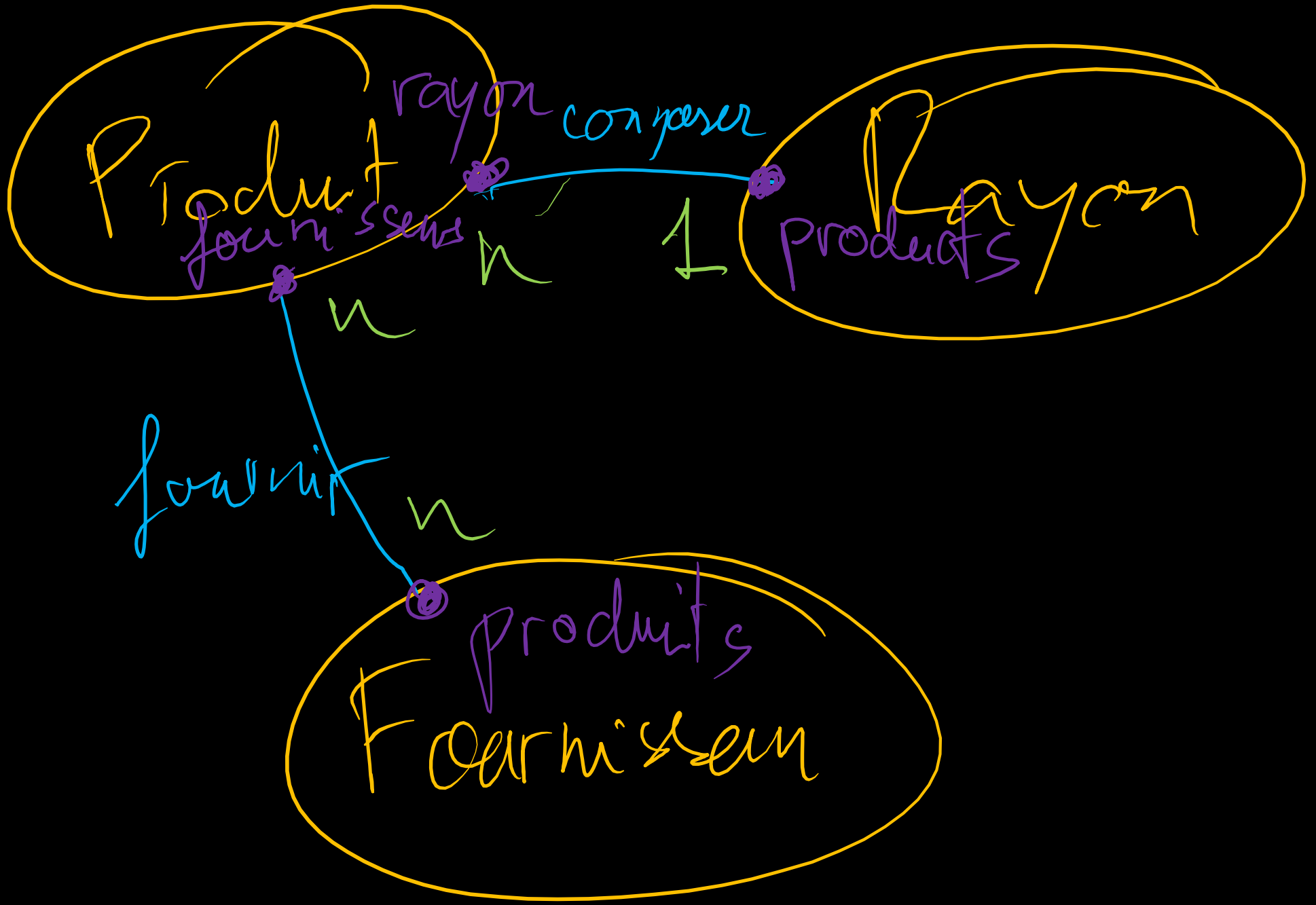
C’est donc une relation OneToMany (ou 1-n) :

- Plusieurs produits
- Un seul rayon

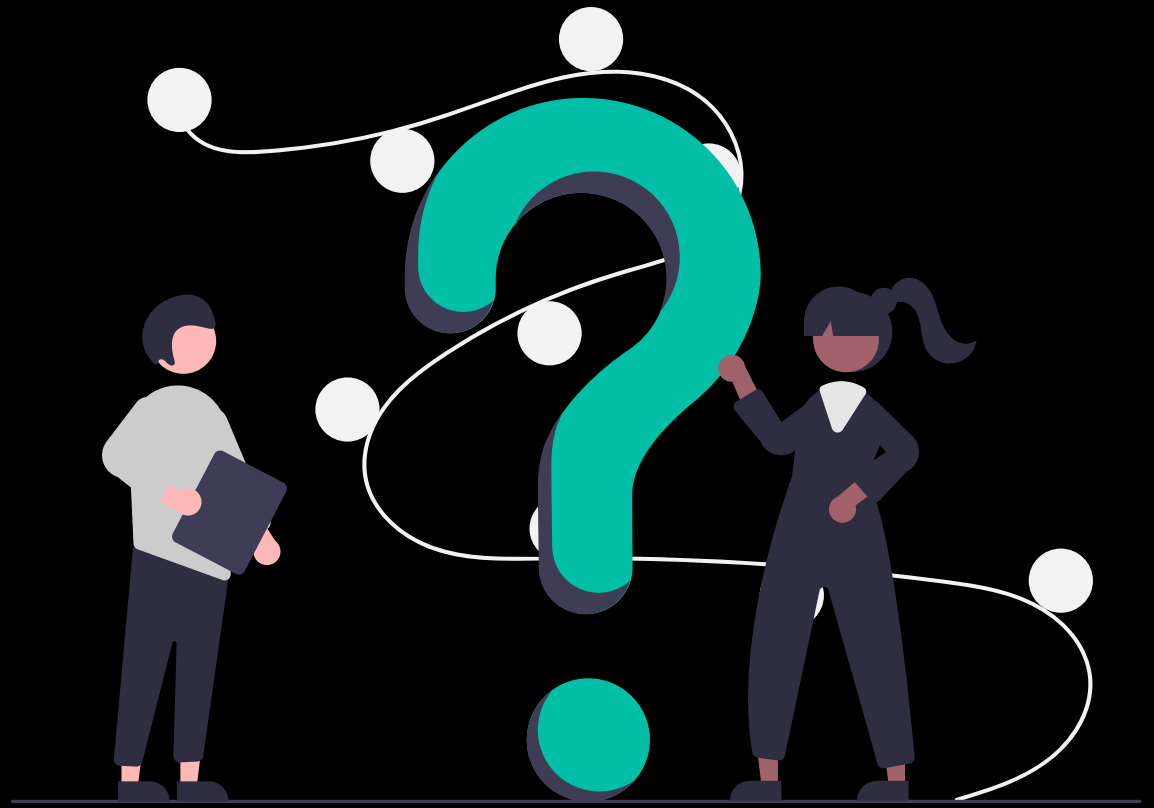
CERTIF ACADEMY

1. Identifier les entités (noms communs)
2. Identifier les relations (verbes)
3. Identifier la cardinalité (se poser les bonnes questions)
4. Ecrire les classes pour construire les objets

CERTIF ACADEMY



CERTIF ACADEMY



CERTIF ACADEMY

Vous souhaitez réaliser un site internet comme Doctolib. Le client vous demande de gérer les médecins, les patients et leur rendez-vous.

Veuillez concevoir l‘architecture objet de la persistence.

CERTIF ACADEMY

SOLUTION

CERTIF ACADEMY

1. Identifier les entités (noms communs)
2. Identifier les relations (verbes)
3. Identifier la cardinalité (se poser les bonnes questions)
4. Ecrire les classes pour construire les objets

CERTIF ACADEMY

Il s‘agit tout d‘abord d‘identifier les entités/tables de la demande :

- Medecin
- Patient
- Rendez-vous

CERTIF ACADEMY

1. Identifier les entités (noms communs)
2. Identifier les relations (verbes)
3. Identifier la cardinalité (se poser les bonnes questions)
4. Ecrire les classes pour construire les objets

CERTIF ACADEMY

Regardons les relations possibles en chaque paire de tables :

1. Medecin – Patient ?
2. Medecin – Rendez-vous ?
3. Patient – Rendez-vous ?

CERTIF ACADEMY

Regardons les relations possibles en chaque paire de tables :

1. Medecin – Patient ? (aucun)
2. Medecin – Rendez-vous ? (s’occuper)
3. Patient – Rendez-vous ? (prendre)

CERTIF ACADEMY

Regardons les relations possibles en chaque paire de tables :

1. ~~Medecin – Patient ? (aucun)~~
2. Medecin – Rendez-vous ? (s’occuper)
3. Patient – Rendez-vous ? (prendre)

CERTIF ACADEMY

Regardons les relations possibles en chaque paire de tables :

1. Medecin – Rendez-vous ? (s’occuper)
2. Patient – Rendez-vous ? (prendre)

CERTIF ACADEMY

1. Identifier les entités (noms communs)
2. Identifier les relations (verbes)
3. Identifier la cardinalité (se poser les bonnes questions)
4. Ecrire les classes pour construire les objets

CERTIF ACADEMY

On va traiter la 1er relation :
Medecin – Rendez-vous ? (s’occuper)

CERTIF ACADEMY

Par exemple, une bonne réponse peut être :

- Un « Medecin » peut « s’occuper » de combien de « Rendez-vous» ? Plusieurs rendez-vous
- Un «Rendez-vous » peut être « occupé » par combien de « Medecin » ? 1 médecin

CERTIF ACADEMY

C’est donc une relation OneToMany (ou n-n) :

- Plusieurs rendez-vous
- Un seul medecin

Pour une relation OneToMany, nous allons créer une nouvelle colonne.

CERTIF ACADEMY

On va traiter la 2eme relation :
Patient – Rendez-vous ? (prendre)

CERTIF ACADEMY

Par exemple, une bonne réponse peut être :

- Un « Patient » peut « prendre » combien de « Rendez-vous » ? Plusieurs rendez-vous
- Un « Rendez-vous » peut être « pris » par combien de « Patient » ? 1 seul patient

CERTIF ACADEMY

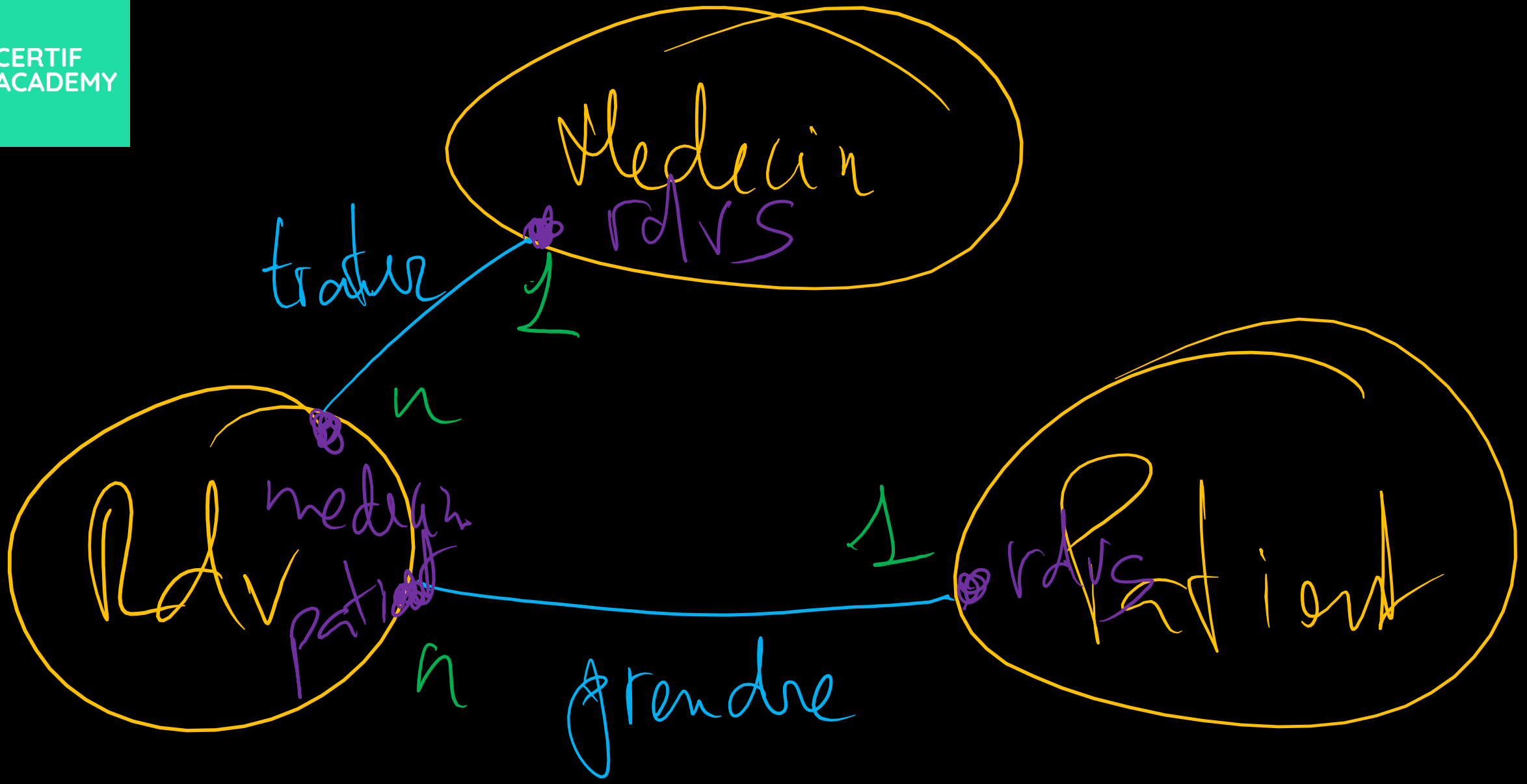
C’est donc une relation OneToMany (ou 1-n) :

- Plusieurs rendez-vous
- Un seul patient

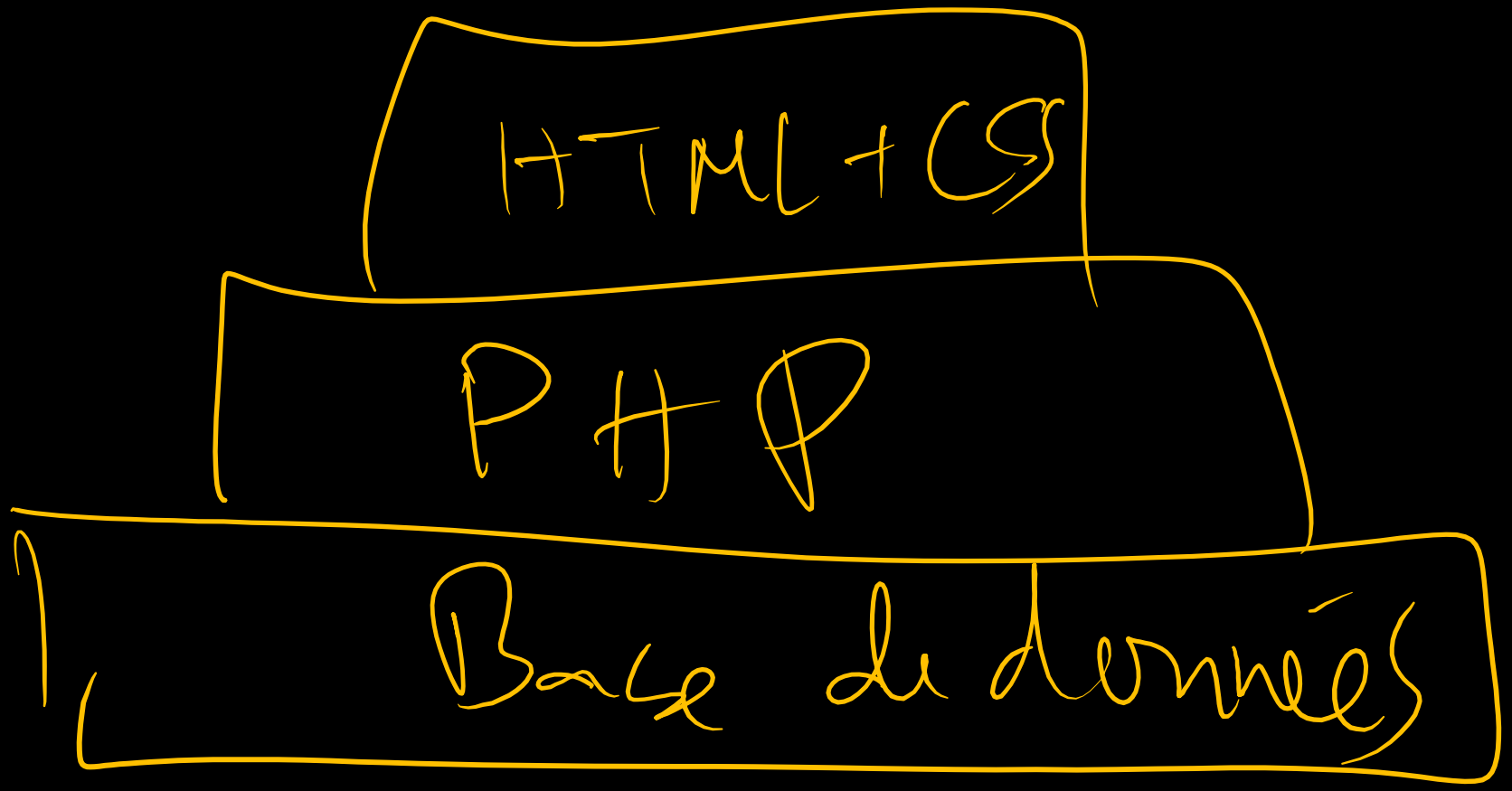
CERTIF ACADEMY

1. Identifier les entités (noms communs)
2. Identifier les relations (verbes)
3. Identifier la cardinalité (se poser les bonnes questions)
4. Ecrire les classes pour construire les objets

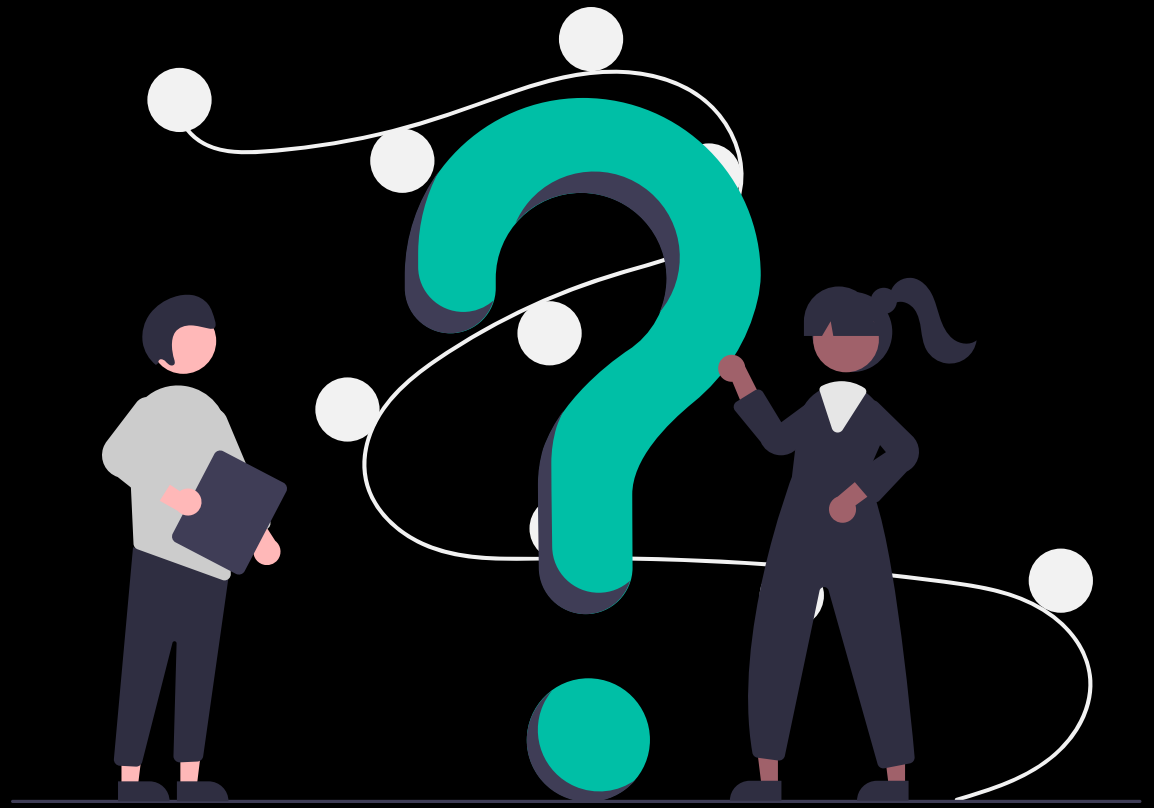
CERTIF ACADEMY



CERTIF ACADEMY



CERTIF ACADEMY



CERTIF ACADEMY

On vous demande de faire un site internet comme youtube. Le client vous demande de gérer les videos, les auteurs qui créent ces videos et les channels de videos.

Veuillez concevoir l‘architecture objet de la persistence.

CERTIF ACADEMY

SOLUTION

CERTIF ACADEMY

1. Identifier les entités (noms communs)
2. Identifier les relations (verbes)
3. Identifier la cardinalité (se poser les bonnes questions)
4. Ecrire les classes pour construire les objets

CERTIF ACADEMY

Il s‘agit tout d‘abord d‘identifier les entités/tables de la demande :

- Video
- Auteur
- Channel

CERTIF ACADEMY

1. Identifier les entités (noms communs)
2. Identifier les relations (verbes)
3. Identifier la cardinalité (se poser les bonnes questions)
4. Ecrire les classes pour construire les objets

CERTIF ACADEMY

Regardons les relations possibles en chaque pair de tables :

1. Video – Auteur ?
2. Video – Channel ?
3. Auteur – Channel ?

CERTIF ACADEMY

Regardons les relations possibles en chaque pair de tables :

1. Video – Auteur ? (créer)
2. Video – Channel ? (inclure)
3. Auteur – Channel ? (créer)

CERTIF ACADEMY

1. Identifier les entités (noms communs)
2. Identifier les relations (verbes)
3. Identifier la cardinalité (se poser les bonnes questions)
4. Ecrire les classes pour construire les objets

CERTIF ACADEMY

On va traiter la 1er relation :
Video – Auteur ? (créer)

CERTIF ACADEMY

Par exemple, une bonne réponse peut être :

- Une « Auteur » peut « créer » combien de « Video » ? Plusieurs videos
- Une « Video » peut être « créée » par combien de « Auteur » ? A seul auteur

CERTIF ACADEMY

C’est donc une relation OneToMany (ou 1-n) :

- Plusieurs videos
- Un seul auteur

CERTIF ACADEMY

On va traiter la 2eme relation :
Video – Channel ? (inclure)

CERTIF ACADEMY

Par exemple, une bonne réponse peut être :

- Une « Channel » peut « inclure » combien de «Video » ? Plusieurs videos
- Un « Video » peut être « inclus » dans combien de « Channel » ? Plusieurs channels

CERTIF ACADEMY

C’est donc une relation ManyToMany (ou n-n) :

- Plusieurs videos
- Plusieurs channels

CERTIF ACADEMY

On va traiter la 3eme relation :
Auteur – Channel ? (créer)

CERTIF ACADEMY

Par exemple, une bonne réponse peut être :

- Une « Auteur » peut « créer » combien de « Channel » ? Plusieurs channels
- Un « Channel » peut être « inclus » dans combien de « Auteur » ? Un seul auteur

CERTIF ACADEMY

C’est donc une relation OneToMany (ou 1-n) :

- Plusieurs channels
- 1 seul auteur

CERTIF ACADEMY

1. Identifier les entités (noms communs)
2. Identifier les relations (verbes)
3. Identifier la cardinalité (se poser les bonnes questions)
4. Ecrire les classes pour construire les objets

CERTIF
ACADEMY

CERTIF ACADEMY