

Normalisation

Projet : CVthèque

Avant-propos : dans tout le document, on préfère ajouter une clé artificielle plutôt que de faire une relation toute clé ce qui n'est pas pratique.

→ candidat

candidat(#identifiant : VARCHAR, mot_de_passe : VARCHAR,
nom : VARCHAR, prenom : VARCHAR, e_mail : VARCHAR,
cv=>CV.IDCV)
avec identifiant clé, mot_de_passe, nom, prenom, e_mail NOT
NULL et cv UNIQUE NOT NULL.

L'identifiant est immuable et détermine les autres attributs qui eux ne sont pas immuables. On peut aussi dire que e-mail détermine les autres attributs.

identifiant ->mot_de_passe, nom, prenom, e_mail, cv

e_mail->identifiant, mot_de_passe, nom, prenom, cv

Ainsi, on a deux clés candidates qui sont identifiant et e_email, elle est immuable et non composée. Donc on a une 3NF.

→ cv

cv(#IDCV : INTEGER ,titre : VARCHAR, langue : ENUM, statut :
ENUM, date_crea : DATE, date_modif : DATE, info =>
info_complementaire.IDINFO)

On ajoute une clé étrangère car aucune clé potentielle, IDCV clé, et titre, langue, statut, date_crea, date_modif NOT NULL.

Plusieurs CV peuvent avoir le même titre, et être créer ou modifier à la même date, ainsi on n'a pas de dépendances fonctionnelles, donc on peut en faire une relation toute clé, mais cela n'est pas pratique donc on doit ajouter une clé artificielle pour passer en 3NF, avec IDCV qui détermine tous les attributs. De cette manière, on est en 3NF.

cv(#IDCV : INTEGER ,titre : VARCHAR, langue : ENUM, statut : ENUM,
date_crea : DATE, date_modif : DATE, info =>
info_complementaire.IDINFO)

On ajoute une clé étrangère car aucune clé potentielle, IDCV clé, et titre, langue, statut, date_crea, date_modif NOT NULL.

→ référent

referant(#nom_entreprise : VARCHAR, nom : VARCHAR, prenom : VARCHAR, candidat => candidat.identifiant)
avec nom_entreprise clé, nom, prenom NOT NULL, et candidat UNIQUE et NOT NULL.

Le nom de l'entreprise ne peut déterminer le nom, le prénom du référent et le candidat. Aucune dépendance ne peut être exprimée. On peut là aussi faire une relation toute clé mais cela n'est pas pratique donc on doit ajouter une clé artificielle pour passer en 3NF comme IDref qui détermine tous les autres attributs. On est en 3NF.

referant(#IDref : VARCHAR, nom_entreprise : VARCHAR, nom : VARCHAR, prenom : VARCHAR, candidat => candidat.identifiant)
avec IDref clé, nom_entreprise, nom, prenom NOT NULL, et candidat UNIQUE et NOT NULL.

→ publication

publication(#ISBN : INTEGER, titre : VARCHAR, date : DATE, contenu : VARCHAR, candidat => candidat.identifiant)
avec ISBN clé, titre, date, contenu NOT NULL, et candidat NOT NULL et UNIQUE.

L'ISBN détermine le titre, la date de parution et l'auteur (qui est supposé seul). Le titre détermine le contenu. Par transitivité, l'ISBN détermine le contenu. Cependant, en réalité toute publication n'a pas forcément d'ISBN. Donc ça ne peut pas être une clé. On doit donc ajouter une clé artificielle comme IDpub.

publication(#IDpub : VARCHAR, ISBN : INTEGER, titre : VARCHAR, date : DATE, contenu : VARCHAR, candidat => candidat.identifiant)
avec IDpub clé, titre, date, contenu NOT NULL, et candidat NOT NULL et UNIQUE.

→ web

web(#URL : VARCHAR, type_web : ENUM, candidat => candidat.identifiant)

avec URL clé, type NOT NULL et candidat NOT NULL et UNIQUE.

Une URL ne peut déterminer le type web, en effet celui ci peut changer.

De plus, une URL peut correspondre à plusieurs candidats s'il s'agit d'un site de groupe par exemple. Ainsi, on ne peut déterminer de clé. On ajoute une clé artificielle comme IDweb pour passer en 3NF.

web(#IDweb : VARCHAR, URL : VARCHAR, type_web : ENUM, candidat => candidat.identifiant)

avec IDweb clé, URL UNIQUE NOT NULL, type NOT NULL et candidat NOT NULL et UNIQUE.

→ téléphone

telephone(#numero : VARCHAR, type_tel = ENUM, candidat => candidat.identifiant}

avec numero clé, type NOT NULL, et candidat NOT NULL.

En réalité numero ne peut ni déterminer type_tel ni candidat, car il peut être changé. Ainsi, il n'existe pas de clé. On peut ajouter une clé artificielle comme IDtel qui détermine tous les autres attributs. On est en 3NF.

telephone(#IDtel : VARCHAR, numero : VARCHAR, type_tel = ENUM, candidat => candidat.identifiant}

avec IDtel clé, numero UNIQUE NOT NULL, type NOT NULL, et candidat NOT NULL.

→ expérience

experience(#nom_entreprise : VARCHAR, secteur_act : VARCHAR)

avec nom_entreprise clé, et secteur d'activité NOT NULL.

Le nom de l'entreprise détermine de secteur d'activité. Donc nom_entreprise est bien une clé. La relation est en 3NF.

→ asso_exp

asso_exp(#cv => CV.IDCV,

#experience=>experience.nom_entreprise, titre_poste :

VARCHAR, date_debut : DATE, date_fin : DATE, fonction :
VARCHAR)

avec (cv, experience) clé, titre_poste, date_debut, date_fin,
fonction NOT NULL et date_debut < date_fin.

On peut voir les dépendances fonctionnelles suivantes :

(cv,experience)->titre_poste, date_debut, date_fin, fonction
titre_poste->fonction.

On doit donc décomposer pour passer en 3NF :

asso_exp(#cv => CV.IDCV, #experience=>experience.nom_entreprise,
date_debut : DATE, date_fin : DATE, titre_poste=>poste)

avec (cv, experience) clé, date_debut, date_fin NOT NULL et
date_debut < date_fin.

poste(#titre_poste : VARCHAR, fonction : VARCHAR)

avec titre_poste clé, fonction NOT NULL.

→ infos complémentaires

info_complementaire(#IDINFO : INTEGER, permis : ENUM,
vehicule : ENUM)

avec IDINFO clé étrangère car aucune clé candidate.

On ne peut pas dire que le fait d'avoir un permis détermine le fait qu'on ait un véhicule ou non. De même, on ne peut pas dire qu'avoir un véhicule détermine le fait qu'on ait le permis, en effet on peut acheter une voiture sans avoir le permis, ou alors avoir une voiture mais avoir perdu son permis. Ainsi, nous n'avons aucune clé candidate. Pour normaliser, nous avons besoin d'une clé artificielle comme IDINFO tel que IDINFO détermine permis et véhicule. On a une 3NF.

info_complementaire(#IDINFO : INTEGER, permis : ENUM, vehicule :
ENUM)

avec IDINFO clé étrangère car aucune clé candidate.

→ formation

formation(#titre : VARCHAR, #établissement : VARCHAR,
type_formation : VARCHAR, pays : VARCHAR, ville : VARCHAR)
avec (titre, établissement) clé, type, pays et ville NOT NULL.

On peut déterminer les dépendances fonctionnelles suivantes :

(titre,établissement)=>type_formation, pays, ville.

En revanche, on ne peut pas dire que la ville détermine la pays car plusieurs villes homonymes peuvent être dans des pays différents.

Donc, (titre,établissement) est une clé et on a une 3NF.

→ compétence

competence(#nom : VARCHAR)

avec nom clé.

La clé est atomique, et aucun attribut ne dépend d'elle, donc on est en 3NF.

→ langue

langue(#langue : VARCHAR)

avec langue clé.

La clé est atomique, et aucun attribut ne dépend d'elle, donc on est en 3NF.

→ association

association(#nom : VARCHAR)

avec nom clé.

La clé est atomique, et aucun attribut ne dépend d'elle, donc on est en 3NF.

→ vie associative

vie_associative(date_debut : DATE, date_fin : DATE, poste :

VARCHAR, #association => association, #cv => CV)

avec (association, cv) clé, date_debut NOT NULL, date_debut < date_fin.

(association,cv,poste)->(date_debut,date_fin), en effet le cv peut nous permettre d'avoir le candidat ajouté au poste et à l'association on peut retrouver les dates auxquelles il était membre de l'association. On a donc la clé (association, cv, poste) qui détermine tous les autres

attributs, et les attributs ne dépendent que de l'entièreté de cette clé. On a donc une 3NF.

vie_associative(date_debut : DATE, date_fin : DATE, #poste : VARCHAR, #association => association, #cv => CV)
avec (association, cv, poste) clé, date_debut NOT NULL, date_debut < date_fin.

→ niveau

niveau(#langue => langue, #cv => CV, niveau = VARCHAR)
avec (langue, cv) clé, et langue, cv, niveau NOT NULL.

(langue,cv)->niveau, niveau n'est pas immuable, en effet le niveau d'un candidat en une langue peut évoluer. Il s'agit d'une transformation d'une association N:M, l'attribut niveau n'est pas déterminé par une partie de la clé. On a donc une 3NF.

→ association compétence

assoc_comp(#competence => competence, #cv => CV)
avec (competence, cv) clé.

C'est une transformation d'une association N:M, on a une relation toutes clés, les clés candidates sont indépendantes entre elles et atomiques.
Ainsi, cette relation est en 3NF.

→ assoc_formation

assoc_formation(date_debut : DATE, date_fin : DATE, #formation => formation, #cv => CV)
avec (formation, cv) clé, date_debut, date_fin NOT NULL et
date_debut < date_fin.

C'est une transformation d'une association N:M, (cv,formation) détermine (date_debut,date_fin) car à partir du cv (rélié au candidat) et de sa formation, on peut en déduire les années qu'il a passé dans l'établissement grâce aux bases de données de l'école. Donc (formation,cv) est bien une clé. Et on a une 3NF.