



UNIVERSITÉ COTE D'AZUR/FDS

TRAVAIL PRATIQUE TRANSVERSAL MCD MERISE,
SCHÉMA DE TYPES SQL3, PLSQL, DBA1 ET
TUNING DE BDR

Sujet : Gestion d'un cabinet médical

Etudiants :

- * PIERRE Bob Charlemagne
- * DOUILLY Rodely
- * MILORME Pierre Rubens
- * SURLIN Djimy

Professeur :

Gabriel MOPOLO-MOKE

Date :

Septembre 2024

Table des matières

1	Choix du sujet	3
1.1	Description du sujet	3
2	Descriptions textuelles des requêtes de mise à jour	4
2.1	2 requêtes impliquant 1 table	4
2.1.1	REQUÊTE 1	4
2.1.2	REQUÊTE 2	4
2.2	2 requêtes impliquant 2 tables	4
2.2.1	REQUÊTE 1	4
2.2.2	REQUÊTE 2	4
2.3	2 requêtes impliquant plus de 2 tables	4
2.3.1	REQUÊTE 1	4
2.3.2	REQUÊTE 2	5
3	Description textuelles des requêtes de suppression	6
3.1	2 requêtes impliquant 1 table	6
3.1.1	REQUÊTE 1	6
3.1.2	REQUÊTE 2	6
3.2	2 requêtes impliquant 2 tables	6
3.2.1	REQUÊTE 1	6
3.2.2	REQUÊTE 2	6
3.3	2 requêtes impliquant plus de 2 tables	6
3.3.1	REQUÊTE 1	6
3.3.2	REQUÊTE 2	6
4	Description textuelles des requêtes de consultation	7
4.1	5 requêtes impliquant 1 table dont 1 avec un group By et une avec un Order By	7
4.1.1	REQUÊTE 1	7
4.1.2	REQUÊTE 2	7
4.1.3	REQUÊTE 3	7
4.1.4	REQUÊTE 4	7
4.1.5	REQUÊTE 5	7
4.2	5 requêtes impliquant 2 tables avec jointures internes dont 1 externe + 1 group by + 1 tri	8
4.2.1	REQUÊTE 1	8
4.2.2	REQUÊTE 2	8
4.2.3	REQUÊTE 3	8
4.2.4	REQUÊTE 4	8

4.2.5	REQUÊTE 5	8
4.3	5 requêtes impliquant plus de 2 tables avec jointures internes dont 1 externe + 1 group by + 1 tri	9
4.3.1	REQUÊTE 1	9
4.3.2	REQUÊTE 2	9
4.3.3	REQUÊTE 3	9
4.3.4	REQUÊTE 4	9
4.3.5	REQUÊTE 5	9
5	Dictionnaire de données MERISE	10
6	Descriptions textuelles des associations	14
6.1	Association « Effectuer » entre MEDECIN et CONSULTATION . . .	14
6.2	Association « Inclure » entre FACTURE et CONSULTATION . . .	14
6.3	Association « Passer » entre PATIENT et CONSULTATION	14
6.4	Association « Necessiter » entre CONSULTATION et EXAMEN . .	14
6.5	Association « Recevoir » entre PATIENT et FACTURE	14
6.6	Association « Contenir » entre CONSULTATION et PRESCRIPTION	15
6.7	Association « Avoir_Rendez_Vous » entre PATIENT et MEDECIN .	15
6.7.1	Propriétés de l'association Avoir_Rendez_Vous	15
7	Définition du Modèle Entité-Association MERISE	16
8	Définition du schéma de type à partir du modèle conceptuel	17
9	Spécification méthodes dans chaque type	17
9.1	Type PERSONNE_T	17
9.2	Type PATIENT_T	18
9.3	Type MEDECIN_T	18
9.4	Type EXAMEN_T	19
9.5	Type FACTURE_T	19
9.6	Type CONSULTATION_T	19
9.7	Type PRESCRIPTION_T	20
9.8	Type RENDEZ_VOUS_T	20
10	Modèle de classe UML	21

1 Choix du sujet

Sujet 13 : Gestion d'un cabinet médical

1.1 Description du sujet

Cette application pour un cabinet médical permet la gestion complète des patients, des médecins, des rendez-vous, des consultations, des prescriptions, des examens et de la facturation. Les patients peuvent être enregistrés avec leurs détails personnels, tandis que les médecins sont répertoriés avec leur spécialité respective. Les rendez-vous entre patients et médecins sont programmés, enregistrés dans la table RENDEZ-VOUS. Chaque consultation est consignée dans la table CONSULTATION, associée à un patient, un médecin et une facture. Les prescriptions médicales sont enregistrées dans la table PRESCRIPTION, liées à la consultation correspondante. Les détails des examens sont stockés dans la table EXAMEN, également liés à la consultation. Chaque consultation génère une facture, enregistrée dans la table FACTURE avec le montant total à payer. Des contraintes référencielles (avec les types REF) garantissent l'intégrité des données et les relations entre les tables objets.

En résumé, cette application fournit un système complet pour gérer les opérations quotidiennes d'un cabinet médical, optimisant le suivi des patients et la gestion des consultations et des facturations.

2 Descriptions textuelles des requêtes de mise à jour

2.1 2 requêtes impliquant 1 table

2.1.1 REQUÊTE 1

Cette requête modifie la date de naissance du patient dont l'adresse mail est thomas.leclerc@email.com. La nouvelle date de naissance sera 22-AUG-1986. Seules les lignes correspondant à l'adresse mail spécifiée seront affectées.

2.1.2 REQUÊTE 2

Cette requête modifie l'adresse du patient dont l'identifiant est 1. La nouvelle adresse sera 90, DELMAS 75. Seule la ligne correspondant à l'identifiant spécifié sera affectée.

2.2 2 requêtes impliquant 2 tables

2.2.1 REQUÊTE 1

Cette requête SQL permet de modifier la date de rendez-vous des patients dont le rendez-vous est prévu entre le 14-FEB-2024 et le 18-FEB-2024 dans la table virtuelle issue de la jointure entre PATIENT et RENDEZ VOUS. La nouvelle date de rendez-vous sera le 01-MAR-24.

2.2.2 REQUÊTE 2

Cette requête parcourt toutes les factures dont le montant total est inférieur à 200 et pour lesquelles un patient correspondant existe. Pour chacune de ces factures, elle augmente le montant total de 10

2.3 2 requêtes impliquant plus de 2 tables

2.3.1 REQUÊTE 1

Cette requête met à jour la colonne DETAILS_PRESCRIPTION de la table PRESCRIPTION pour les enregistrements associés à des consultations de patients ayant un identifiant de patient égal à 1 et où la date de consultation est le 5 février 2024. La nouvelle valeur de la colonne DETAILS_PRESCRIPTION sera Zinoboost.

2.3.2 REQUÊTE 2

Cette requête met à jour la colonne DETAILS_EXAMEN de la table EXAMEN pour les enregistrements associés à des consultations de patients ayant un identifiant de patient égal à 1 et où la date de consultation est le 5 février 2024. La nouvelle valeur de la colonne DETAILS_EXAMEN sera HAC1.

3 Description textuelles des requêtes de suppression

3.1 2 requêtes impliquant 1 table

3.1.1 REQUÊTE 1

Cette requête supprime toutes les lignes de la table EXAMEN où la valeur de la colonne DATE_EXAMEN est égale à la date du 5 février 2024.

3.1.2 REQUÊTE 2

Cette requête supprime toutes les lignes de la table PRESCRIPTION où la valeur de la colonne DATE PRESCRIPTION est égale à la date du 5 février 2024.

3.2 2 requêtes impliquant 2 tables

3.2.1 REQUÊTE 1

Cette requête supprime tous les rendez-vous des patients dont l'adresse e-mail est thomas.leclerc@email.com.

3.2.2 REQUÊTE 2

Cette requête supprime tous les rendez-vous du patient dont l'identifiant est 3.

3.3 2 requêtes impliquant plus de 2 tables

3.3.1 REQUÊTE 1

Cette requête supprime tous les examens associés à une consultation qui a eu lieu le 5 février 2024 et qui est liée à un patient dont l'identifiant est 1.

3.3.2 REQUÊTE 2

Cette requête supprime tous les examens associés à une consultation qui a eu lieu le 6 février 2024 et qui est liée à un patient dont l'identifiant est 2.

4 Description textuelles des requêtes de consultation

4.1 5 requêtes impliquant 1 table dont 1 avec un group By et une avec un Order By

4.1.1 REQUÊTE 1

Cette requête récupère toutes les informations (toutes les colonnes) stockées dans la table PATIENT, ce qui signifie qu'elle retournera toutes les lignes de cette table.

4.1.2 REQUÊTE 2

Cette requête récupère toutes les informations (toutes les colonnes) stockées dans la table CONSULTATION, ce qui signifie qu'elle retournera toutes les lignes de cette table.

4.1.3 REQUÊTE 3

Cette requête compte le nombre de lignes dans chaque groupe de données regroupées selon les valeurs uniques de la colonne DETAILS EXAMEN. Chaque groupe dans le résultat final représente une valeur unique de DETAILS EXAMEN, et le nombre de lignes dans chaque groupe est renvoyé.

4.1.4 REQUÊTE 4

Cette requête compte le nombre de lignes dans chaque groupe de données regroupées selon les valeurs uniques de la colonne DETAILS EXAMEN, et les présente dans l'ordre croissant basé sur ces valeurs.

4.1.5 REQUÊTE 5

Cette requête renvoie le nombre total de factures, la somme totale des montants de toutes les factures pour chaque patient, regroupées par ID_PATIENT_, et les présente dans l'ordre croissant des ID_PATIENT_.

4.2 5 requêtes impliquant 2 tables avec jointures internes dont 1 externe + 1 group by + 1 tri

4.2.1 REQUÊTE 1

Cette requête retourne toutes les colonnes des consultations, jointes avec les informations des patients correspondants où les ID PATIENT sont égaux dans les deux tables. Cela permet d'obtenir des données combinées sur les consultations et les patients dans une seule table résultante.

4.2.2 REQUÊTE 2

Cette requête retourne toutes les colonnes des factures, jointes avec les informations des patients correspondants où les ID_PATIENT_ sont égaux dans les deux tables. Cela permet d'obtenir des données combinées sur les factures et les patients dans une seule table résultante.

4.2.3 REQUÊTE 3

Cette requête renvoie toutes les colonnes des consultations, jointes avec les informations des patients correspondants, et les ordonne par date de consultation croissante. Cela permet d'obtenir une liste de consultations associées à leurs patients, triées par date de consultation.

4.2.4 REQUÊTE 4

Cette requête retourne l'identifiant du patient, son adresse e-mail, et la somme des montants totaux de ses factures, regroupés par identifiant de patient et e-mail, et triés par identifiant de patient puis par e-mail. Cela permet d'obtenir une vue agrégée des montants totaux de factures pour chaque patient avec leurs adresses email correspondantes.

4.2.5 REQUÊTE 5

Cette requête retourne toutes les colonnes des consultations et des patients, incluant tous les patients et seulement les consultations qui leur sont associées, triées par date de consultation croissante. Les consultations sans patients associés apparaîtront avec des valeurs NULL dans les colonnes correspondantes de la table "CONSULTATION".

4.3 5 requêtes impliquant plus de 2 tables avec jointures internes dont 1 externe + 1 group by + 1 tri

4.3.1 REQUÊTE 1

Cette requête retourne toutes les colonnes des consultations, des patients et des examens associés, où chaque consultation est liée à son patient correspondant via la jointure avec la table "PATIENT", et chaque consultation est liée à ses examens correspondants via la jointure avec la table "EXAMEN".

4.3.2 REQUÊTE 2

Cette requête retourne toutes les colonnes des factures, des patients et des consultations associées, où chaque facture est liée à son patient correspondant via la jointure avec la table "PATIENT", et chaque facture est liée à sa consultation correspondante via la jointure avec la table "CONSULTATION".

4.3.3 REQUÊTE 3

Cette requête renvoie toutes les colonnes des consultations, des patients et des examens associés, où chaque consultation est liée à son patient correspondant via la jointure avec la table "PATIENT", et chaque consultation est liée à ses examens correspondants via la jointure avec la table "EXAMEN". Les résultats sont ensuite triés par date de consultation croissante.

4.3.4 REQUÊTE 4

Cette requête retourne l'identifiant du patient, son adresse e-mail, la somme des montants totaux de ses factures, ainsi que les dates de la facture et de la consultation correspondantes, regroupés par identifiant de patient, adresse e-mail, date de facture et date de consultation, et triés dans cet ordre. Cela permet d'obtenir une vue agrégée des montants totaux de factures pour chaque patient, avec les détails des factures et des consultations.

4.3.5 REQUÊTE 5

Cette requête retourne toutes les colonnes des consultations, des examens et des patients, où chaque consultation est liée à ses examens correspondants et chaque consultation est également liée à son patient correspondant. Les résultats sont triés par date de consultation croissante. Les patients pour lesquels il n'y a pas de consultation correspondante apparaîtront avec des valeurs NULL dans les colonnes correspondantes de la table "CONSULTATION".

5 Dictionnaire de données MERISE

Entité : PATIENT					
Attributs	Description	Types	Formats	Contraintes	Identifiants
Id_Patient#	Identifiant unique du patient	Entier	Long	Valeur unique, non nulle	Oui
Num_Sec_Social	Numéro Sec. Social du patient	Chaîne	Caractère variable, 50 caractères max	Valeur unique, non nulle	Oui
Sexe	Sexe du patient	Chaîne	Caractère variable, 50 caractères max	non nulle	Oui
Date_Naissance	date de naissance du patient	Date		non nulle	Oui
Nom	Nom de famille du patient	Chaîne	Caractère variable, 50 caractères max	Non null	Non
listPrenoms	Prénoms du patient	ListPrenoms_t			Non
listTelephones	telephones du patient	ListTelephones_t			Non
Adresse	Adresse du patient	Adresse_t		Non null	Non
Email	Email du patient	Chaîne	Caractère variable, 50 caractères max	Non null	Non
Hauteur	Hauteur du patient	Décimal		Non null	Non
Poids	Poids du patient	Décimal		Non null	Non

Entité : FACTURE					
Attributs	Description	Types	Formats	Contraintes	Identifiants
Id_Facture#	Identifiant unique de la facture	Entier	Long	Valeur unique, non nulle	Oui
Montant_Total	Montant total de la facture	Chaîne	Numérique	Non null	Non
Date_Facture	Date d'émission de la facture	Date		Non null	Non
Entité : EXAMEN					
Attributs	Description	Types	Formats	Contraintes	Identifiants
Id_Examen#	Identifiant unique de l'examen	Entier	Long	Valeur unique, non nulle	Oui
Details_Examen	Détails de l'examen	Chaîne	200 caractères max	Non null	Non
Date_Examen	Date de l'examen	Date		Non null	Non
Entité : PRESCRIPTION					
Attributs	Description	Types	Formats	Contraintes	Identifiants
Id_Prescription#	Identifiant unique de la prescription	Entier	Long	Valeur unique, non nulle	Oui
Details_Prescription	Détails de la prescription	Chaîne	200 caractères max	Non null	Non
Date_Prescription	Date de la prescription	Date		Non null	Non
Entité : CONSULTATION					
Attributs	Description	Types	Formats	Contraintes	Identifiants

Id_Consultation#	Identifiant unique de la consultation	Entier	Long	Valeur unique, non nulle	Oui
Raison	Raison de la consultation	Chaîne	Caractère variable, 200 caractères max	Non null	Non
Diagnostic	Diagnostic au terme de la consultation	Chaîne	Caractère variable, 300 caractères max	Non null	Non
Date_Consultation	Date de la consultation	Date		Non null	Non
Entité : RENDEZVOUS					
Attributs	Description	Types	Formats	Contraintes	Identifiants
Id_RendezVous#	Identifiant unique du rendez-vous	Entier	Long	Valeur unique, non nulle	Oui
Motif	Motif du rendezvous	Chaîne	Caractère variable, 200 caractères max	Non null	Non
Date_RendezVous	Date du rendez-vous	Date		Non null	Non
Entité : MEDECIN					
Attributs	Description	Types	Formats	Contraintes	Identifiants
Id_Medecin#	Identifiant unique du médecin	Entier	Long	Valeur unique, non nulle	Oui

Nom	Nom de famille du médecin	Chaîne	Caractère variable, 50 caractères max	Non null	Non
Num_Sec_Social	Numéro Sec. Social du patient	Chaîne	Caractère variable, 50 caractères max	Valeur unique, non nulle	Oui
Sexe	Sexe du patient	Chaîne	Caractère variable, 50 caractères max	non nulle	Oui
listPrenoms	Prénoms du médecin	ListPrenoms_t		Non null	Non
Adresse	Adresse du médecin	Adresse_t		Non null	Non
Email	Email du médecin	Chaîne	Caractère variable, 50 caractères max	Non null	Non
listTelephones	Numéros de téléphone du médecin	ListTelephones_t		Non null	Non
Date_naissance	Date de naissance du médecin	Date		Non null	Non
CV	CV du médecin	CLOB		Non null	Non

TABLE 1: Dictionnaire de données MERISE

6 Descriptions textuelles des associations

Dans cette section sont décrites les associations entre les différentes entités. Une association permet de mettre en relation deux ou plusieurs entités :

6.1 Association « Effectuer » entre MEDECIN et CONSULTATION

Cette association indique que chaque consultation est effectuée par un médecin. Un médecin peut effectuer plusieurs consultations, mais une consultation est effectuée par un seul médecin à la fois.

6.2 Association « Inclure » entre FACTURE et CONSULTATION

Cette association représente le fait qu'une facture inclut une consultation. Chaque consultation peut être incluse dans une seule facture, et réciproquement une facture peut référencer une seule consultation.

6.3 Association « Passer » entre PATIENT et CONSULTATION

Cette association signifie qu'un patient passe une consultation. Chaque consultation est passée par un seul patient, mais un patient peut passer plusieurs consultations.

6.4 Association « Nécessiter » entre CONSULTATION et EXAMEN

Cette association indique que chaque consultation peut nécessiter plusieurs examens. Chaque examen est associé à une seule consultation.

6.5 Association « Recevoir » entre PATIENT et FACTURE

Cette association représente le fait qu'un patient peut recevoir une facture. Chaque facture est destinée à un seul patient, mais un patient peut recevoir plusieurs factures.

6.6 Association « Contenir » entre CONSULTATION et PRESCRIPTION

Cette association signifie qu'une consultation peut contenir plusieurs prescriptions. Chaque prescription est associée à une seule consultation.

6.7 Association « Avoir_Rendez_Vous » entre PATIENT et MEDECIN

Cette association indique que chaque rendez-vous est entre un patient et un médecin. Chaque rendez-vous est pris par un patient avec un médecin spécifique.

6.7.1 Propriétés de l'association Avoir_Rendez_Vous

1. Date_Rendez_Vous

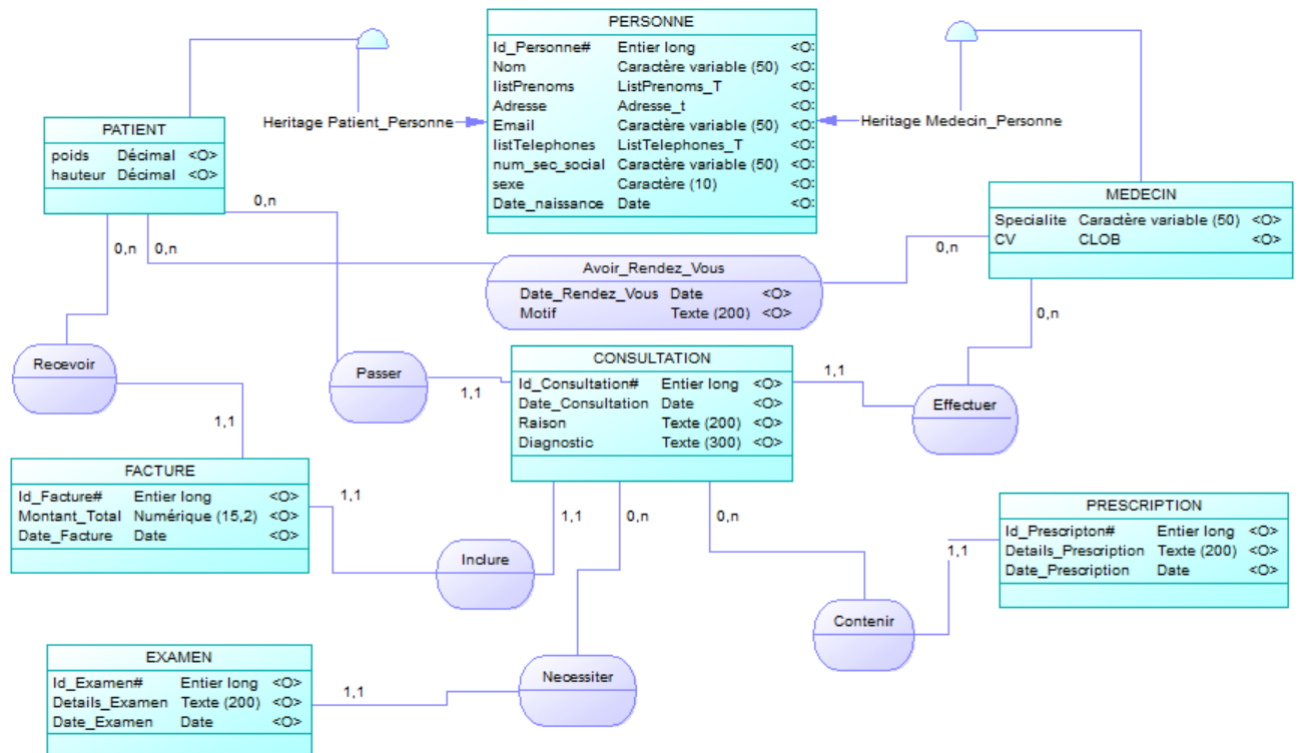
- **Description** : Date retenue pour le rendez-vous
- **Types** : Date
- **Formats** : NA
- **Contraintes** : Non null
- **Identifiants** : Non

2. Motif

- **Description** : Motif justifiant le rendez-vous
- **Types** : Chaîne
- **Formats** : Caractère variable, 50 caractères max
- **Contraintes** : Non null
- **Identifiants** : Non

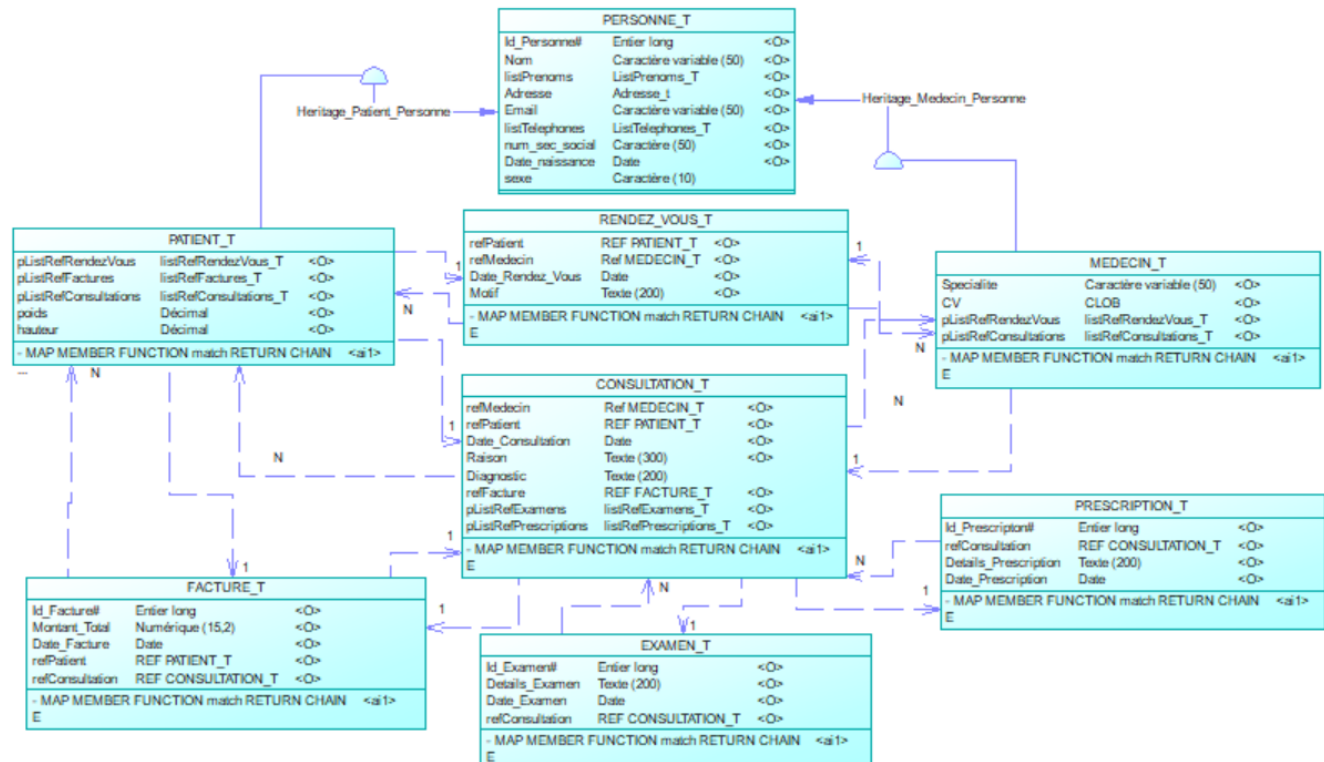
7 Définition du Modèle Entité-Association MERISE

Ci-dessous le Modèle Entité-Association MERISE :



8 Définition du schéma de type à partir du modèle conceptuel

Ci-dessous le schéma de type à partir du modèle conceptuel :



9 Spécification méthodes dans chaque type

N.B. Nous n'avons pas pu mettre les méthodes dans le diagramme/schéma de types au risque de ne pas pouvoir tout afficher. Les méthodes sont donc lister ci-dessous pour chaque type :

9.1 Type PERSONNE_T

— MAP MEMBER FUNCTION compareDate RETURN DATE

9.2 Type PATIENT_T

- STATIC FUNCTION getPatient(idPersonne in number) RETURN PATIENT_T
- STATIC FUNCTION getFActureInfo(idPersonne in number) RETURN set-FACTURES_T
- STATIC FUNCTION getConsultationInfo(idPersonne in number) RETURN setCONSULTATIONS_T
- member PROCEDURE addLinkListeRendezVous(refRendezVous1 REF RENDEZ_VOUS_T)
- member PROCEDURE deleteLinkListeRendezVous(refRendezVous1 REF RENDEZ_VOUS_T)
- member PROCEDURE updateLinkListeRendezVous(refRendezVous1 REF RENDEZ_VOUS_T, refRendezVous2 REF RENDEZ_VOUS_T)
- member PROCEDURE addLinkListeFactures(refFacture1 REF FACTURE_T),
- member PROCEDURE deleteLinkListeFactures(refFacture1 REF FACTURE_T)
- member PROCEDURE updateLinkListeFactures(refFacture1 REF FACTURE_T, refFacture2 REF FACTURE_T)
- member PROCEDURE addLinkListeConsultations(refConsultation1 REF CONSULTATION_T),
- member PROCEDURE deleteLinkListeConsultations(refConsultation1 REF CONSULTATION_T)
- member PROCEDURE updateLinkListeConsultations(refConsultation1 REF CONSULTATION_T, refConsultation2 REF CONSULTATION_T)

9.3 Type MEDECIN_T

- Static FUNCTION getMedecin(idMedecin in NUMBER) RETURN MEDECIN_T
- static FUNCTION getConsultationInfo(idMedecin in NUMBER) RETURN setCONSULTATIONS_T
- static FUNCTION getRendezVousInfo(idMedecin in NUMBER) RETURN setRENDEZ_VOUS_T
- member PROCEDURE addLinkListeRendezVous(refRendezVous1 REF RENDEZ_VOUS_T)
- member PROCEDURE deleteLinkListeRendezVous(refRendezVous1 REF RENDEZ_VOUS_T)
- member PROCEDURE updateLinkListeRendezVous(refRendezVous1 REF RENDEZ_VOUS_T, refRendezVous2 REF RENDEZ_VOUS_T)
- member PROCEDURE addLinkListeConsultations(refConsultation1 REF CONSULTATION_T),
- member PROCEDURE deleteLinkListeConsultations(refConsultation1 REF

CONSULTATION_T)

- member PROCEDURE updateLinkListeConsultations(refConsultation1 REF CONSULTATION_T, refConsultation2 REF CONSULTATION_T)

9.4 Type EXAMEN_T

- MEMBER FUNCTION getId RETURN NUMBER
- MEMBER FUNCTION getDate RETURN DATE
- MAP MEMBER FUNCTION compareDate RETURN DATE
- MEMBER PROCEDURE linkToConsultation(c REF CONSULTATION_T)
- MEMBER FUNCTION getRefConsultation RETURN REF CONSULTATION_T
- MEMBER PROCEDURE updateDetails(newDetails VARCHAR2)
- MEMBER PROCEDURE deleteExamen

9.5 Type FACTURE_T

- MEMBER FUNCTION getId RETURN NUMBER
- MEMBER FUNCTION getDate RETURN DATE
- MAP MEMBER FUNCTION compareDate RETURN DATE
- MEMBER PROCEDURE linkToPatient(p REF PATIENT_T)
- MEMBER PROCEDURE linkToConsultation(c REF CONSULTATION_T)
- MEMBER FUNCTION getRefPatient RETURN REF PATIENT_T
- MEMBER FUNCTION getRefConsultation RETURN REF CONSULTATION_T
- MEMBER PROCEDURE updateMontant(newMontant NUMBER)
- MEMBER PROCEDURE deleteFacture

9.6 Type CONSULTATION_T

- static FUNCTION getConsultation(idConsultation in NUMBER) RETURN CONSULTATION_T
- static FUNCTION getExamenInfo (idConsultation in NUMBER) RETURN setEXAMEN_T
- static FUNCTION getPrescriptionInfo (idConsultation in NUMBER) RETURN setPRESCRIPTION_T,
- member PROCEDURE addLinkListeExamen(refExamen1 REF EXAMEN_T),
- member PROCEDURE deleteLinkListeExamen(refExamen1 REF EXAMEN_T),
- member PROCEDURE updateLinkListeExamen(refExamen1 REF EXAMEN_T, refExamen2 REF EXAMEN_T),

- member PROCEDURE addLinkListePrescription(refPrescription1 REF PRESCRIPTION_T),
- member PROCEDURE deleteLinkListePrescription(refPrescription1 REF PRESCRIPTION_T),
- member PROCEDURE updateLinkListePrescription(refPrescription1 REF PRESCRIPTION_T, refPrescription2 REF PRESCRIPTION_T),
- MAP MEMBER FUNCTION compareDate RETURN DATE

9.7 Type PRESCRIPTION_T

- MEMBER FUNCTION getId RETURN NUMBER
- MEMBER FUNCTION getDate RETURN DATE
- MAP MEMBER FUNCTION compareDate RETURN DATE,
- MEMBER PROCEDURE linkToConsultation(c REF CONSULTATION_T)
- MEMBER FUNCTION getRefConsultation RETURN REF CONSULTATION_T
- MEMBER PROCEDURE updateDetails(newDetails VARCHAR2)
- MEMBER PROCEDURE deletePrescription

9.8 Type RENDEZ_VOUS_T

- MEMBER FUNCTION getId RETURN NUMBER
- MEMBER FUNCTION getDate RETURN DATE
- MAP MEMBER FUNCTION compareDate RETURN DATE
- MEMBER PROCEDURE linkToPatient(p REF PATIENT_T)
- MEMBER PROCEDURE linkToMedecin(m REF MEDECIN_T)
- MEMBER FUNCTION getRefPatient RETURN REF PATIENT_T
- MEMBER FUNCTION getRefMedecin RETURN REF MEDECIN_T
- MEMBER PROCEDURE updateMotif(newMotif VARCHAR2)
- MEMBER PROCEDURE deleteRendezVous

10 Modèle de classe UML

Ci-dessous le diagramme de classes à partir du modèle conceptuel :

