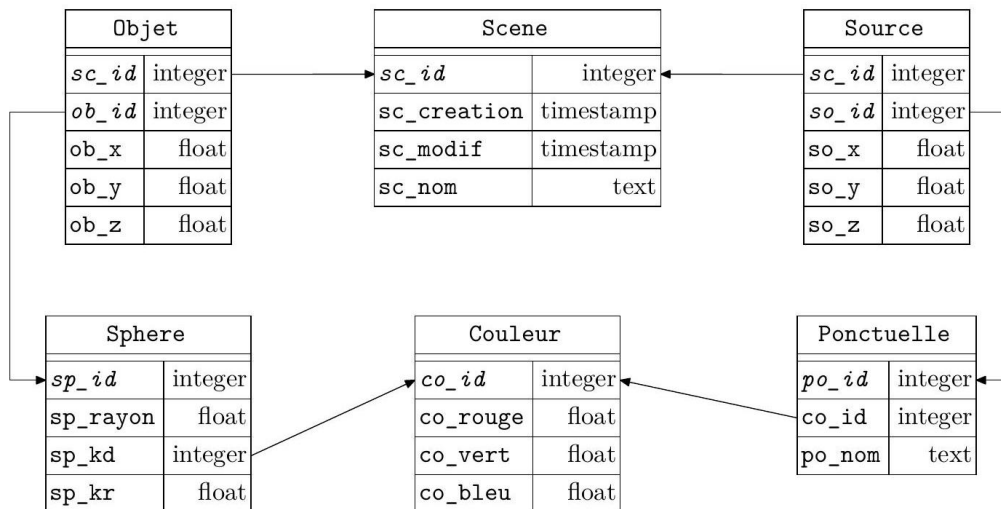


**I - Centrale 2021**

On considère une base de données de structure :



Voici la description des colonnes des six tables :

- la table Scene répertorie les scènes
  - o sc\_id identifiant (entier arbitraire) de la scène (clé primaire)
  - o sc\_creation date de création de la scène
  - o sc\_modif date de dernière modification de la scène
  - o sc\_nom nom de la scène
- la table Couleur définit les couleurs utilisées
  - o co\_id identifiant (entier arbitraire) de la couleur (clé primaire)
  - o co\_rouge valeur de la composante rouge (dans l'intervalle [0, 1])
  - o co\_vert valeur de la composante verte (dans l'intervalle [0, 1])
  - o co\_bleu valeur de la composante bleue (dans l'intervalle [0, 1])
- la table Sphere liste les sphères utilisées pour construire les scènes
  - o sp\_id identifiant (entier arbitraire) de la sphère (clé primaire)
  - o sp\_rayon rayon de la sphère
  - o sp\_kd coefficients de diffusion de la sphère (référence dans la table Couleur)
  - o sp\_kr coefficient de réflexion de la sphère
- la table Ponctuelle répertorie les sources ponctuelles disponibles
  - o po\_id identifiant (entier arbitraire) de la source ponctuelle (clé primaire)

- co\_id couleur de la source (référence dans la table Couleur)
- po\_nom nom de la source
- la table `Objet` fait le lien entre les scènes et les objets qu'elles contiennent, ses deux premières colonnes constituent sa clé primaire
  - sc\_id identifiant de la scène
  - ob\_id identifiant de l'objet
  - ob\_x, ob\_y, ob\_z coordonnées du centre de l'objet dans la scène considérée
- la table `Source` indique quelles sont les sources qui éclairent chaque scène, ses deux premières colonnes constituent sa clé primaire
  - sc\_id identifiant de la scène
  - so\_id identifiant de la source
  - so\_x, so\_y, so\_z coordonnées de la source dans la scène considérée

**Q 14.** Écrire une requête SQL qui donne le nom des scènes créées au cours de l'année 2021.

**Q 15.** Écrire une requête SQL qui donne, pour chaque scène, son identifiant et le nombre de sources qui l'éclairent.

**Q 16.** Écrire une requête SQL qui liste l'identifiant, les coordonnées du centre et le rayon de toutes les sphères contenues dans la scène dont le nom est `woodbox`. On suppose qu'une seule scène possède ce nom.

Il est possible en SQL de définir des fonctions utilisateur qui s'utilisent comme les fonctions SQL pré-définies.

On dispose d'une fonction utilisateur booléenne de signature `OCCULTE(sc_id, objr_id, so_id, objo_id)` où `sc_id`, `objr_id`, `so_id` et `objo_id` sont les identifiants respectifs d'une scène, d'un objet de cette scène dit «récepteur», d'une source de cette scène et d'un autre objet de la scène dit «occultant». La fonction renvoie `TRUE` si l'objet occultant projette son ombre sur l'objet récepteur lorsqu'il est éclairée par la source considérée.

**Q 17.** Écrire une requête SQL qui, pour la scène `woodbox`, renvoie tous les triplets `objr_id`, `so_id`, `objo_id` pour lesquels l'objet d'identifiant `objo_id` occulte la source d'identifiant `so_id` pour l'objet d'identifiant `objr_id`.

## II - Mines 2021 - Marchons, marchons, marchons

Lors de la préparation d'une randonnée, une accompagnatrice doit prendre en compte les exigences des participants. Elle dispose d'informations rassemblées dans deux tables d'une base de données :

- La table `Rando` décrit les randonnées possibles - la clef primaire entière `rid`, le nom du parcours, son niveau de difficulté (entier de 1 à 5), le dénivelé en mètres, la durée moyenne en minutes

rid	rnom	diff	deniv	duree
1	La belle des champs	1	20	30
2	Lac de Castellan	4	650	150
3	Le tour du mont	2	200	120
4	Les crêtes de la mort	5	1200	360
5	Yukon ho !	3	700	210
...	...	...	...	...

- La table `Participant` décrit les randonneurs - la clef primaire entière `pid`, le nom du randonneur, son année de naissance, le niveau de difficulté maximum de ses randonnées

pid	pnom	ne	diff_max
1	Calvin	2014	2
2	Hobbes	2015	2
3	Susie	2014	2
4	Rosalyn	2001	4
...	...	...	...

Donner une requête SQL de cette base pour :

**Q1.** compter le nombre de participants nés entre 1999 et 2003 inclus;

**Q2.** calculer la durée moyenne des randonnées pour chaque niveau de difficulté;

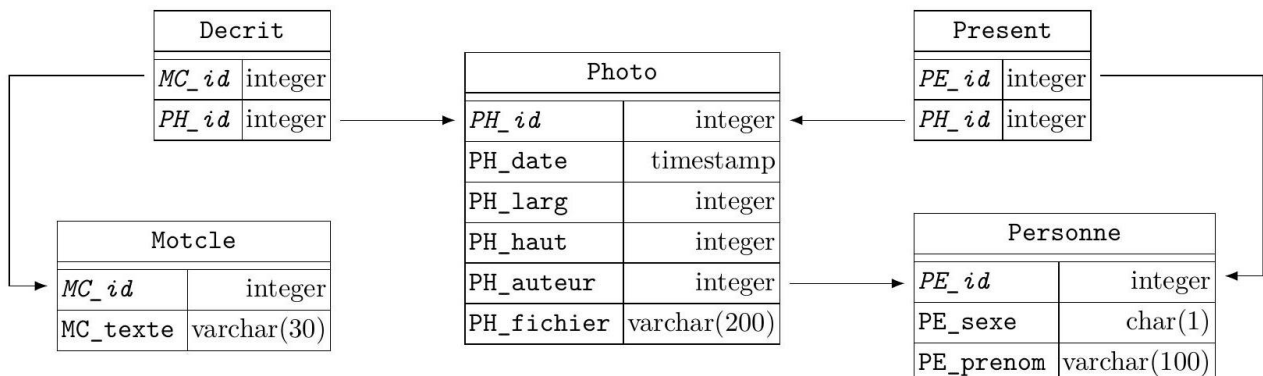
**Q3.** extraire le nom des participants pour lesquels la randonnée  $n^{\circ}42$  est trop difficile;

**Q4.** extraire les clés primaires des randonnées qui ont un ou des homonymes (nom identique et clé primaire distincte), sans redondance.

### III - Centrale 2000

Cette partie est consacrée à une sélection d'images dans une banque d'images.

Les images de la banque sont répertoriées dans une base de données dont le modèle physique est présenté dans la figure ci-dessous, dans laquelle les clés primaires sont notées en italique.



Cette base comporte les cinq tables listées ci-dessous avec la description de leurs colonnes :

- la table **Photo** répertorie les photographies
  - o *PH\_id* identifiant (entier arbitraire) de la photographie (clé primaire)
  - o *PH\_date* date et heure de la prise de vue
  - o *PH\_larg*, *PH\_haut* largeur et hauteur de la photographie en pixels
  - o *PH\_auteur* identifiant de l'auteur de la photographie
  - o *PH\_fichier* nom du fichier contenant la photographie
- la table **Personne** des modèles et des photographes
  - o *PE\_id* identifiant (entier arbitraire) de la personne (clé primaire)
  - o *PE\_sexe* sexe de la personne (M ou F)
  - o *PE\_prenom* prénom de la personne
- la table **Motcle** des mots-clés utilisés pour décrire une photographie
  - o *MC\_id* identifiant (entier arbitraire) du mot-clé (clé primaire)
  - o *MC\_texte* le mot-clé lui-même
- la table **Decrit** fait le lien entre les photographies et les mots-clés qui les décrivent, ses deux colonnes constituent sa clé primaire
  - o *MC\_id* identifiant du mot-clé (décrivant la photographie)
  - o *PH\_id* identifiant de la photographie (décrite par le mot-clé)
- la table **Present** fait le lien entre les photographies et les personnes qui y figurent, ses deux colonnes constituent sa clé primaire
  - o *PE\_id* identifiant de la personne (figurant sur la photographie)
  - o *PH\_id* identifiant de la photographie (représentant la personne)

Pour réaliser les photomosaïques du mariage d'Alice et Bernard, on dispose de plus de 20000 photographies répertoriées dans une base de données.

**Q 19.** Écrire une requête SQL donnant les identifiants de toutes les photographies au ratio 4 :3, c'est-à-dire dont le rapport largeur sur hauteur vaut exactement 4/3.

**Q 20.** Écrire une requête qui compte le nombre de photos prises par Alice ou Bernard.

**Q 21.** Écrire une requête qui fournit l'identifiant et la date des photographies prises avant 2006 et associées au mot-clé «surf».

**Q 22.** Écrire une requête qui donne le prénom de l'auteur et l'identifiant de tous les selfies, c'est-à-dire les photographies sur lesquelles l'auteur est présent.

**Q 23.** Écrire une requête qui sélectionne toutes les photographies sur lesquelles sont présents Alice et Bernard, à l'exclusion de toute autre personne.

Afin de partager et d'enrichir la banque d'images, il a été décidé de faire évoluer la structure de la base de données afin de gérer les mots-clés dans différentes langues. Le cahier des charges de cette évolution stipule :

- a. l'ensemble des photographies sélectionnées à l'aide de mots-clés ne doit pas dépendre de la langue utilisée pour exprimer ces mots-clés ; autrement dit, les photographies décrites par le mot-clé «montagne» exprimé en français doivent être les mêmes que celles sélectionnées par les mots-clés «mountain» si la langue choisie est l'anglais, «Berg» pour l'allemand, «montaña» pour l'espagnol, etc. ;
- b. il doit être possible, pour cette nouvelle base de données, d'écrire une requête de recherche de photographies par mot-clé en spécifiant la langue utilisée pour exprimer le mot-clé de telle sorte que changer de langue se fasse en modifiant uniquement des constantes dans la clause WHERE.

**Q 24.** Proposer un nouveau modèle de base de données répondant à cette évolution du cahier des charges en ne détaillant que ce qui change (tables modifiées, nouvelles tables).

**Q 25.** Avec cette nouvelle base de données, écrire une requête qui permet de sélectionner les identifiants des photographies associées au mot-clé «mountain» exprimé en anglais.

## IV - Centrale 2018 - cinétique des gaz parfaits

On dispose d'une version plus générale de la fonction simulation pour laquelle toutes les particules ne sont plus nécessairement identiques. Cette fonction enregistre ses résultats dans une base de données dont la structure est donnée par la figure suivante.

SIMULATION	
SI_NUM	integer
SI_DEB	datetime
SI_DUR	float
SI_DIM	integer
SI_N	integer
SI_L	float

REBOND	
SI_NUM	integer
RE_NUM _	integer
PA_NUM _	integer
RE_T _	float
RE_DIR	integer
RE_VIT	float
RE_P _	float

PARTICULE	
PA_NUM	integer
PA_NOM	varchar(100)
PA_M	float
PA_R	float

Figure 4 Structure physique de la base de données des résultats de simulation

Cette base comporte les trois tables suivantes :

- la table SIMULATION donne les caractéristiques de chaque simulation effectuée. Elle contient les colonnes
  - SI\_NUM numéro d'ordre de la simulation (clef primaire)
  - SI\_DEB date et heure du lancement du programme de simulation
  - SI\_DUR durée (en secondes) de la simulation (il ne s'agit pas du temps d'exécution du programme, mais du temps simulé)
  - SI\_DIM nombre de dimensions de l'espace de simulation
  - SI\_N nombre de particules pour cette simulation
  - SI\_L (en mètres) taille du récipient utilisé pour la simulation
- la table PARTICULE des types de particules considérées. Elle contient les colonnes
  - PA\_NUM numéro (entier) identifiant le type de particule (clef primaire)

- PA\_NOM nom de ce type de particule
  - PA\_M masse de la particule (en grammes)
  - PA\_R rayon (en mètres) de la particule
- la table REBOND, de clef primaire (SI\_NUM, RE\_NUM), liste les chocs des particules avec les parois du récipient. Elle contient les colonnes
- SI\_NUM numéro d'ordre de la simulation ayant généré ce rebond
  - RE\_NUM numéro d'ordre du rebond au sein de cette simulation
  - PA\_NUM numéro du type de particule concernée par ce rebond
  - RE\_T temps de simulation (en secondes) auquel ce rebond est arrivé
  - RE\_DIR paroi concernée : entier non nul de l'intervalle  $[-SI\_DIM, SI\_DIM]$  donnant la direction de la normale à la paroi. Ainsi  $-2$  désigne la paroi située en  $y = 0$  alors que  $1$  désigne la paroi située en  $x = L$
  - RE\_VIT norme de la vitesse de la particule qui rebondit (en  $m.s^{-1}$ )
  - RE\_VP valeur absolue de la composante de la vitesse normale à la paroi (en m.s)

**Q.34.** Écrire une requête SQL qui donne le nombre de simulations effectuées pour chaque nombre de dimensions de l'espace de simulation.

**Q.35.** Écrire une requête SQL qui donne, pour chaque simulation, le nombre de rebonds enregistrés et la vitesse moyenne des particules qui frappent une paroi.

**Q.36.** Écrire une requête SQL qui, pour une simulation  $n$  donnée, calcule, pour chaque paroi, la variation de quantité de mouvement due aux chocs des particules sur cette paroi tout au long de la simulation. On se rappellera que lors du rebond d'une particule sur une paroi la composante de sa vitesse normale à la paroi est inversée, ce qui correspond à une variation de quantité de mouvement de  $2m|v_{\perp}|$  où  $m$  désigne la masse de la particule et  $v_{\perp}$  la composante de sa vitesse normale à la paroi.

## V - Mines 2018 - Mesure de la houle par des capteurs sur des bouées

Plusieurs indicateurs sont couramment considérés pour définir l'état de la mer. Parmi eux, on note :

- $H_{\max}$  : la hauteur de la plus grande vague observée sur l'intervalle d'enregistrement  $[0, T]$ ;
- $H_{1/3}$  : la valeur moyenne des hauteurs du tiers supérieur des plus grandes vagues observées sur  $[0, T]$ ;
- $T_{H_{1/3}}$  : la valeur moyenne des périodes du tiers supérieur des plus grandes vagues observées sur  $[0, T]$ .

On dispose d'une base de données relationnelle Vagues.

La première table est Bouee. On se limite aux attributs suivants : le numéro d'identification `idBouee`, le nom du site `nomsite`, le nom de la mer ou de l'océan `localisation`, le type du capteur `typeCapteur` et la fréquence d'échantillonnage `frequence`.

Bouee

idBouee	nomsite	localisation	typeCapteur	frequence
831	Porquerolles	Mediterranee	Datawell non directionnelle	2.00
291	Les pierres noires	Mer d'iroise	Datawell directionnelle	1.28
...	...	...	...	...

La seconde table est Campagne. On se limite aux attributs suivants : le numéro d'identification `idCampagne`, le numéro d'identification de la bouée `idBouee`, la date de début `debutCampagne` et la date de fin `finCampagne`.

Campagne

idCampagne	idBouee	debutCampagne	finCampagne
08301	831	01/01/201000h00	15/01/201000h00
02911	291	15/10/200518h30	18/10/200508h00
...	...	...	...

La troisième table est Tempete. Les informations fournies relatives à un événement «tempête» sont les suivantes :

- date de début et fin de tempête;
- évolution des paramètres  $H_{1/3}$  et  $H_{\max}$  en fonction du temps
- Le détail de certains paramètre non définis ici, obtenus au pic de tempête.

On se limite aux attributs suivants : le numéro d'identification de la tempête  $idTempete$ , le numéro d'identification de la bouée  $idBouee$ , la date de début  $debutTempete$ , la date de fin  $finTempete$ , la valeur maximale de hauteur de vague  $Hmax$ .

Tempete

$idTempete$	$idBouee$	$debutTempete$	$finTempete$	$Hmax$
083010	831	07/01/201020h00	09/01/201015h30	5.3
029012	291	16/10/200508h30	18/10/200509h00	8.5
...	...	...	...	

Le schéma de la base de donnée est donc : Vagues = Bouee, Campagne, Tempete.

**Q19.** Formuler les requêtes SQL permettant de répondre aux questions suivantes :

- Quels sont le numéro d'identification et le nom de site des bouées localisées en Méditerranée?
- Quel est le numéro d'identification des bouées où il n'y a pas eu de tempêtes?
- Pour chaque site, quelle est la hauteur maximale enregistrée lors d'une tempête?

## VI - X 2017 - ensemble de points

On suppose que l'on représente des points à coefficients entiers dans le plan à l'aide d'une base de données. Cette base comporte deux tables. La table POINTS contient trois colonnes :

- $id$  (clé primaire) qui est un entier naturel unique représentant le point;
- $x$  qui est un entier naturel représentant son abscisse;
- $y$  qui est un entier naturel représentant son ordonnée.

On suppose qu'il n'existe pas deux points d'identifiants distincts et de mêmes coordonnées.

La relation d'appartenance à un ensemble de points est représentée par la table MEMBRE à deux colonnes :

- $idpoint$ , un entier naturel qui identifie un point;
- $idensemble$ , un entier naturel qui identifie un ensemble de points.

**Q4.** Écrire une requête SQL qui renvoie les identifiants des ensembles auxquels appartient le point de coordonnées  $(a, b)$

**Q5.** Écrire une requête SQL qui renvoie les coordonnées des points qui appartiennent à l'intersection des ensembles d'identifiants  $i$  et  $j$ .

**Q6.** Écrire une requête SQL qui renvoie les identifiants des points appartenant à au moins un des ensembles auxquels appartient le point de coordonnées  $(a, b)$ .

## VII - X MP PC 2016 - réseaux sociaux

Une représentation simplifiée, réduite à deux tables, de la base de données d'un réseau social est donnée dans la figure suivante :

$id$	$nom$	$prenom$
1	Potter	Harry
2	Granger	Hermione
...	...	...

$id1$	$id2$
1	2
2	1
...	...

La table INDIVIDUS répertorie les individus et contient les colonnes :

- $id$  (clé primaire), un entier identifiant chaque individu;
- $nom$  une chaîne de caractères donnant le nom de famille de l'individu;
- $prenom$  une chaîne de caractères donnant le prénom de l'individu.

La table LIENS répertorie les liens d'amitiés entre individus et contient les colonnes :

- id1 entier identifiant le premier individu du lien d'amitié;
- id2 entier identifiant le second individu du lien d'amitié.

On supposera par ailleurs que pour tout couple  $(x, y)$  dans la table LIENS, le couple  $(y, x)$  est également présent dans la table.

**Q16.** Écrire une requête SQL qui renvoie les identifiants des amis de l'individu d'identifiant  $x$ .

**Q17.** Écrire une requête SQL qui renvoie les nom et prénom de chacun des amis de l'individu d'identifiant  $x$ .

**Q18.** Écrire une requête SQL qui renvoie les identifiants des individus qui sont amis avec au moins un ami de l'individu d'identifiant  $x$ .

## VIII - Mines 2016 : propagation d'épidémies

Pour suivre la propagation des épidémies, de nombreuses données sont recueillies par les institutions internationales comme l'OMS. Par exemple, pour le paludisme, on dispose de deux tables :

- La table palu recense le nombre de nouveaux cas confirmés et le nombre de décès liés au paludisme; certaines lignes de cette table sont données en exemple (on précise que iso est un identifiant unique pour chaque pays) :

nom	iso	annee	cas	deces
Bresil	BR	2009	309316	85
Bresil	BR	2010	334667	76
Kenya	KE	2010	898531	26017
Mali	ML	2011	307035	2128
Ouganda	UG	2010	1581160	8431

- la table demographie recense la population totale de chaque pays; certaines lignes de cette table sont données en exemple :

pays	periode	pop
BR	2009	193020000
BR	2010	194946000
KE	2010	40909000
ML	2011	14417000
UG	2010	33987000

**Q5.** Au vu des données présentées dans la table palu, parmi les attributs nom, iso et annee, quels attributs peuvent servir de clé primaire? Un couple d'attributs pourrait-il servir de clé primaire? (on considère qu'une clé primaire peut posséder plusieurs attributs). Si oui, en préciser un.

**Q6.** Écrire une requête en langage SQL qui récupère depuis la table palu toutes les données de l'année 2010 qui correspondent à des pays où le nombre de décès dus au paludisme est supérieur ou égal à 1000 .

On appelle taux d'incidence d'une épidémie le rapport du nombre de nouveaux cas pendant une période donnée sur la taille de la population-cible pendant la même période. Il s'exprime généralement en «nombre de nouveaux cas pour 100000 personnes par année». Il s'agit d'un des critères les plus importants pour évaluer la fréquence et la vitesse d'apparition d'une épidémie.

**Q7.** Écrire une requête en langage SQL qui détermine le taux d'incidence du paludisme en 2011 pour les différents pays de la table palu.

**Q8.** Écrire une requête en langage SQL permettant de déterminer le nom du pays ayant eu le deuxième plus grand nombre de nouveaux cas de paludisme en 2010 (on pourra supposer qu'il n'y a pas de pays *ex aequo* pour les nombres de cas

## IX - Mines 2015 - Tests de validation d'une imprimante

Une représentation simplifiée de deux tables de la base de données qu'on souhaite utiliser est donnée ci-dessous :

nSerie	dateTest	...	Imoy	Iec	...	fichierMes
230 - 588ZX2547	2012 - 04 - 2214 - 25 - 45		0.45	0.11		mesure31025.csv
230 - 588ZX2548	2012 - 04 - 2214 - 26 - 57		0.43	0.12		mesure41026.csv
...	...	...	...	...	...	...

Num	nSerie	dateProd	type
20	230 - 588ZX2547	2012 - 04 - 2215 - 52 - 12	JETDESK-1050
21	230 - 588ZX2549	2012 - 04 - 2215 - 53 - 24	JETDESK-3050
...	...	...	...

Après son assemblage et avant les différents tests de validation, un numéro de série unique est attribué à chaque imprimante. À la fin des tests de chaque imprimante, les résultats d'analyse ainsi que le fichier contenant l'ensemble des mesures réalisées sur l'imprimante sont rangées dans la table `testfin`. Lorsqu'une imprimante satisfait les critères de validation, elle est enregistrée dans la table `production` avec son numéro de série, la date et l'heure de sortie de production ainsi que son type.

**Q8.** Rédiger une requête SQL permettant d'obtenir les numéros de série des imprimantes ayant une valeur de `Imoy` comprise strictement entre deux bornes `Imin` et `Imax`.

**Q9.** Rédiger une requête SQL permettant d'obtenir les numéros de série, la valeur de l'écart `type` et le fichier de mesures des imprimantes ayant une valeur de `Iec` strictement inférieure à la valeur moyenne de la colonne `Iec`.

**Q10.** Rédiger une requête SQL qui permettra d'extraire à partir de la table `testfin` le numéro de série et le fichier de mesures correspondant aux imprimantes qui n'ont pas été validées en sortie de production.