|  |  |
| --- | --- |
| **BTS SERVICES INFORMATIQUES AUX ORGANISATIONS** | **Entrainement 2018** |

**E5 : PRODUCTION ET FOURNITURE DE SERVICES INFORMATIQUES**

**Durée : 4h**

|  |  |
| --- | --- |
|  |  |

Cas pour ENC

## Barème

|  |  |  |  |
| --- | --- | --- | --- |
| **SLAM3** | **Mission 1** | Évaluation du logiciel Musicol |  |
| **SLAM4** | **Mission 2** | Adaptation du logiciel Musicol |  |
| **SLAM3** | **Mission 3** | Évolution du logiciel Musicol |  |
|  |  | **Total** | **100 points** |

Présentation du contexte

#### L'organisation cliente : l'école de musique municipale de la ville de D.

Inscrite dans le paysage culturel de la ville de D. (15 000 habitants) depuis une quarantaine d'années, l'école de musique municipale compte aujourd'hui quelques 300 usagers.

Ouverte à tous, enfants, adolescents, adultes, et couvrant le territoire de la communauté de communes, elle a pour vocation l'apprentissage de la musique.

Son projet d’établissement est axé vers les grandes missions suivantes :

* formation des musiciens (instrumentale, musicale, orchestrale, ateliers, etc.) ;
* diffusion, création (concerts, auditions) ;
* sensibilisation d’un large public et développement de partenariats locaux (éducation nationale, scène nationale voisine, maison des jeunes et de la culture, centres d’animation, pratiques amateurs, *master class*, conférences).

Dotée de locaux de qualité, l’école de musique dispose d’une équipe pédagogique de seize enseignants et propose au public différentes prestations d'enseignement musical.

L’école dispense des cours d’instruments individuels ou collectifs pour enfants, adolescents et adultes, en fonction des places disponibles. L’apprentissage est accessible à partir de 7 ans, sauf précision particulière pour certains instruments.

#### L'entreprise prestataire de services

La DSI - direction des systèmes d’information de la mairie de D. - regroupe deux personnes à temps plein afin de répondre aux besoins des différentes entités de la mairie réparties sur une dizaine de sites.

L’école de musique municipale est placée sous la tutelle de la mairie et profite à ce titre de services informatiques fournis par la DSI : bureautique, messagerie, accès au réseau et à internet, maintenance préventive des postes de travail et assistance aux utilisateurs.

La DSI assure également la fourniture de services spécifiques à l’école de musique : gestion des inscriptions, facturation et suivi des règlements.

#### Le projet

Avec les outils informatiques actuellement en place, la gestion de l’école de musique est jugée peu efficace par les personnels de l’école. La DSI de la mairie mène une étude en vue de moderniser ces outils.

Intégré-e en tant que salarié-e à la DSI, vous assistez M. Lejoyau, responsable de la DSI, et participez à cette étude et aux différentes missions qui la composent.

Un dossier documentaire vous est remis pour compléter votre information.

Mission 1 : Évaluation du logiciel Musicol

Documents à utiliser : 1, 2, 3 et 4

Sur la forge logicielle de l’Adullact (association des développeurs et des utilisateurs de logiciels libres pour l'administration et les collectivités territoriales), M. Lejoyau a pu observer l’existence du logiciel nommé *Musicol*. Il s'agit d'une application *web* développée en PHP autour du cadriciel (*framework*) libre nommé *CakePHP* basé sur une architecture MVC (modèle vue contrôleur).

Compte-tenu des réflexions déjà engagées, le responsable de la DSI pense qu’une évaluation de ce produit permettra de vérifier à quel degré il peut convenir au besoin. Vous commencez par télécharger la documentation du logiciel *Musicol*. Dans le manuel du développeur figure le schéma relationnel de la base de données utilisée par l’application ainsi qu’une copie des scripts de création de la base de données.

Mme Lebec, quant à elle, a besoin pour la gestion de l'école de connaître les éléments suivants :

* l'intitulé des instruments qui ne font l'objet d'aucun cours ;
* pour chaque instrument, le nombre d'élèves inscrits à des cours (par exemple : Batterie 7, Piano 19 …) ;
* le libellé des cours contenant moins de 3 inscrits.

Votre mission consiste à évaluer si *Musicol* couvre les besoins fonctionnels de l'école de Musique.

### Votre mission

1. **Indiquer s’il est possible d’enregistrer dans la base de données un paiement en 4 fois comme le pratique l'école. Si non, proposer les modifications à effectuer pour le permettre, et implémentez les en SQL / Transact SQL**
2. **Indiquer si un élève peut s’inscrire à plusieurs cours. Si non, proposer les modifications à effectuer pour le permettre.**
3. **Écrire les requêtes, en langage SQL sur Microsoft SQL Server, permettant d'obtenir les éléments dont Mme Lebec a besoin, et fournissez les résultats**
4. **A quoi sert le script « Requetes pour Tranches - Analyse.sql » ?**

Mission 2 : Adaptation du logiciel Musicol

Documents à utiliser : 4, 5, 6, 7, 8, 9, 10

*IMPORTANT : le code sera impérativement du PHP*

M. Lejoyau a engagé auparavant un prestataire, M. Robert, en qui il semble hasardeux de placer toute sa confiance. En revanche, conscient de ses limites, a laissé quelques tests unitaires.

M. Lejoyau compte sur vous pour compléter et tester ce qu’a fait M. Robert.

En outre, comme décrit dans la plaquette de présentation de l’école, l’école de musique propose :

* des cours individuels sur un instrument en particulier et
* des cours collectifs.

Afin de différencier ces deux types de cours, M. Lejoyau a adapté la couche métier associée et produit le diagramme de classes métiers et le code PHP de ces classes.

De même un planning des différents cours dispensés chacun des jours de la semaine étant établi chaque année, M. Lejoyau a ajouté une classe Planning.

Un programme nommé analysePlanning devrait permettre également de valider l’écriture des méthodes manquantes.

Votre mission consiste à adapter la solution à ces modifications.

### Votre mission

1. Finir le constructeur de la classe Planning et initialiser également correctement la collection $lesJours
2. **Écrire le code des méthodes *getLibelle* des classes *CoursCollectif* et *CoursIndividuel* permettant de retourner le libellé du cours.**
3. **Écrire la méthode *aLieuLeJour* de la classe *Cours*.**
4. **Écrire la méthode *coursJour* de la classe *Planning.***
5. Finir les tests unitaires concernant la classe planning
6. **Indiquer de manière détaillée ce que retourne la méthode *CoursSemaine* de la classe *Planning*.**

Mission 3 : Évolution du logiciel Musicol

Documents à utiliser : 11, 12, 13, 14

*IMPORTANT : la candidate ou le candidat présentera les évolutions de la structure de la base de données sous la forme d’un diagramme de classes UML.*

La communauté de communes alloue depuis de nombreuses années un budget annuel pour la maintenance et le renouvellement d'un parc d'instruments de musique. Ce parc est destiné à deux utilisations distinctes :

* des instruments qui restent à l'école de musique, par exemple des instruments ne pouvant facilement être déplacés (piano, harpe…) ou alors des instruments mis à disposition des professeurs pour leurs cours et ateliers ;
* des instruments qui sont prêtés gracieusement aux élèves débutants, ce qui évite aux familles d'investir, dès le début de l'apprentissage, dans un instrument souvent coûteux.

Pour le moment, la gestion du parc est manuelle et s'effectue de la façon suivante :

* Chaque instrument possède une étiquette discrètement collée sur laquelle est inscrit un code (par exemple le code **"*bas-2*"** signifie qu'il s'agit de la 2ème basse électrique du parc) et une fiche sur laquelle figure la description détaillée, une photo ainsi que les interventions de réglage et de réparation.
* Pour les instruments de prêt, une deuxième fiche récapitule les prêts aux élèves.

Dans le cadre de la maintenance du parc, l'école de musique travaille avec plusieurs professionnels partenaires (accordeurs de piano, luthiers, facteurs de divers instruments, …) à qui l'on confie les instruments pour des réglages ou des réparations. Au retour de chaque intervention, la fiche de l'instrument concerné est alors complétée.

En début d'année, les familles qui désirent emprunter un instrument font une demande motivée (non stockée dans la base de données). Une commission se réunit pour attribuer les instruments, la priorité étant donnée aux débutants. Le jour de la remise de l'instrument, l'école de musique établit un contrat avec la famille qui contient les éléments suivants :

* la date de signature et la date de fin de contrat généralement située vers le 7 septembre de l'année suivante ;
* un descriptif de l'état de l'instrument qui fait suite à une vérification minutieuse effectuée en présence de la famille dans lequel chaque détail est noté (par exemple un petit enfoncement dû à un choc,…) ;
* le numéro de la police d'assurance de la famille qui couvre les risques liés à l’usage de l'instrument (casse, perte ou vol).

Il est à noter qu'un instrument de prêt est fourni avec des accessoires (housse, câble jack, archet, amplificateur…) indispensables pour sa pratique et sa protection. Lors de la restitution de l'instrument, un an plus tard, le contrat est complété par l'état global. Si l'instrument doit subir un réglage ou une réparation non due à une usure normale, l'école de musique est en droit de facturer tout ou partie du prix de l'intervention, dans ce cas on notera la quotité demandée à la famille (100% pour tout le montant, 50% pour la moitié du montant,…).

### Votre mission

1. **Fournir un schéma de données présentant les évolutions de la structure de la base**

**de données relatives à la gestion du parc d'instruments : acquisitions, prêts,**

**Interventions de réglage et de réparation.**

1. **Donner l'instruction (ou les instructions) permettant d'ajouter concrètement une catégorie d'instruments d'une classe d'instrument donnée : Type : « Guitare classique », classe : « Corde frottée »**
2. **Donner l'instruction (ou les instructions) permettant de connaître le nombre de fois qu'un responsable d'élève est impliqué dans une opération d'intervention sur un instrument.**

**Ressources documentaires**

# Script pour questions mission 1

Programme : 005-Scripts pour données.sql

# Script Trigger pour question 1 mission 1

CREATE TRIGGER music.INSC\_INSERT

ON music.INSCRIPTIONS

AFTER INSERT AS

BEGIN

DECLARE @NEW\_NOMBRE\_PAIEMENTS INT ;

SELECT @NEW\_NOMBRE\_PAIEMENTS = nombre\_de\_paiements FROM inserted ;

IF @NEW\_NOMBRE\_PAIEMENTS < 1 or @NEW\_NOMBRE\_PAIEMENTS > 3

BEGIN

ROLLBACK TRANSACTION ;

THROW 99999,'Insertion interdite',1;

END;

END;

# Script « Requete pour Tranches.sql » pour question 4 mission 1

CREATE VIEW music.INTERVALLE\_TRANCHES (ID\_TRANCHE , QUOTIENTMIN , QUOTIENTMAX )

AS SELECT A.ID AS ID\_TRANCHE, A.QUOTIENT\_MIN AS QUOTIENTMIN,

CASE WHEN B.QUOTIENT\_MIN IS NULL THEN 10000

ELSE B.QUOTIENT\_MIN END AS QUOTIENTMAX

FROM music.TRANCHES AS A

LEFT JOIN music.TRANCHES AS B

ON A.ID = B.ID-1

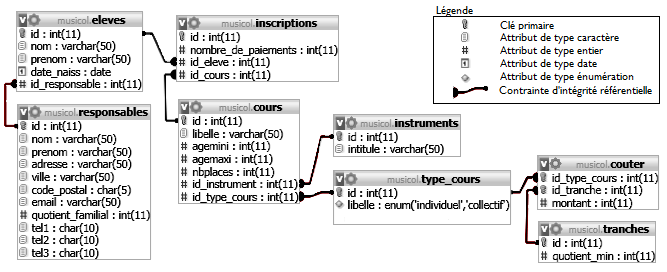
;

# Tarifs des tranches de quotient familial

|  |  |  |  |
| --- | --- | --- | --- |
| **Q.F.** | **Tranche** | **Cours individuel** | **Cours collectif** |
| ***Résidents de la communauté de communes*** | | | |
| 0 à 250 | A | 60 | 30 |
| 251 à 425 | B | 96 | 48 |
| 426 à 680 | C | 126 | 63 |
| 681 à 934 | D | 192 | 96 |
| 935 à 1800 | E | 282 | 141 |
| + de 1800 | F | 330 | 165 |

Pour information.

# Schéma relationnel initial de la base de données Musicol

****

**responsables (**id, nom, prenom, adresse, ville, code\_postal, email, quotient\_familial, tel1, tel2, tel3**)**

*Clé primaire : id*

**eleves (**id, nom, prenom, date\_naiss, id\_responsable**)**

*Clé primaire : id*

*Clé étrangère : id\_*responsable *en référence à id de responsables*

**instruments** (id, intitule)

*Clé primaire : id*

**type\_cours** (id, libelle) *//libellé a pour valeur* ***individuel*** *ou* ***collectif***

*Clé primaire : id*

**cours** (id, libelle, agemini, agemaxi, nbplaces, id\_instrument, id\_type\_cours)

*Clé primaire : id*

*Clé étrangère : id\_instrument en référence à id de instruments*

*Clé étrangère : id\_type\_cours en référence à id de type\_cours*

**tranches** (id, quotient\_min)

*Clé primaire : id*

**couter** (id\_type\_cours, id\_tranche, montant)

*Clé primaire : id\_type\_cours, id\_tranche*

*Clé étrangère : id\_tranche en référence à id de tranches*

*Clé étrangère :* id\_type\_cours *en référence à id de type\_cours*

**inscriptions** (id, nombre\_de\_paiements, id\_eleve, id\_cours)

*Clé primaire : id*

*Clé étrangère : id\_eleve en référence à id de eleves*

*Clé étrangère : id\_cours en référence à id de cours*

# Écran de présentation de la grille tarifaire (produit par *grilleView.php*)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tarifs de l'année 2014**   |  |  |  | | --- | --- | --- | | **Quotient familial** | **Cours individuel** | **Cours collectif** | | **0 à 250** | **60** | **30** | | **251 à 425** | **96** | **48** | | **426 à 680** | **126** | **63** | | **681 à 935** | **192** | **96** | | **936 à 1800** | **282** | **141** | | **1801 et supérieur** | **330** | **165** | |

# Éléments de documentation PHP

Implémentation d’une collection d’objets :

$maCollection = array() ; *// Création de la collection*

$maCollection[] = $objetUn ; *// Ajout d’un objet à la collection*

$maCollection[] = $objetDeux ;

$maCollection[] = $objetTrois ;

foreach($maCollection as $unObjet) *// Parcours de la collection*

{

*// Utilisation de l’objet $unObjet*

echo $unObjet->getLibelle() ; *// Appel d’une méthode de l’objet*

}

Implémentation d’un dictionnaire<clé, valeur> de <chaîne, objet> :

*Remarque : une particularité du langage PHP est de permettre de parcourir un dictionnaire en récupérant à la fois la clé et la valeur de chaque élément.*

$monDico = array() ; *// Création du dictionnaire, ou tableau associatif*

$monDico["un"] = $unObjet1 ; *// Ajout du couple <"un", unObjet1> au dictionnaire*

$monDico["deux"] = $unObjet2 ; *// Ajout du couple <"deux", unObjet2  > au dictionnaire*

$monDico["trois"] = $unObjet3 ; *// Ajout du couple <"trois", unObjet3  > au dictionnaire*

$unObjet = $monDico["deux"] ; *// $unObjet est affecté de unObjet2*

if (isset($monDico["deux"]))… *// Ce test retourne vrai, car la clé "deux" existe*

if (isset($monDico["quatre"]))… *// Ce test retourne faux, la clé "quatre" n’existe pas*

foreach($monDico as $laCle=>$unObjet) *// Parcours du dictionnaire*

{

// Utilisation de l’objet $unObjet

echo $laCle; *// Affichage de la clé*

echo $unObjet->uneMethode(); *// Appel d’une méthode de l’objet*

}

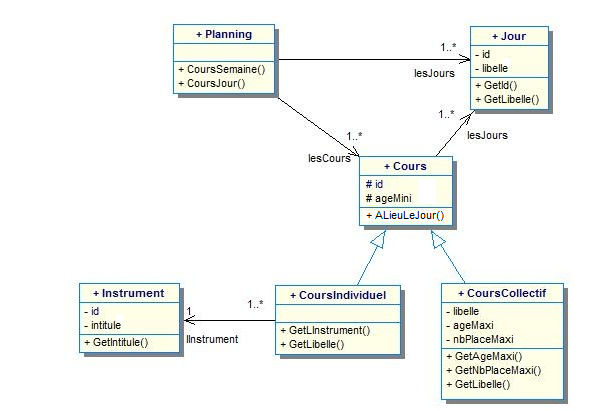
# Présentation et code de la page *grilleView*.*php*

La vue ***grilleView.php*** permet l’affichage de la grille tarifaire. Elle dispose de l’année courante et des 3 tableaux suivants :

* Le tableau associatif (ou dictionnaire) **$lesTranches** contient l’ensemble des tranches de quotient familial : l’indice est l’identifiant de tranche ; la valeur est un objet de la classe *Tranche* exposant deux méthodes publiques pour obtenir l’une le quotient minimum de la tranche, l’autre le quotient maximum de la tranche.
* Le tableau **$lesTarifs** contient l’ensemble des tarifs de l’école, présentant deux dimensions : la première est l’identifiant du type de cours, la seconde est l’identifiant de la tranche. La valeur contenue est le tarif associé. Par exemple : *$lesTarifs[1][2]* contient le tarif du type de cours 1 pour la tranche 2 (quotient compris entre 251 et 425).
* Le tableau associatif (ou dictionnaire) **$lesTypesCours** contient l'ensemble des types de cours : l'indice est l'identifiant de typeCours ; la valeur est un objet de la classe *TypeCours*.

Le script est fourni, pour information, non utilisé.

# Diagramme de classes pour la planification des cours



# Code PHP des classes métier pour la planification des cours

**Toutes les classes sont fournies. Le prestataire Mr Robert semble avoir préparé un fichier testPlanning.php, en complément des tests unitaires, destiné à valider la bonne implémentation des classes (et méthodes à compléter) .**

Cf répertoire sources.

# Fiche de renseignement d'un instrument

**bas-2** (Basse électrique)



Date d'achat : 06/06/2011

Prix d'achat : 380 €

Marque : Squier Modèle : Jazz Bass

Numéro de série : Q 07219

Couleur dominante : Noire

Utilisation : prêt

**Classe d'instruments :** Instruments amplifiés

**Interventions**

**N° Date début Date fin Professionnel Travaux effectués Prix**

1 12/09/2013 14/09/2013 Hall de musique Réglage du manche 30€

24 rue Amiral Ronarc'h

29000 Quimper

2 12/09/2014 12/09/2014 Ets Poënot Réparation de

Port Rhu la prise jack 10€

29100 Douarnenez

3 14/09/2014 15/09/2014 Hall de musique Réglage du manche 32€

24 rue Amiral Ronarc'h

29000 Quimper

# Fiche de prêts d'un instrument

**bas-2** (Basse électrique)



Accessoires fournis : housse souple, jack, ampli 10W

**Prêts**

**Elève Date début Date fin N° intervention quotité**

**suite à un prêt famille**

Horellou Simon 14/09/2011 07/09/2012

15, rue Ar Ster

29100 Pouldergat

Queffelec Annaël 15/09/2012 08/09/2013 1 0%

31 rue Édouard Branly

29100 Douarnenez

Le Boulch Thibault 16/09/2013 09/09/2014 2 100%

56 rue Victor Hugo 3 0%

29100 Douarnenez

**Remarque** : Le prêt à l'élève Simon Horellou n'a nécessité aucune intervention.

# Liste des instruments enseignés et leur classe

|  |  |
| --- | --- |
| **Classes d'instruments** | **Instruments** |
| Claviers | Orgue |
| Piano |
| Instruments  amplifiés | Clavier amplifié |
| Guitare électrique |
| Basse électrique |
| Bois | Saxophone |
| Clarinette |
| Flute traversière |
| Cuivres | Trombone |
| Trompette |
| Tuba |
| Cordes | Violon |
| Violoncelle |
| Harpe celtique |
| Percussions | Batterie |

# Modélisation conceptuelle de la base de données modifiée Musicol

|  |
| --- |
| **Diagramme de classes** |
|  |