

Travail d'intégration (420-KED, 420-KA5)

Des circuits et des monuments :

Objectifs

Ce travail vise principalement à vous faire expérimenter les aspects suivants:

- Réaliser un travail d'intégration
- Définir un prototype visuel.
- Définir le modèle de données adéquat.
- Développer l'application C#, ADO.NET
- Travailler en équipe. **Vous devez être en équipe de deux**

Mise en contexte :

L'agence Tourism est une agence qui œuvre dans le domaine du tourisme. Une de ces tâches consiste à faire visiter à ces clients de monuments historiques en leur proposant des circuits

Un circuit à une ville début, une ville d'arrivée, un prix, un nom unique, la durée totale pour le visiter, le nombre maximum de clients pour le circuit. Un circuit est identifié par un numéro séquentiel.

Sur un circuit, nous avons des monuments à visiter. Chaque monument a un nom, une date de construction, une brève histoire du monument, une d'image (qui n'est pas dans la BD), un nombre d'étoiles¹. Les étoiles vont de 1 à 5 (5 étoiles veut dire à ne pas rater, 4 il faut absolument le visiter etc..) et un prix pour une visite guidée.

Un monument peut se trouver sur plusieurs circuits. Et évidemment un circuit a plusieurs monuments.

Avant d'aller visiter un circuit, les clients doivent faire une réservation. Une réservation a un numéro unique qui est séquentiel, une date de réservation et une date limite à laquelle on peut annuler sans pénalité. Un client peut faire plus qu'une réservation. Une réservation concerne un client.

Un client peut réserver pour plusieurs circuits. Sur un circuit on peut avoir plusieurs clients.

¹ Vous devez utiliser un contrôle personnalisé pour afficher et saisir les cotations. (420-KA5)

Le travail demandé :

Après avoir implémenté le modèle de la base de données, vous devez réaliser une application C# ADO.NET qui permet de faire les opérations suivantes:

1. Circuit
 - a. Créer
 - b. Supprimer
 - c. Modifier
 - i. le prix d'un circuit,
 - ii. la ville de départ,
 - iii. la ville d'arrivée (le nom et le numéro ne peuvent être changés).
2. Liste des circuits offerts par l'agence. La recherche doit se faire selon les critères suivants :
 - a. Tous les circuits, **cette liste doit être ordonnée par la cotation (nombre d'étoiles) puis par coût**
 - b. La ville de départ
 - c. Le prix : liste des circuits dont le coût est inférieur à un certain prix (que l'on doit saisir)
 - d. Le nom ou le début du nom
 - e. Contenant un monument donné.
3. Recherche de monuments selon la cotation (nombre d'étoiles). Vous devez afficher les circuits sur lesquels ils se trouvent et sa position dans le circuit (l'ordre dans lequel il sera visité)
4. Rechercher Liste des monuments sur un circuit. La liste des monuments doit se faire un enregistrement à la fois. On doit se déplacer d'un enregistrement à un autre (à l'aide des boutons suivant et précédent²). Il est obligatoire d'avoir une photo du monument. Cet affichage **ne doit pas** se faire dans un DGV. On doit afficher l'ordre dans lequel les monuments seront visités.
5. Quelle est la liste des clients ayant fait (réservé) un circuit donné
6. Quel est le meilleur circuit (Coûts, nombre de monuments de trois étoiles ou plus) à choisir pour visiter un monument en particulier.
7. Ajouter un monument, ajouter un monument pour un circuit.

² Utilisez des boutons flash pour avancer et reculer. (420-KA5)

8. Réserveation pour un client.
 - a. Création/suppression de circuit
 - b. Création/suppression de réserveation de visite guidée de monumentsDans tous les cas (question 8) vous devez afficher le prix total du circuit.

Contrainte concernant la base de données

- la Base de données est Oracle.
- Les numéros de circuits, de clients et de réserveation doit être fait par des séquences.
- Toutes les insertions doivent utiliser l'objet OracleParameter.
- Vous devez insérer au minimum
 - o 15 monuments
 - o 10 circuits, sur chaque circuit, je dois trouver au moins cinq monuments distincts.
 - o 10 clients
 - o 5 réserveations pour chaque client.
 - o En d'autres mots vous devez peupler la base de données de sorte que les recherches donnent des résultats.

Contrainte concernant l'interface graphique :

Remise et correction

Vous devez remettre :

1. Le modèle relationnel de la base de données : Le 30 novembre pour le groupe1 et le 01 décembre pour le groupe 2. La remise doit se faire en mains propres. Tout retard dans la remise entraine la note zéro.
2. Le prototype visuel de l'application C# : 6 décembre.
3. Le script contenant :
 - a. la création des tables
 - b. la définition des données (ALTER)
 - c. les insertions

d. la création des séquences

4. L'application C# ADO.Net.

Pour les points de 3 et 4, une boîte de remise sera ouverte à cet effet. La date limite pour la remise est le 19 décembre. Aucune remise par courriel n'est acceptée. Tout retard implique la note zéro.

La correction de l'application aura lieu le lundi 19 décembre de 8h à 12h. Un ordre de passage vous sera envoyé par Colnet.

Grille de correction : La partie Bases de données compte pour 25% de la note finale

Critères d'évaluation	Pondération	Date de remise
Normalisation et modèles	15%	Le 29 novembre, le 01 décembre
Démonstration du fonctionnement	60%	Le 19 décembre
Efficacité de programmation et validation	25%	Le 19 décembre

Grille de correction : La partie interface 30% de la note finale du cours 420-KA5

Critères d'évaluation	Pondération	Date de remise
Respect du document de conception (Spécifications d'interface)	15%	Le 19 décembre
Utilisation de contrôle personnalisé	15%	Le 19 décembre
Esthétisme, navigation et mises en page adaptées dynamiquement à la taille des fenêtres	35%	Le 19 décembre
Ergonomie et clarté	35%	Le 19 décembre

Important :

À cette évaluation, un mécanisme qui va tenir compte du travail des membres d'une équipe sera ajouté. Vous en serez informés.

D'autres indications et complément suivront.