

Support pour le MOODLE 1 _ R2.01 Approfondissement C#

Consignes : au sein du moodle, vos réponses ne doivent pas seulement permettre de compiler le code.

1. Vous devez :
 - respecter les principes de base : encapsulation, ...
 - bien lire le diagramme et réutiliser tout traitement déjà présent
 - différencier explicitement :
 - l'appel aux champs, propriétés, méthodes d'instance
 - l'appel aux champs, propriétés, méthodes statiques
 - l'appel aux variables locales et paramètres
2. Utilisez le copier/coller pour éviter toute faute de frappe, quand c'est possible. Sinon recopiez à la majuscule près. Aucune faute ne sera acceptée.
3. Ne mettez pas d'espace au sein de vos saisies.

Remarque : Il y a des points négatifs uniquement sur les questions de type QCM

Contexte : Une petite entreprise réunit 2 à 3 peintres en bâtiment. Elle a un planning de chantiers. Pour simplifier, on suppose qu'un chantier est pris en charge du début à la fin par un et un seul peintre. Un chantier se situe à une adresse et peut comporter une ou plusieurs pièces. (Ex : une maison complète ou bien seulement une cuisine)

Pour calculer le prix d'un chantier, on doit :

- Calculer le montant lié à l'achat de la peinture nécessaire pour chaque pièce (ainsi que le total pour le chantier complet) :
 - à partir du prix au m2 de la peinture qui dépend du type de peinture : Acrylique, Glycero ou AvecEffet
 - à partir de la surface totale de la pièce à peindre (4 murs + plafond) à multiplier par le nombre de couches (toujours 2 !)
- Calculer au prorata le nombre d'heures de travail nécessaire pour chaque pièce (ainsi que le total pour le chantier complet) en suivant cette règle simple : en fonction de l'état de la pièce, pour une pièce type de 15m2 (au sol) : on estime nécessaire :
 - 7h (1 jour) de travail pour un support (murs et plafonds) en bon état
 - 14h (2 jours) de travail pour un support en état moyen
 - 21h (3 jours) de travail pour un support en état très dégradé
- Calculer le montant lié à la main d'œuvre : à partir du nombre d'heures totales et du cout horaire du peintre. Le cout horaire d'un peintre dépend de son expérience : débutant, confirmé, expert.
- Calculer les frais de déplacement du peintre : à partir du nombre d'heures totales et à raison de 7h par jour, on peut calculer le nombre de jours nécessaires (CalculeNbJours) et donc le nombre de déplacements (à raison de 2 par jours : aller/retour)

Adresse
Classe

▲ Champs

- codePostal : string
- numéro : string
- ville : string
- voie : string

▲ Propriétés

- CodePostal { get; set; } : string
- Numéro { get; set; } : string
- Ville { get; set; } : string
- Voie { get; set; } : string

▲ Méthodes

- Adresse(string numéro, string voie, string codePostal, string ville)
- CalculeNbKM(Adresse adresse1, Adresse adresse2) : int

Planning
Classe

▲ Champs

- desChantiers : List<Chantier>

▲ Propriétés

- DesChantiers { get; set; } : List<Chantier>

▲ Méthodes

- LesChantiersAVenir() : List<Chantier>
- LesChantiersDuPeintre(Peintre unPeintre) : ...
- NbChantierPourUneVille(string ville) : int
- NbChantiersEnCours() : int
- TriParDate() : void
- TriParPeintre() : void

Peintre
Classe

▲ Champs

- COUTS_HORAIRE : double[]
- experience : ExperiencePeintre
- nom : string
- uneAdresse : Adresse

▲ Propriétés

- Experience { get; set; } : ExperiencePeintre
- Nom { get; set; } : string
- UneAdresse { get; set; } : Adresse

▲ Méthodes

- CoutHoraire() : double
- Equals(object obj) : bool
- GetHashCode() : int
- Peintre(string nom, ExperiencePeintre experience, Adresse uneAdresse)

ExperiencePeintre
Enum

- Debutant
- Confirmé
- Expert

```
public static readonly double[] COUTS_HORAIRE = new double[] { 25, 30, 35 };
```

```
public static readonly double[] NB_HEURES_ETAT = new double[] { 7, 14, 21 };
public static readonly double[] PX_PEINTURE_M2 = new double[] { 5, 8, 10 };
public const int TAILLE_PIECE_TYPE_NB_HEURES = 15;
private const int NB_COUCHES = 2;
```

Peinture
Enum

- Acrylique
- Glycero
- AvecEffet

Etat
Enum

- Bon
- Moyen
- Degrade

Chantier
Classe

▲ Champs

- dateDebutChantier : DateTime
- desPieces : List<Piece>
- NB_DEPLACEMENTS : int
- NB_HEURES_JOUR : int
- nomClient : string
- PX_KM : double
- uneAdresse : Adresse
- unPeintre : Peintre

▲ Propriétés

- DateDebutChantier { get; set; } : DateTime
- DesPieces { get; set; } : List<Piece>
- NomClient { get; set; } : string
- UneAdresse { get; set; } : Adresse
- UnPeintre { get; set; } : Peintre

▲ Méthodes

- CalculeFraisDeplacement() : double
- CalculeMainOeuvre() : double
- CalculeNbHeures() : double
- CalculeNbJours() : double
- CalculePrixPeinture() : double
- CalculeTotal() : double
- DonneDateFinChantier() : DateTime

Piece
Classe

▲ Champs

- choixPeinture : Peinture
- etat : EtatPiece
- hauteur : double
- largeur : double
- longueur : double
- NB_COUCHES : int
- NB_HEURES_ETAT : double[]
- nom : string
- PX_PEINTURE_M2 : double[]
- TAILLE_PIECE_TYPE_NB_HEURES : int

▲ Propriétés

- ChoixPeinture { get; set; } : Peinture
- Etat { get; set; } : EtatPiece
- Hauteur { get; set; } : double
- Largeur { get; set; } : double
- Longueur { get; set; } : double
- Nom { get; set; } : string

▲ Méthodes

- CalculeM2AuSol() : double
- CalculeNbHeures() : double
- CalculePrixPeinture() : double
- CalculeSurfaceTotale() : double
- Equals(object obj) : bool
- GetHashCode() : int
- Piece(double largeur, double longueur, double hauteur, string nom, EtatPiece etat, Peinture choixPeinture)
- ToString() : string

```
public const int NB_HEURES_JOUR = 7;
public const int NB_DEPLACEMENTS = 2;
public const double PX_KM = 0.5;
```