Examen : 01

1. Informations et Consignes de l'Examen (Réf: EX0129102016)

Objectif principal : Évaluer les compétences en Programmation Orientée Objet (POO), la manipulation des collections (List) et l'interaction avec les contrôles Windows Forms avancés (ListBox, ComboBox).

Règles de l'épreuve

- **Durée :** 2 heures.
- **Documents :** Seules les notes de cours manuscrites (brouillons) validées par le formateur sont autorisées.
- **Environnement :** Visual Studio. L'utilisation d'internet, de clés USB ou de téléphones portables entraînera l'annulation immédiate de l'épreuve.
- **Évaluation :** La clarté du code, l'indentation, la pertinence des noms de variables et la gestion des erreurs (blocs try/catch) constituent 30% de la note finale.

2. Contexte du Projet : Système de Gestion des Employés

Une entreprise souhaite informatiser la gestion de son personnel. Vous êtes chargé de développer une application Windows Forms permettant d'ajouter des employés, de calculer leurs salaires en fonction de leur statut, et d'afficher la liste complète du personnel.

Partie 1 : Conception des Classes (POO)

Dans un dossier nommé `Classes` de votre projet, créez l'architecture suivante :

1. **La classe de base `Employe` (Abstraite) :**
   - **Propriétés :** `Matricule` (int), `Nom` (string), `Prenom` (string), `SalaireDeBase` (double).
   - **Constructeur :** Un constructeur initialisant toutes ces propriétés.
   - **Méthode :** Une méthode abstraite `CalculerSalaire()` qui retourne un `double`.
   - **Méthode :** Redéfinissez la méthode `ToString()` pour qu'elle retourne : "*Matricule - Nom Prenom*".

1. **La classe dérivée `Manager`:**
   - Hérite de la classe `Employe`.
   - **Propriété supplémentaire :** `PrimeManagement` (double).

apcpedagogie.com

- **Méthode :** Implémentez `CalculerSalaire()` pour qu'elle retourne le `SalaireDeBase + PrimeManagement`.

1. **La classe dérivée `Technicien`:**
   - Hérite de la classe `Employe`.
   - **Propriété supplémentaire :** `HeuresSupplementaires` (int) et `TauxHoraire` (double).
   - **Méthode :** Implémentez `CalculerSalaire()` pour qu'elle retourne le `SalaireDeBase + (HeuresSupplementaires * TauxHoraire)`.

3. Travail à réaliser : Interface Graphique

Créez une interface Windows Forms principale (`FrmGestionPersonnel`) contenant les éléments suivants :

Zone de l'interface	Contrôles à utiliser	Description de l'action attendue
Saisie des Informations	4 `TextBox`, 1 `ComboBox`	Permet à l'utilisateur de saisir le Matricule, Nom, Prénom et Salaire de base. La ComboBox permet de choisir le type d'employé ("Manager" ou "Technicien").
Bouton "Ajouter"	1 `Button` (BtnAjouter)	Au clic, instancie l'objet correspondant (Manager ou Technicien) et l'ajoute à une liste globale `List<Employe>`, puis met à jour l'affichage.
Affichage du Personnel	1 `ListBox` (LstEmployes)	Affiche la liste de tous les employés enregistrés (grâce à la méthode ToString() redéfinie).
Calcul du Salaire	1 `Button` (BtnCalculer)	Au clic, récupère l'employé sélectionné dans la ListBox, appelle sa méthode `CalculerSalaire()` et affiche le résultat dans une `MessageBox`.

4. Questions Bonus (Pour obtenir la note maximale)

- **Sécurité :** Que se passe-t-il si l'utilisateur clique sur "Calculer le salaire" sans avoir sélectionné un employé dans la ListBox ? Ajoutez une condition pour empêcher le plantage de l'application.
- **Validation :** Assurez-vous que le champ "Matricule" n'accepte que des nombres entiers lors de la saisie. Si l'utilisateur tape des lettres, affichez un message d'erreur et videz la TextBox.

Rédigé et structuré par : **Med Khalil Kribi**

apcpedagogie.com

Source : https://apcpedagogie.com/c-examen-ex0117102016 · téléchargé le 08/11/2026 · © APC Pédagogie · apcpedagogie.com

apcpedagogie.com