

C34 Développer à l'aide d'un langage de programmation à objets

Le candidat est invité à rechercher une illustration, tirée de son activité, de chaque aspect technique de la compétence correspondant présenté dans cette fiche. Cette illustration pourra être mentionnée dans le dossier, et sera présentée lors de l'épreuve.

La présente fiche se présente sous la forme d'une énumération, elle n'est ni un format ni un plan pour l'épreuve ou le dossier d'activité.

Les termes en gras appartiennent au vocabulaire technique de référence de la compétence, ils doivent donc être naturellement employés par le candidat lors de l'épreuve

1. Conception et utilisation de classe :

Diagramme de Classe : Présente l'ensemble des classes définies dans l'activité, leurs attributs et méthodes, et les liens entre les classes.

Classe : Possède et définit des attributs, méthodes, accesseurs et un constructeur.

Classes techniques : Permettent des traitements grâce à des fonctions ou procédures déjà existantes. Utilisées pour le développement

Classes métiers : Spécifiques à un domaine de gestion.

Contrat de la classe : Liste des méthodes, fonctionnalités, interface... C'est l'ensemble des services fourni par l'objet.

Interface de la classe : Tout ce qui est visible depuis les autres classes : constructeur et méthodes de la classe, y compris les accesseurs (« get-eurs » et « set-eurs ») permettant la lecture et l'écriture des attributs.

Objet : Instance de classe.

Instancier une classe : Créer un nouvel objet. Permet d'accéder aux éléments publics de la classe et d'utiliser son contrat.

Attributs : Variables privées.

Encapsulation : Afin de garantir l'intégrité des données, l'implémentation de la classe est cachée ; les attributs sont visibles et utilisables uniquement dans la classe à laquelle ils appartiennent.

Accesseurs : Méthodes qui permettent d'accéder aux attributs privés de la classe depuis une autre classe. Renvoie la valeur. Définit les possibilités de lecture et écriture. Get pour la lecture (fonction), set pour l'écriture (procédure).

Constructeur : Méthode qui permet d'initialiser les attributs d'un objet lors de sa création pour le rendre utilisable. Des paramètres peuvent être passés. Pas obligatoire. Il peut en exister plusieurs pour une même classe.

Destructeur : Détruit un objet, avec toutes les ressources système qui lui ont été allouées lors de son instanciation. Certains environnements de développement prennent automatiquement en charge cette destruction (« ramasse-miettes » ou « garbage collector » en anglais).

Méthodes : Fonctions ou procédures publiques.

Héritage entre classes définies dans l'activité : Les classes filles héritent des attributs et méthodes de la classe mère.

2. Enrichissements possibles :

Surcharge : (polymorphisme paramétrique)

Redéfinition d'une méthode avec des paramètres (signature) différents.

Classes abstraites.

Polymorphisme d'héritage : La possibilité de redéfinir une méthode dans des classes héritant d'une classe de base s'appelle la spécialisation. Il est alors possible d'appeler la méthode d'un objet sans avoir à se soucier de son type. Ceci permet de faire abstraction des détails des classes spécialisées d'une famille d'objet, en les masquant par une interface commune (classe de base).