# **Exercice :** Développement de logiciels

## Modélisation UML des objets :

Une image contenant texte, capture d’écran, Police, conception

Description générée automatiquement

## Mise en contexte :

Des employés travaillent au développement de logiciels.

Les employés ont des compétences diverses et travaillent sur différents aspects du développement selon leur spécialité. Dans le projet, il y a essentiellement des Programmeur, Designer et Tech\_Reseau.

Les employés sont regroupés par équipes de développement, production et support.

## Création des différentes classes dans Python. À cette étape, dans chaque classe, faites seulement la signature des méthodes et utilisez pass pour terminer chaque méthode tout de suite.

Une image contenant texte, capture d’écran, Police, conception

Description générée automatiquement

Vous pouvez voir qu’il y a de l’héritage. Un Programmeur est un Employe. Et c’est la même chose pour un Designer qui est un Employe et un Tech\_Reseau qui est un Employe.

Quand vous allez créer vos classes en Python. Commencez par créer la classe Employe et les autres classes qui héritent de la classe Employe

Les instances de Programmeur possèdent un attribut privé « \_langages » ainsi qu’une propriété « langages\_connu » qui permet d’obtenir la valeur de « \_langages ». Vous devez aussi faire un setter qui permet D’AJOUTER des langages connus, mais pas de remplacer ceux existants.

Quand un Designer fait l’action de dessiner(), il produit un artefact qui est ajouté à la liste d’artefacts. ( pour l’instant il s’agit simplement d’un str, on pourrait décider d’implémenter une classe artefact éventuellement )

Quand un Tech\_Reseau fait l’action d’intervenir(), il ajoute une intervention à sa liste d’intervention. ( aussi simplement un str pour l’instant )

Une fois que la classe Employe et ses différents types d’employés ont été créés, vous pouvez passer à l’étape de créer la classe Equipe. Et finalement vous pourrez créer la classe Departement.

Une image contenant texte, capture d’écran, Police, conception

Description générée automatiquement

L’attribut « employes » est une liste de dictionnaires. Chaque dictionnaire possède 3 clefs : Employe, date\_debut, et date\_fin. Quand un employé joint une équipe, il est ajouté à la liste des employés de l’équipe avec la date\_debut à maintenant et la date\_fin à None.

(On obtient la date courante avec datetime.date.today() après avoir importé le module datetime).

Quand un employé cesse de travailler pour une équipe, il n’est pas enlevé de la liste des employés de l’équipe, mais on change sa date\_fin pour être le moment où il quitte.

Quand une équipe est ajoutée à la liste des équipes pour un département, on n’a pas besoin de la date d’ajout.

## Faites ici le pseudo-code pour ajouter un employé à une équipe

## Faites un diagramme de flux pour ajouter un employé à une équipe et copiez-le ici.

## Faites ici le pseudo-code pour enlever un employé d’une équipe

## Faites un diagramme de flux pour enlever un employé d’une équipe et copiez-le ici.

## Voici le pseudo-code pour imprimer les infos d’une équipe (le nom de l’équipe et son nombre d’employés actifs)

(On sait qu’un employé actif dans une équipe est un employé dont sa date\_fin est toujours None.)

Créer un compteur pour compter le nombre d’employés encore dans l’équipe et lui donner la valeur de 0 au départ

Faire une boucle pour passer à travers chaque employé qui est dans la liste des employés de l’équipe

Vérifier si la date de fin de cet employé est toujours None (il n’a pas quitté l’équipe)

Alors augmenter notre compteur de 1

Créer une chaine de caractères pour le nom de l’équipe et le nombre d’employés

Retourner cette chaine de caractères

## Voici le diagramme de flux pour imprimer les infos d’une équipe

|  |  |
| --- | --- |
|  | Détail : |

## Faites ici le pseudo-code pour imprimer les infos des employés d’une équipe

## Faites un diagramme de flux pour imprimer les infos des employés d’une équipe et copiez-le ici.

.

## Complétez maintenant le code python des différentes classes.

Regardez le résultat attendu plus bas pour savoir ce qu’il faut coder dans les différentes méthodes, surtout au niveau des méthodes < \_\_str\_\_ > de chaque classe.

## Instanciez les objets des différentes classes et faites les tests demandés ci-dessous.

Voyez le résultat attendu ci-dessous pour voir quelles informations donner pour la création des objets des différentes classes.

Les étapes sont les suivantes :

|  |  |
| --- | --- |
| Action | Résultat attendu |
| -Créer un nouveau Departement  -Imprimer les infos sur ce département | Infos sur le nouveau département, avec 0 équipe  Nom: ZenithSuit,  Nb Equipes: 0 |
| -Créer une nouvelle équipe  -Imprimer les infos sur cette équipe | Infos sur la nouvelle équipe  Nom: Developpement,  Nb Employes: 0 |
| -Ajoutez l’équipe au Departemet¸  -Imprimez à nouveau les infos sur le département | On devrait voir que le nouveau département a maintenant 1 équipe  Nom: ZenithSuit,  Nb Equipes: 1 |
| -Créez un Programmeur avec le rôle de gestionnaire  -Ajoutez-lui le langage Python  -Imprimez les infos sur ce programmeur | Infos sur le gestionnaire:  ID: 1001,  Nom: Dupré,  Prénom: Marc,  Role: gestionnaire  Liste des langages : ["Python"] |
| -Créez un Designer avec le rôle top-designer  -Ajoutez-lui un artefact qui est le logo  -Imprimez les infos de ce designer | Infos sur le designer  ID: 1002,  Nom: Fortier,  Prénom: Amy,  Rôle: top designer  Liste d’artefacts complétés: ['logo'] |
| -Créez un Tech\_reseau avec le rôle « architecte principal »  -Ajoutez-lui 2 interventions  -Imprimez les infos de ce tech\_reseau | Infos sur l'architecte principal  ID: 1003,  Nom: Michaud,  Prénom: Luc,  Rôle: architecte principal  Liste des interventions complétées: ['Mise en place de 3 serveurs', "Installation du vlan pour l'équipe de développement"] |
| -Imprimez les infos sur l’équipe  -Imprimez la liste des employés de cette équipe | Infos sur la nouvelle équipeNom: Developpement,Nb Employes: 3Employe: Dupré, Marc - Date début: 2023-04-09 - Date fin: NoneEmploye: Fortier, Amy - Date début: 2023-04-09 - Date fin: NoneEmploye: Michaud, Luc - Date début: 2023-04-09 - Date fin: None |
| -Enlevez la top\_designer de l’équipe  -Imprimez les infos sur l’équipe  -Imprimez la liste des employés de cette équipe | Infos sur la nouvelle équipe, après le départ de la top designer de l'équipe de développementNom: Developpement,Nb Employes: 2Employe: Dupré, Marc - Date début: 2023-04-09 - Date fin: NoneEmploye: Fortier, Amy - Date début: 2023-04-09 - Date fin: 2023-04-09Employe: Michaud, Luc - Date début: 2023-04-09 - Date fin: None |

## Remise :

Dans GitHub, faites la remise de votre code python, de ce document qui contiendra votre pseudo-code et finalement la remise de votre diagramme de flux.

Cette remise est à faire AVANT la fin de ce laboratoire.