

- Examen sur papier
- Durée : 2h
- Une feuille A4 manuscrite recto/verso autorisée
- Sans calculatrice

À lire avant de commencer

Le sujet est composé de 10 pages, numérotées de 1 à 10

- Les exercices peuvent être traités dans l'ordre de votre choix.
- Dans chaque exercice, les questions ne sont pas forcément de difficulté croissante.
- Les différentes questions attendent des **réponses courtes**.
- Respectez scrupuleusement les consignes. Cependant, si une question ne vous semble pas claire, notez sur votre copie la manière dont vous l'avez comprise et traitée.
- Quand du code est demandé, veuillez respecter les règles de bases du langage (mots clefs et indentation). Des erreurs seront tolérées, car vous êtes sur papier. En cas d'oubli d'un nom de méthode Python, utilisez du pseudo-code.
- Sauf mention contraire, il n'est pas demandé d'écrire la documentation du code (même si la doc c'est bien!)

Bonne chance!

1 POO, Bases de données et Sécurité (7 points)**a) Quels sont les 3 grands principes de la Programmation Orientée Objet?**

- Encapsulation
- Héritage
- Polymorphisme

b) Quels sont les 4 piliers de la sécurité informatique? Détaillez celui de votre choix (définition, mécanismes associés)

Ce sont les principes CAID

- Confidentialité : Seules les personnes autorisées doivent avoir accès aux informations qui leur sont destinées. Mécanismes associés : gestion des droits, cryptographie
- Authentification
- Intégrité
- Disponibilité

c) Quels sont les intérêts de séparer votre code en couches?

- Lisibilité du code : répartir le code dans de multiples fichiers évite d'avoir des classes de 5000 lignes
- Travail en groupe : permet de se répartir plus facilement les tâches
- Facilite le débogage
- Permet également de simplifier la migration vers un nouveau système (ex : changement de base de données)

d) Quel est le rôle principal de la couche DAO?

Elle permet de faire le lien entre la couche métier et la base de données. Elle expose les méthodes du CRUD pour communiquer avec la BDD.

e) Expliquez comment se prémunir des injections SQL? Donnez quelques bonnes pratiques.

Une injection SQL Consiste à saisir du SQL dans une zone de texte pour exécuter une autre requête que celle prévue. Les bonnes pratiques pour se protéger sont :

- Échapper les caractères spéciaux
- Utiliser une requête préparée

- f) **On vous demande de gérer le stockage de mots de passe en base de données. Comment procédez-vous ?**

Il ne faut surtout pas stocker les mots de passe en clair. La bonne pratique consiste à concaténer le mot de passe avec un autre élément (appelé le sel) relatif à l'utilisateur, par exemple son nom. Puis de hacher l'ensemble. Ce hash est alors stocké en base de données. Lorsque l'utilisateur se connecte et saisit son mot de passe, le même procédé est appliqué et l'algorithme compare les 2 hash.

Le fait d'ajouter le sel complique la tâche des hackers qui doivent tester énormément plus de combinaisons pour craquer un mot de passe.

- g) **Complétez la classe `PersonneDao` (ci-après) en écrivant la méthode `trouver_toutes` qui retourne la liste de toutes les personnes de la base de données (pour cette méthode, le résultat renvoyé par le curseur pourra être traité comme une liste de dictionnaires).**

```
1 class Personne:
2     def __init__(self, nom, prenom, age, id=None):
3         self.nom = nom
4         self.prenom = prenom
5         self.age = age
6         self.id = id
7
8 class PersonneDao(metaclass=Singleton):
9     def trouver_par_id(self, id: str) -> Personne:
10         """Trouver une Personne à partir de son id"""
11         with DBConnection().connection as connection:
12             with connection.cursor() as cursor:
13                 cursor.execute(
14                     "SELECT nom, "
15                     "      prenom, "
16                     "      age "
17                     " FROM exo.personne "
18                     " WHERE id=%(id)s ",
19                     {"id": id},
20                 )
21                 res = cursor.fetchone()
```

```
22
23     personne = None
24     if res:
25         personne = Personne(
26             nom=res["nom"],
27             prenom=res["prenom"],
28             age=res["age"],
29             id=id,
30         )
31     return personne
32
33 def trouver_toutes(self) -> List[Personne]:
34     """Retourne toutes les Personnes de la BDD"""
35
36     # Requête pour lister toutes les personnes de la base de données
37     request = (
38         "SELECT nom,                "
39         "        prenom,            "
40         "        age,              "
41         "        id                "
42         " FROM exo.personne        "
43     )
44
45     with DBConnection().connection as connection:
46         with connection.cursor() as cursor:
47             cursor.execute(request)
48             # on récupère toutes les lignes retournées par le curseur
49             res = cursor.fetchall()
50
51             # initialisation d'une liste vide pour stocker les Personnes
52             liste_personnes = []
53
54             # si le curseur a retourné quelque chose (variable res non vide)
```

```
55     if res:
56         # pour chaque élément retourné
57         for raw_personne in res:
58             # instanciación d'une Personne en utilisant les champs disponibles
59             personne = Personne(
60                 nom=raw_personne["nom"],
61                 prenom=raw_personne["prenom"],
62                 age=raw_personne["age"],
63                 id=raw_personne["id"],
64             )
65             # ajout de la Personne créée à la liste
66             liste_personnes.append(personne)
67
68     return liste_personnes
```

2 Web services (3 points)

a) **Qu'est-ce qu'un Web service ?**

Un web service est un module applicatif, accessible sur le protocole HTTP via une URL qui va répondre à une demande

b) **Quelles sont les 4 méthodes http pour contacter un Web service ?**

- POST : créer
- GET : lire
- PUT : mettre à jour
- DELETE : supprimer

c) **Nommez les éléments de la requête suivante : GET https://pokeapi.fr:5000/pokemon?limit=10**

- GET : méthode
- https : protocole
- pokeapi.fr : nom de domaine
- 5000 : port
- /pokemon : endpoint
- ?limit=10 : paramètre de la requête

d) **Qu'est-ce qu'un Swagger ?**

Le swagger recense tous les endpoints du webservice, et comment les utiliser. C'est une documentation de l'API.

3 Git et Versionnage (5 points)

Dans cet exercice, vous travaillez dans une équipe composée de 3 personnes sur le projet Argousier. Le code de votre projet est stocké sur GitHub. Comme tous les matins, vous arrivez à 7h30 à votre bureau. Vos collègues, eux sont plutôt du soir. Vous allumez votre machine et lancez Git Bash.

- a) **Vous voulez vérifier si vos collègues ont avancé sur le code hier soir. Quelle commande exécutez-vous ?**

git pull

Pas de chance, hier avant de partir, vous avez oublié de partager votre code sur le dépôt distant. Quelqu'un a modifié le même fichier que vous et vous obtenez un conflit. Quand vous ouvrez le fichier en question, vous trouvez ceci :

```
1 <<<<<< HEAD (current change)
2 print("Bonjour Alain")
3 =====
4 print("Bonjour Philippe")
5 >>>>>> (incoming change)
```

- b) **Vous imposez votre version. Quelles actions faites-vous pour mettre à jour le dépôt distant ?**

- J'accepte le changement courant, c'est à dire que je conserve `print("Bonjour Alain")` et j'efface le reste
- `git add .`
- `git commit -m "resolution du conflit"`
- `git push`

Vous avez maintenant réussi à mettre à jour le dépôt distant. Votre collègue Annabelle arrive à ghoo, développe pendant 30min, puis essaie de partager son code avec les commandes suivantes :

```
1 >>> git add .
2 >>> git commit -m "correction incident 5"
3 >>> git push
```

Cependant lors du push, elle rencontre une erreur : « Updates were rejected because the remote contains work that you do not have locally. »

- c) **Qu'est-ce qu'Annabelle a oublié de faire ?**

git pull

- d) **À 10h00, Michel arrive en panique. Il vient d'entendre à la radio que GitHub vient d'être piraté à l'instant et qu'ils ont perdu tous les dépôts. Est-ce que vous allez devoir recommencer votre projet à zéro (expliquez) ?**

Git est décentralisé, ce qui signifie que chaque copie du référentiel contient l'intégralité de l'historique du projet. Donc tous les membres de l'équipe ont une copie locale du dépôt Git (plus ou moins à jour s'ils n'ont pas fait de pull depuis longtemps). Et d'ailleurs la version présente sur GitHub a forcément été poussée par quelqu'un. Cette personne a la version la plus récente du code.

- e) **À 11h00, Johanna, qui travaille sur un projet codé en R, vient vous voir, car elle est intéressée par l'utilisation de Git. Est-ce possible d'utiliser Git avec du code R ? Expliquez-lui en quelques mots à quoi sert Git.**

Il est tout à fait possible d'utiliser Git pour versionner du code R. Git fonctionne très bien avec les fichiers textes (un peu moins optimal pour les fichiers .pdf, .xls, .doc)

Git est un système qui permet de versionner du code en créant des points de sauvegarde. Il permet aussi de travailler à plusieurs sur un même dépôt. Le dépôt central est celui qui fait référence. Chaque utilisateur peut cloner ce dépôt central pour créer un dépôt local. Il peut ensuite modifier le code sur son dépôt local, et enfin partager ses modifications en mettant à jour le dépôt central. Git aide à synchroniser les différents dépôts.

4 Agilité et Tests (5 points)

a) **À quoi ça sert de faire des tests?**

- Pour vérifier que votre programme fonctionne
- Pour détecter des erreurs
- Pour éviter les régressions (quand vous modifiez du code)

b) **Comment appelle-t-on la méthode qui consiste à écrire des tests avant de coder?**

Le Test Driven Development (TDD)

c) **Citez 4 types de tests?**

- Test unitaire
- Test fonctionnel
- Test de charge
- Test d'intégration

d) **Écrivez 2 tests unitaires pour vérifier le bon fonctionnement de la méthode `trouver_par_id(id)` de l'exercice 1**

- Cas nominal : `id = 1` existant dans la base de données
- Cas d'erreur : `id = 999` n'existant pas dans la base de données

```
1 from unittest import TestCase, TextTestRunner, TestLoader
2 from dao.personne_dao import PersonneDao
3
4 class TestPersonneDao(TestCase):
5     def test_trouver_par_id_ok(self):
6         # GIVEN
7         personne_dao = PersonneDao()
8
9         # WHEN
10        personne = personne_dao.trouver_par_id(1)
11
12        # THEN
13        self.assertEqual(1, personne.id)
14
```

```
15     def test_find_attack_by_id_not_found(self):
16         # GIVEN
17         personne_dao = PersonneDao()
18
19         # WHEN
20         personne = personne_dao.trouver_par_id(999)
21
22         # THEN
23         self.assertIsNone(personne)
24
25     if __name__ == "__main__":
26         # Run the tests
27         result = TextTestRunner(verbosity=2).run(
28             TestLoader().loadTestsFromTestCase(TestPersonneDao)
29         )
```

e) Citez 3 frameworks agile? Lequel est le plus utilisé?

- SCRUM : le plus utilisé
- Kanban
- XP

f) VRAI ou FAUX? (sans justification)

1. Le cycle en V permet d'avoir des cycles de livraisons plus rapides. : FAUX
2. « L'adaptation au changement plus que le suivi d'un plan » est l'une des 4 valeurs fondamentales de l'agilité : VRAI
3. Le principal inconvénient de l'agilité est qu'il peut y avoir un effet tunnel. : FAUX
4. Le Scrum Master est le responsable du bon fonctionnement de l'équipe. : VRAI

g) Bonus : Est-ce qu'en tant que responsable d'une application informatique, vous feriez confiance aux utilisateurs de votre application? Oui ou Non (réponse correcte : 0.25 pt, réponse fausse : -25 pts, non réponse : 0 pt)

NON JAMAIS