

# Examen Formatif partie 2– Programmation Orientée Objet en Python

Thème : Cabinet Médical

## Formulaire Avancé avec Validations Strictes

Développez une application graphique en Python permettant de valider un formulaire de **données d'identité numérique sécurisée**. L'interface doit être réalisée **exclusivement** avec Tkinter.

Toutes les validations doivent obligatoirement être faites avec des expressions régulières (re) — aucune instruction conditionnelle (if, len, in, etc.) n'est autorisée.

### Champs à valider

#### 1. Nom d'utilisateur

- Doit **commencer par une lettre majuscule**, suivie de **2 à 14 caractères alphanumériques**.
- Peut contenir un seul underscore ( \_ ) ou point ( . ) mais **jamais deux caractères spéciaux consécutifs**.
- Ne doit **pas se terminer par un caractère spécial**.

#### 2. Clé d'identification

- Chaîne de **24 caractères exactement**, composée uniquement de :
  - Lettres majuscules (A-Z)
  - Chiffres (0-9)
  - Aucun caractère ne doit être répété deux fois de suite
- Exemple valide : A9B7C2D4E6F8G0H1I3J5K7L9

#### 3. Numéro de passeport international

- Format strict : P-AAA-999999
- Où :
  - P- est un préfixe fixe
  - AAA représente trois lettres majuscules
  - 999999 représente six chiffres

#### 4. Mot de passe ultra-sécurisé

- Minimum 10 caractères
- Doit inclure :
  - Une **lettre minuscule accentuée** (ex : é, à, è, etc.)
  - Une **lettre majuscule non accentuée**
  - Au moins **deux symboles** parmi : @, #, \$, %, \*, ?, !
  - Au moins un **chiffre impair** (1, 3, 5, 7 ou 9)

## Fonctionnalités avancées

- Une **case à cocher** pour **afficher ou masquer** le mot de passe.
- Les **messages de validation doivent apparaître dynamiquement**, à chaque saisie, sans action de bouton.
- Sous chaque champ, un message en :
  - **greenvert** si le champ est valide
  - **redrouge** si le champ est invalide

## Interface graphique

- Bouton principal : **Vérifier l'identité**
- Interface avec champ texte et label pour chaque élément
- Message final clair :
  - **Identité numérique conforme** si tous les champs sont valides
  - **Identité invalide** sinon

## Contraintes techniques

- Le fichier Python doit être nommé : `identite-votreprenom.py`
- Interface **Tkinter uniquement**
- **Aucune logique conditionnelle** : pas de `if`, `in`, `len`, etc.
- Toutes les validations doivent être faites avec le module `re` (expressions régulières)
- Le code doit être **structuré, fonctionnel, lisible et bien commenté**

Souhaitez-vous qu'une maquette d'interface ou un squelette de code vous soit fourni ?