

Questionnaire d'analyse du programme demo slider partie1

1 — Structure générale du programme

1. Où est le point d'entrée du programme
 2. À quoi sert le macro-appel `START_JUCE_APPLICATION(JuceSliderDemoApplication)` ?
Comment peut-il être comparé à une fonction `main()` classique en C++ ?
 3. Pourquoi la classe `MainWindow` est-elle définie à *l'intérieur* de la classe `JuceSliderDemoApplication` ? Quels peuvent être les avantages et inconvénients d'une telle approche ?
 4. Où est instancier la classe `MainWindow` ?
 5. Quelle est le rôle de cette classe ?
 6. Que représente la classe `JuceSliderDemoApplication` pour `juce::JUCEApplication`
 7. Que signifie "override"
 8. Décris le rôle de la méthode `initialise()`. À quel moment est-elle appelée et pourquoi instancie-t-on la fenêtre principale ici ?
 9. Pourquoi la méthode `shutdown()` met-elle simplement `mainWindow = nullptr` ? Que signifie "libérer une fenêtre" dans le contexte d'un programme GUI ?
 10. Que se passe-t-il lorsqu'on clique sur la croix (bouton de fermeture) de la fenêtre ? Décris toute la chaîne d'appels jusqu'à la fermeture finale de l'application.
-

2 — La classe `MainWindow`

11. Quelle est la nature et la fonction de la classe `juce::DocumentWindow` dont hérite `MainWindow` ?
 12. À quoi sert l'appel `setContentOwned(new MainComponent(), true)` ?
 - Pourquoi utilise-t-on `new` ici ?
 - Que signifie le paramètre `true` ?
 13. À quoi sert `centreWithSize(getWidth(), getHeight())` dans ce contexte ?
 14. Explique comment l'**override** de `closeButtonPressed()` permet d'intercepter l'événement de fermeture de la fenêtre.
-

3 — Pointeurs et gestion mémoire (C++)

15. Explique ce qu'est un **pointeur intelligent** en C++. Pourquoi utilise-t-on ici un `std::unique_ptr` ?
16. Compare `std::unique_ptr` avec `std::shared_ptr` :
 - différences de propriété,
 - cas d'usage adaptés,
 - risques éventuels.
17. Pourquoi est-il important d'éviter les pointeurs bruts (`new`, `delete`) dans une application moderne C++ et plus encore dans un framework GUI ?