

Résolution Erreur DLL PyQt6 sur Windows

Erreur Rencontrée

```
ImportError: DLL load failed while importing QtCore: La procédure spécifiée est introuvable.
```

Solutions (dans l'ordre)

Solution 1 : Installer Visual C++ Redistributables (RECOMMANDÉ)

PyQt6 nécessite les bibliothèques Visual C++ de Microsoft.

Téléchargement

Lien officiel Microsoft : https://aka.ms/vs/17/release/vc_redist.x64.exe

Ou depuis le site officiel : <https://learn.microsoft.com/fr-fr/cpp/windows/latest-supported-vc-redist>

Installation

1. Télécharger `vc_redist.x64.exe` (pour Windows 64-bit)
2. Exécuter en tant qu'administrateur
3. Suivre les instructions d'installation
4. Redémarrer votre ordinateur
5. Relancer l'application : `python main.py`

Solution 2 : Réinstaller PyQt6 Proprement

Si la solution 1 ne fonctionne pas :

```
bash
```

```
# 1. Désactiver l'environnement virtuel (si actif)  
deactivate
```

```
# 2. Supprimer l'environnement virtuel  
rmdir /s /q venv
```

```
# 3. Recréer l'environnement
```

```
python -m venv venv  
venv\Scripts\activate
```

```
# 4. Mettre à jour pip
```

```
python -m pip install --upgrade pip
```

```
# 5. Installer les dépendances une par une
```

```
pip install PyQt6==6.6.1  
pip install PyQt6-WebEngine==6.6.0  
pip install openai==1.12.0  
pip install httpx==0.26.0
```

```
# 6. Tester
```

```
python main.py
```

Solution 3 : Utiliser PyQt5 (Alternative)

Si PyQt6 continue de poser problème, vous pouvez utiliser PyQt5 qui est plus stable sur certaines configurations Windows.

Modifier requirements.txt

```
txt  
  
# Remplacer PyQt6 par PyQt5  
PyQt5==5.15.10  
PyQt5-WebEngine==5.15.6  
openai==1.12.0  
httpx==0.26.0
```

Modifier les imports (automatique)

Créer un fichier `fix_pyqt5.py` :

```
python
```

```

import os
import re

def replace_pyqt6_to_pyqt5(directory):
    """Remplace PyQt6 par PyQt5 dans tous les fichiers Python."""
    for root, dirs, files in os.walk(directory):
        # Ignorer venv
        if 'venv' in root:
            continue

        for file in files:
            if file.endswith('.py'):
                filepath = os.path.join(root, file)

                with open(filepath, 'r', encoding='utf-8') as f:
                    content = f.read()

                    # Remplacer les imports
                    new_content = content.replace('PyQt6', 'PyQt5')
                    new_content = new_content.replace('QtWebEngineWidgets', 'QtWebEngineWidgets')

                    if new_content != content:
                        with open(filepath, 'w', encoding='utf-8') as f:
                            f.write(new_content)
                        print(f'✓ Modifié: {filepath}')

    if __name__ == '__main__':
        print("Conversion PyQt6 → PyQt5...")
        replace_pyqt6_to_pyqt5('.')
        print("\n✓ Conversion terminée!")
        print("Installez maintenant: pip install -r requirements.txt")

```

Utilisation :

```

bash

python fix_pyqt5.py
pip install -r requirements.txt
python main.py

```

Solution 4 : Vérifier la Version de Python

PyQt6 nécessite Python 3.8 ou supérieur.

```

bash

```

```
# Vérifier votre version  
python --version  
  
# Si < 3.8, télécharger une version récente depuis :  
# https://www.python.org/downloads/
```

Solution 5 : Nettoyer le Cache Pip

Parfois, des fichiers corrompus dans le cache peuvent causer des problèmes.

```
bash  
  
# Nettoyer le cache pip  
pip cache purge  
  
# Réinstaller  
pip uninstall PyQt6 PyQt6-WebEngine -y  
pip install --no-cache-dir PyQt6==6.6.1 PyQt6-WebEngine==6.6.0
```

Solution 6 : Variables d'Environnement

Ajouter les DLLs Qt au PATH (temporaire pour test).

```
bash  
  
# Dans PowerShell  
$env:PATH += ";$PWD\venv\Lib\site-packages\PyQt6\Qt6\bin"  
python main.py
```

Si ça fonctionne, ajouter de façon permanente :

1. Ouvrir "Modifier les variables d'environnement système"
2. Variables d'environnement → Path → Modifier
3. Ajouter : C:\chemin\vers\otre\projet\venv\Lib\site-packages\PyQt6\Qt6\bin

Test de Diagnostic

Créer un fichier `test_pyqt.py` :

```
python
```

```

"""Test de diagnostic PyQt6"""
import sys

print("== DIAGNOSTIC PyQt6 ==\n")

# 1. Version Python
print(f"Python version: {sys.version}")
print(f"Architecture: {sys.maxsize > 2**32 and '64-bit' or '32-bit'}\n")

# 2. Test import PyQt6
try:
    from PyQt6 import QtCore
    print("✓ PyQt6.QtCore importé avec succès")
    print(f" Version Qt: {QtCore.qVersion()}")
    print(f" Version PyQt: {QtCore.PYQT_VERSION_STR}\n")
except ImportError as e:
    print(f"✗ Erreur import PyQt6.QtCore:")
    print(f" {e}\n")
    sys.exit(1)

# 3. Test import QtWidgets
try:
    from PyQt6.QtWidgets import QApplication
    print("✓ PyQt6.QtWidgets importé avec succès\n")
except ImportError as e:
    print(f"✗ Erreur import PyQt6.QtWidgets:")
    print(f" {e}\n")
    sys.exit(1)

# 4. Test création QApplication
try:
    app = QApplication([])
    print("✓ QApplication créée avec succès")
    print("\n💡 PyQt6 fonctionne correctement !")
except Exception as e:
    print(f"✗ Erreur création QApplication:")
    print(f" {e}\n")
    sys.exit(1)

```

Exécuter :

```

bash
python test_pyqt.py

```

🔍 Diagnostic Avancé

Vérifier les DLLs

```
bash

# Dans PowerShell
cd venv\Lib\site-packages\PyQt6\Qt6\bin
dir *.dll
```

Vous devriez voir des DLLs comme :

- `Qt6Core.dll`
- `Qt6Gui.dll`
- `Qt6Widgets.dll`

Dépendances DLL

Télécharger **Dependency Walker** pour analyser les DLLs manquantes : <https://www.dependencywalker.com/>

📋 Récapitulatif des Solutions

Solution	Probabilité	Temps
1. Visual C++ Redistributables	★★★★★	5 min
2. Réinstaller PyQt6	★★★★	10 min
3. Utiliser PyQt5	★★★	15 min
4. Mettre à jour Python	★★	20 min
5. Nettoyer cache pip	★★	5 min
6. Variables PATH	★	5 min

SOS Si Rien ne Fonctionne

Option A : Utiliser PySide6 (Alternative Qt)

PySide6 est l'implémentation officielle de Qt :

```
bash

pip uninstall PyQt6 PyQt6-WebEngine -y
pip install PySide6==6.6.1
```

Puis modifier les imports (similaire à PyQt5).

Option B : Signaler le Problème

Si après toutes ces solutions le problème persiste :

1. Exécuter le test de diagnostic

2. Copier les résultats

3. Ouvrir une issue GitHub avec :

- Version Windows
- Version Python
- Logs du diagnostic
- Sortie de `pip list`

💡 Prévention

Pour éviter ce problème à l'avenir :

1. Toujours installer Visual C++ Redistributables en premier
2. Utiliser un environnement virtuel
3. Mettre à jour pip : `python -m pip install --upgrade pip`
4. Installer les packages un par un en cas de doute

🎯 Solution Rapide (TL;DR)

90% des cas résolus avec :

```
bash

# 1. Installer Visual C++ Redistributables
# Télécharger: https://aka.ms/vs/17/release/vc_redist.x64.exe
# Exécuter et redémarrer

# 2. Réinstaller PyQt6
pip uninstall PyQt6 PyQt6-WebEngine -y
pip install PyQt6==6.6.1 PyQt6-WebEngine==6.6.0

# 3. Tester
python main.py
```

Besoin d'aide ? Exécutez `python test_pyqt.py` et partagez les résultats.