# How to Manually Configure VS Code to Use Anaconda Python

This guide helps team members manually set up VS Code to use Anaconda’s Python interpreter — especially useful when auto-detection fails.

## 🛠 Prerequisites

- VS Code installed

- Anaconda already installed (C:\Users\<YourName>\anaconda3)

- The Microsoft Python extension installed in VS Code

## ✅ Step-by-Step Instructions

### 🔹 Step 1: Open the User Settings Folder

1. Open File Explorer

2. Go to: %APPDATA%\Code\User

This expands to: C:\Users\<YourName>\AppData\Roaming\Code\User

3. If the 'User' folder doesn’t exist, create it.

### 🔹 Step 2: Create or Edit settings.json

1. In the User folder, create a file named: settings.json

2. Open the file in Notepad or VS Code

3. Add this content:

{  
 "python.defaultInterpreterPath": "C:\\Users\\<YourName>\\anaconda3\\python.exe",  
 "python.pythonPath": "C:\\Users\\<YourName>\\anaconda3\\python.exe"  
}

### 🔹 Step 3: Restart VS Code

Fully close and reopen VS Code. Open any .py file. You should now see in the bottom-right corner:

Python 3.x.x 64-bit ('base': conda)

### 🔹 Step 4: Confirm It’s Working

Open the terminal in VS Code (Ctrl + `), then run:

python --version

You should see your Anaconda Python version.

## 💡 Notes:

- These settings work without needing admin access.

- They apply to all Python files unless overridden by workspace settings.

- This method bypasses any broken interpreter auto-detection logic.