Progress®

Guided Exercise 2.1: Working with projects in your workspace

OpenEdge®

Guided Exercise 2.1: Working with projects in your workspace

Overview

In this Guided Exercise, you will first import an OpenEdge project named Server. Then you will create an OpenEdge project called Client and add an ABL procedure to it. Then, you will start a Database Server and configure the Server project to connect to the database. Finally, you will modify PROPATH for the Client project to use resources in the Server project.

Important: You must complete this Guided Exercise to perform subsequent Guided Exercises in this course.

The exercise steps take approximately 30 minutes to complete.

Before you begin

Before you begin this Guided Exercise, you must set up your exercise environment, if you have not done so already. See the Before you begin module for details.

Location of files:

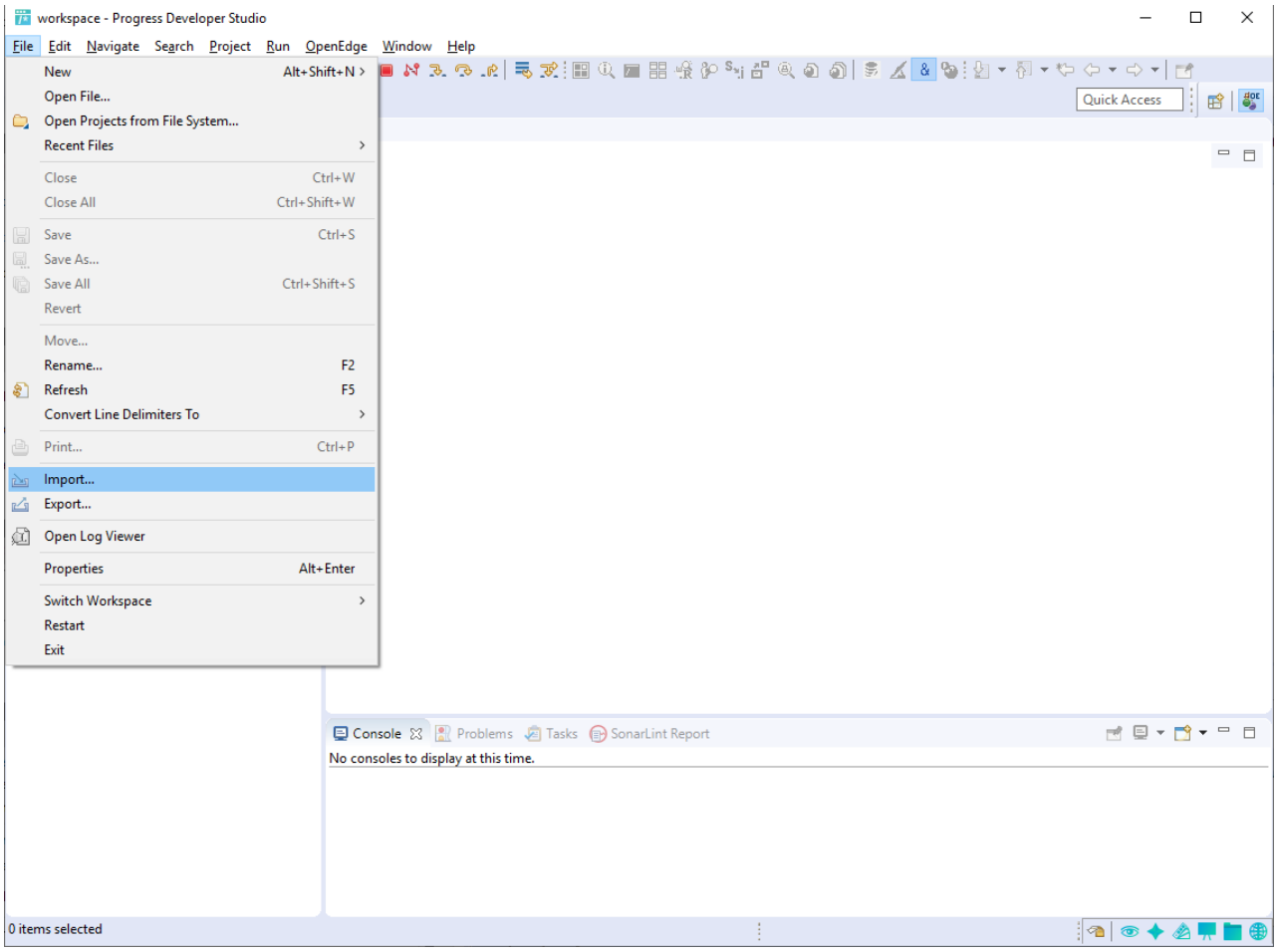
Exercise files: `C:\progress_education\openedge\DeveloperStudio\Exercise`.

Part 1—Importing an OpenEdge project

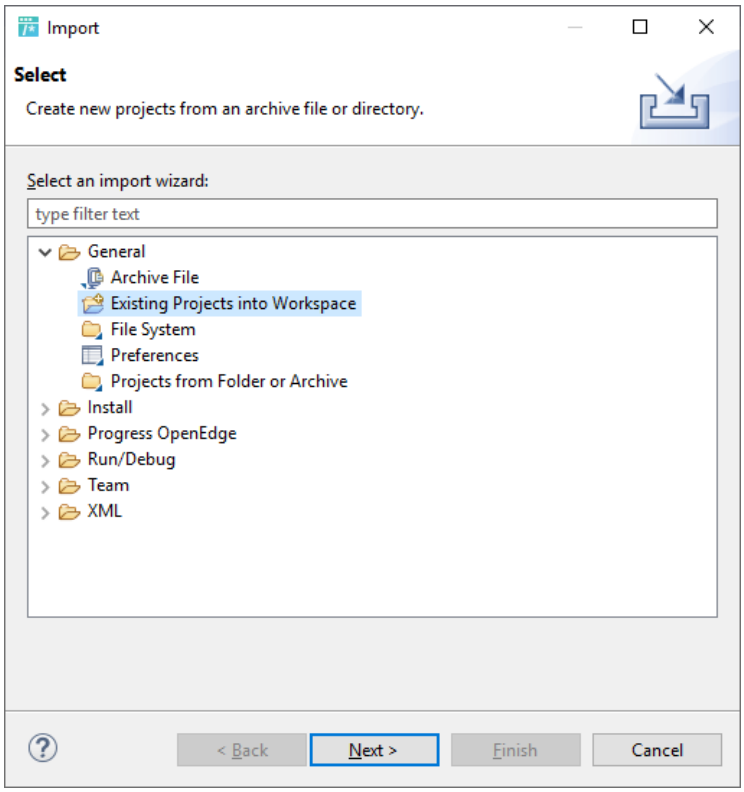
You will now import an existing project called **Server**.

1. Follow these steps to import the **Server** project.:

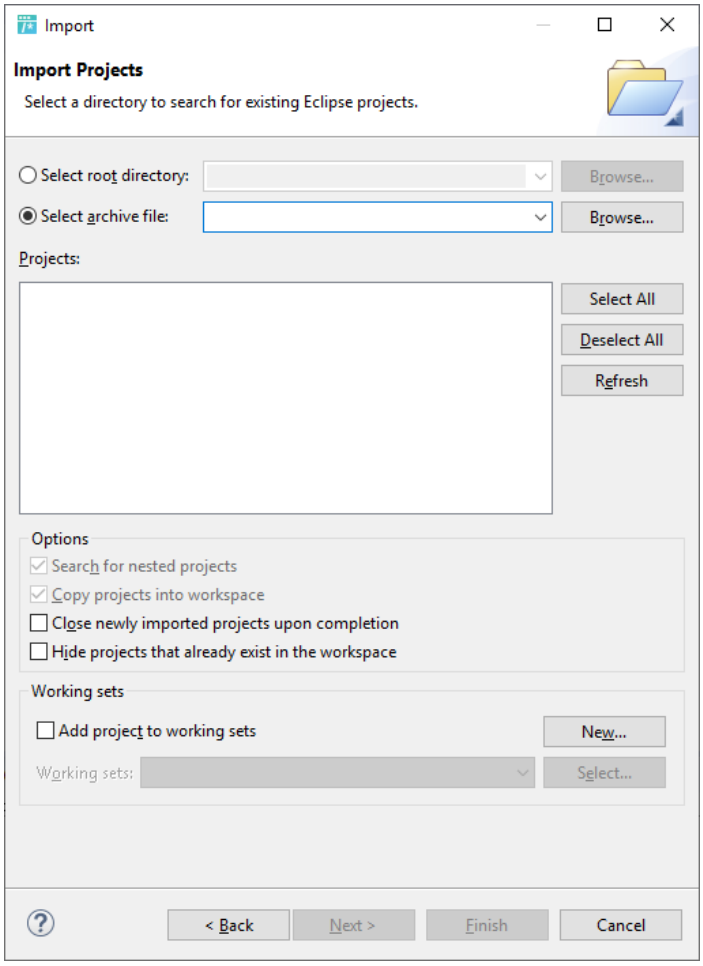
   a. Select **File > Import…**.



   b. Select **General > Existing Projects into Workspace**. And click **Next**.

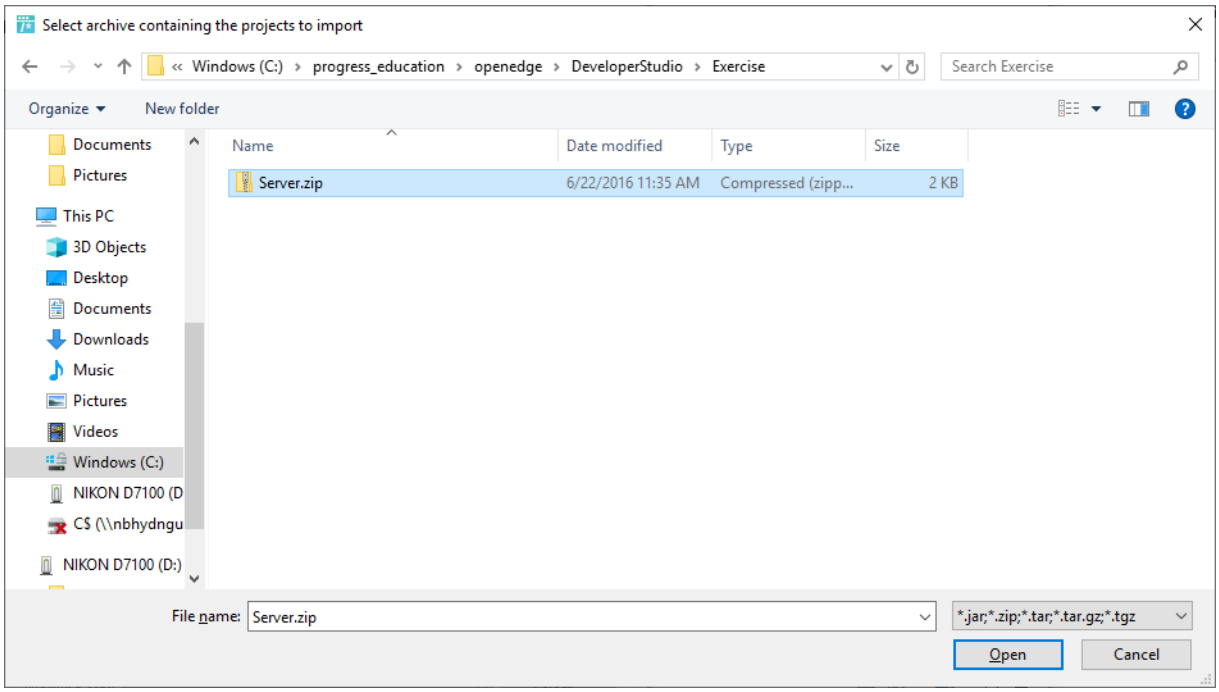

Import
Select
Create new projects from an archive file or directory.
Select an import wizard:
type filter text
General
Archive File
Existing Projects into Workspace
File System
Preferences
Projects from Folder or Archive
Install
Progress OpenEdge
Run/Debug
Team
XML
< Back
Next >
Finish
Cancel

c. Select **Select archive file**.



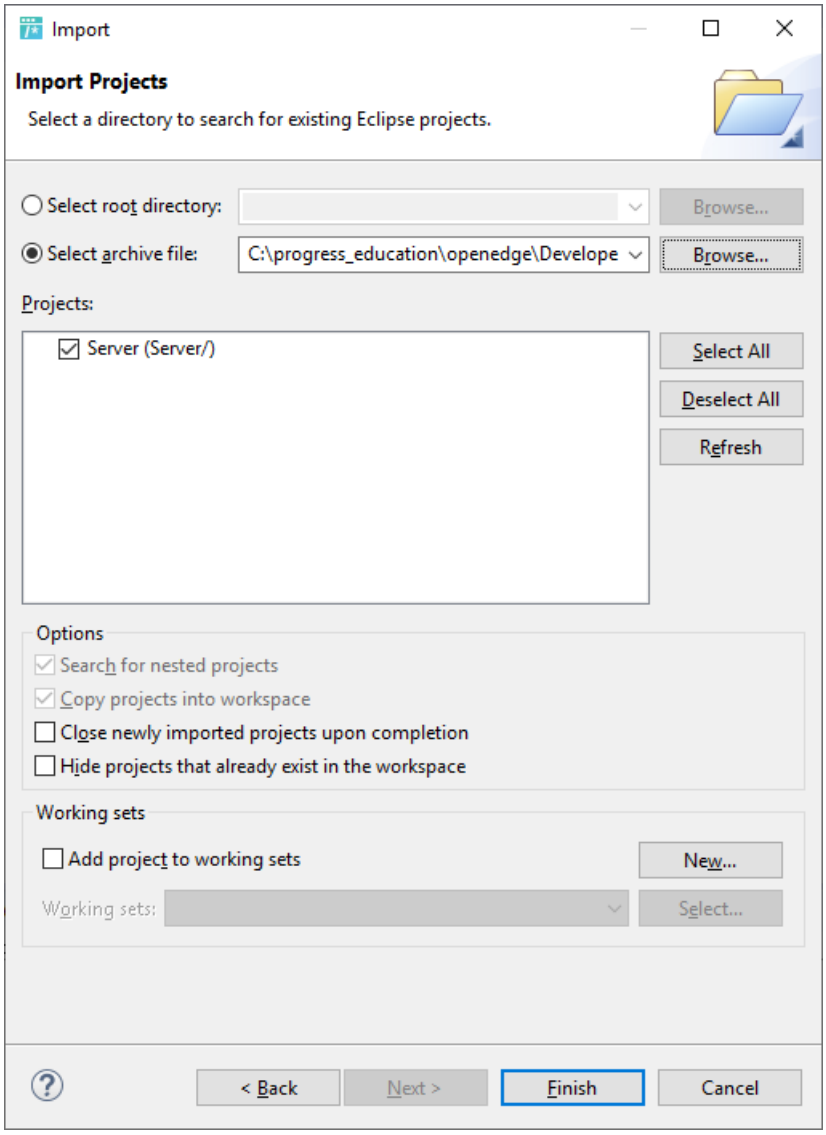
d. Click the browse button and navigate to `C:\progress_education\openedge\DeveloperStudio\Exercise` and select **Server.zip**.

Click **Open**.



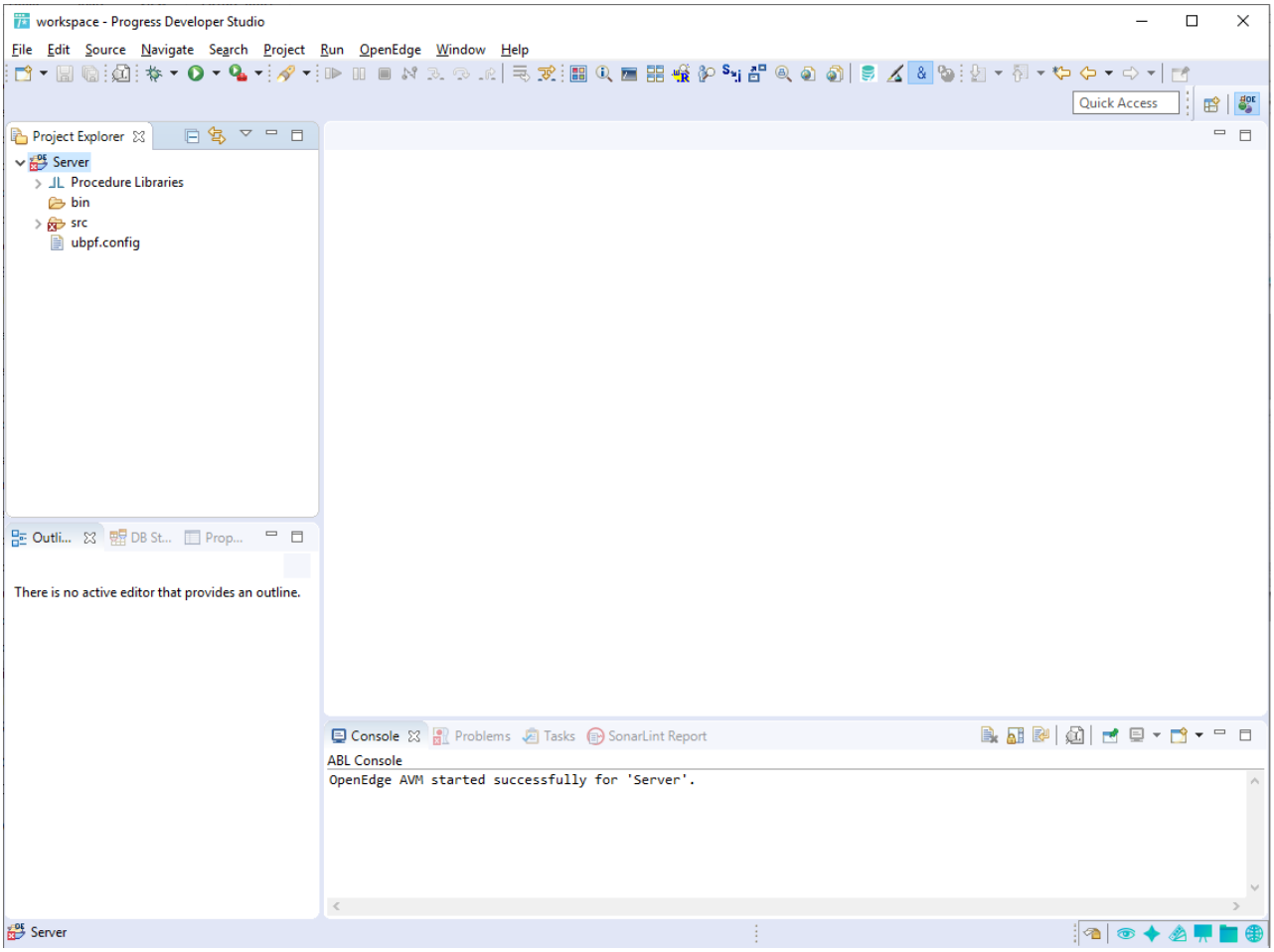
e. Select **Server** in the Projects box.

Click **Finish**.

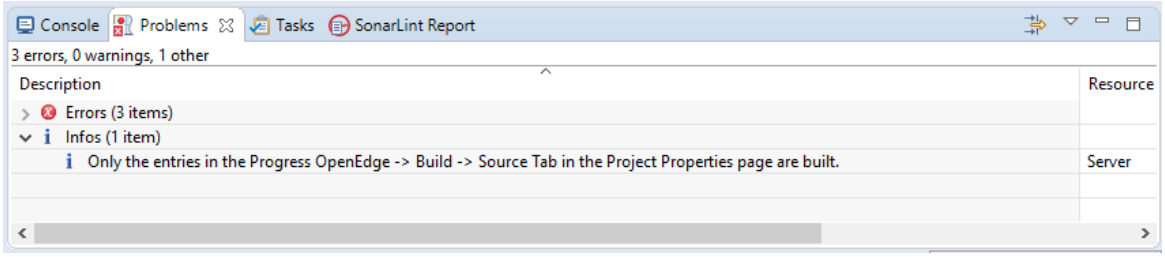


2. When you complete the import, you will see the imported project in your workspace and see that an AVM for the **Server** project has started.

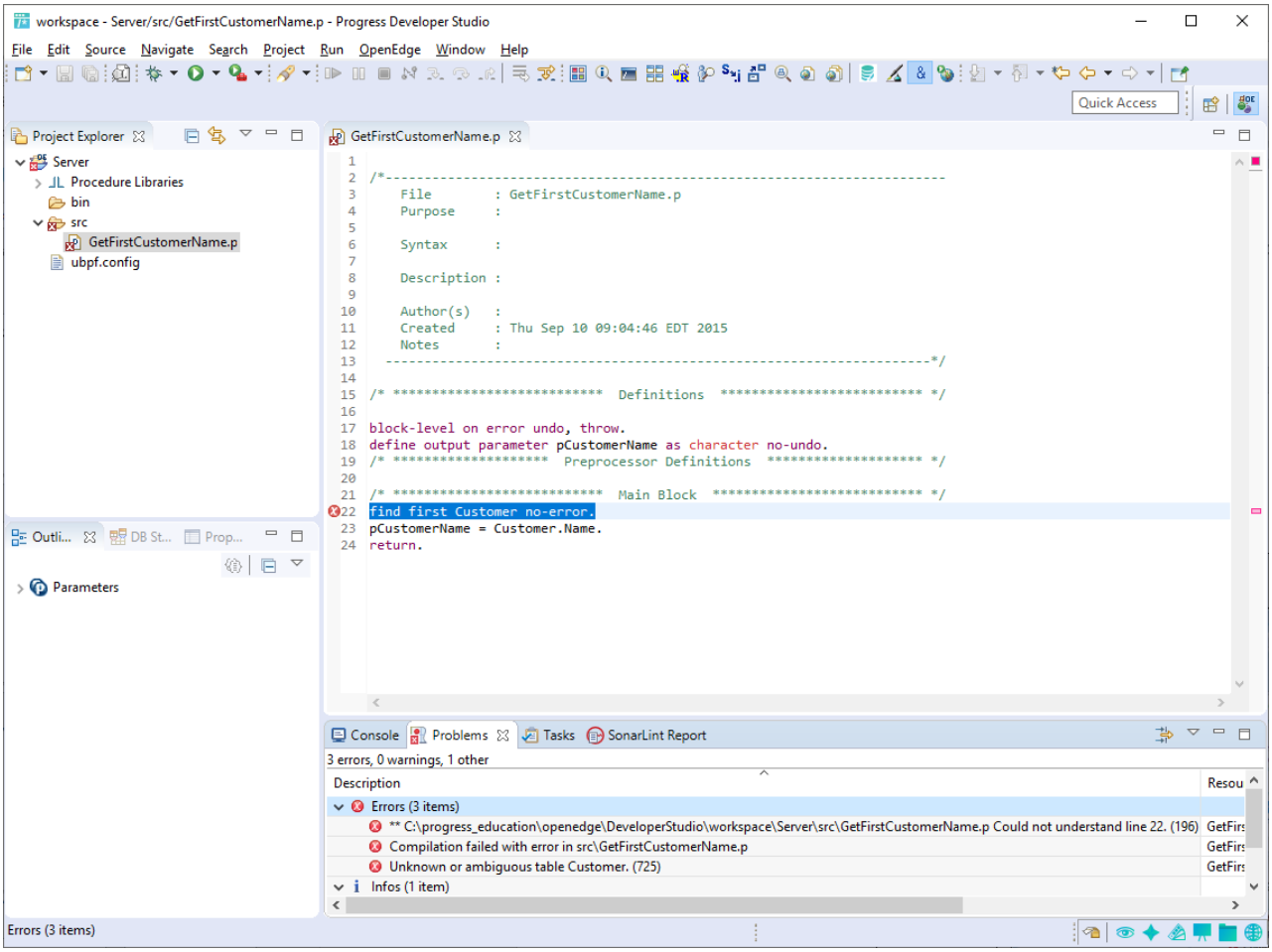
   There will be errors (red “X”) in the code for this project. This is because this project requires a connection to the database.



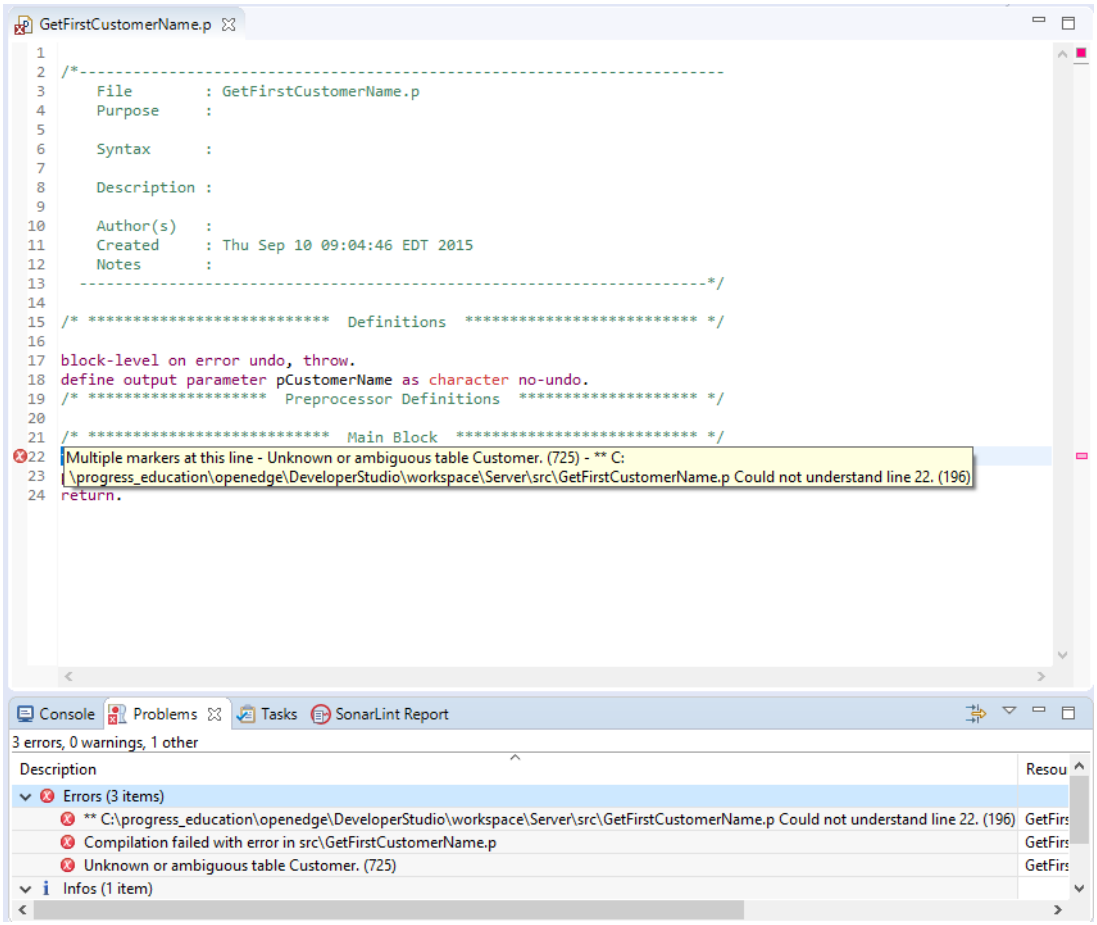
3. Click the **Problems** tab in the lower part of the window. This view lists all compilation errors for the workspace when it was recompiled. You can hover your mouse pointer over the errors to see their details.



4. Double-click any error in the Problems view. It opens the corresponding file in OpenEdge Editor.

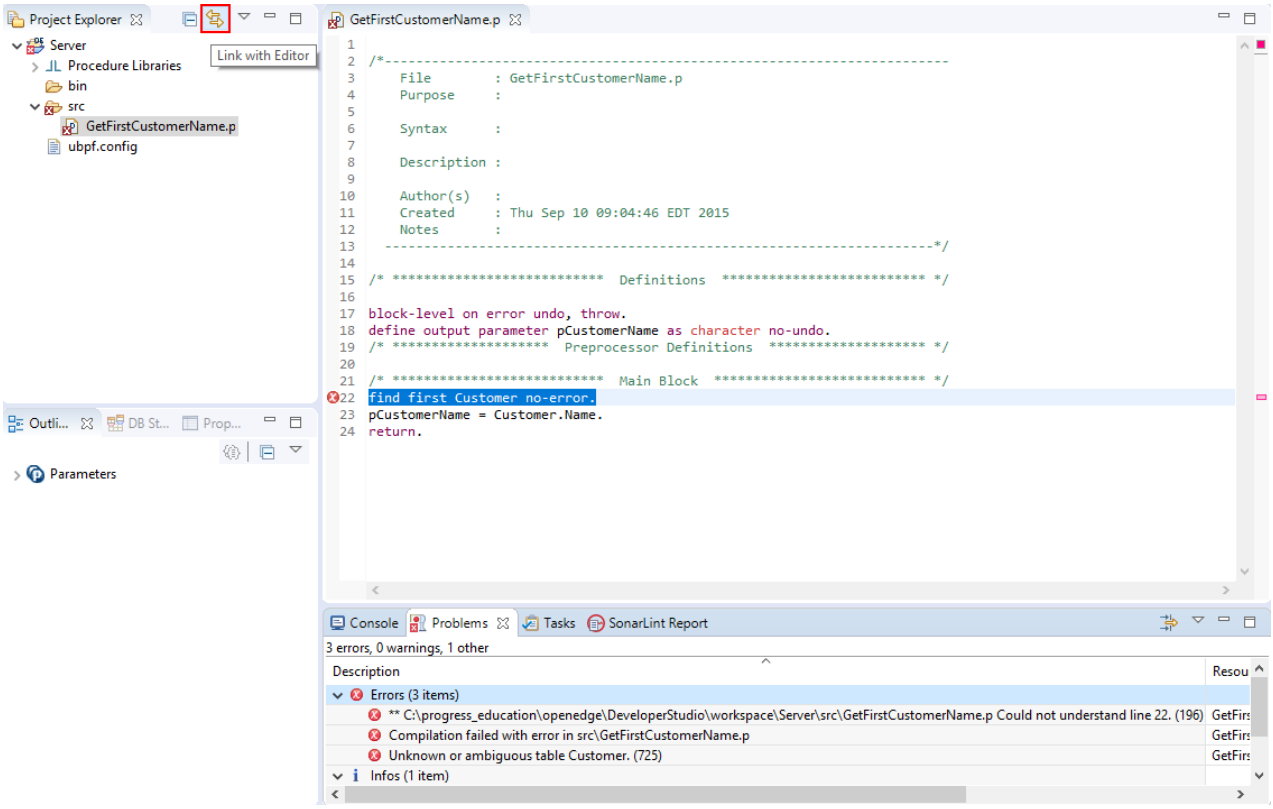


5. If you hover your mouse pointer over the red “X” to the left of the line with the error, the details of the error are also shown in OpenEdge Editor.

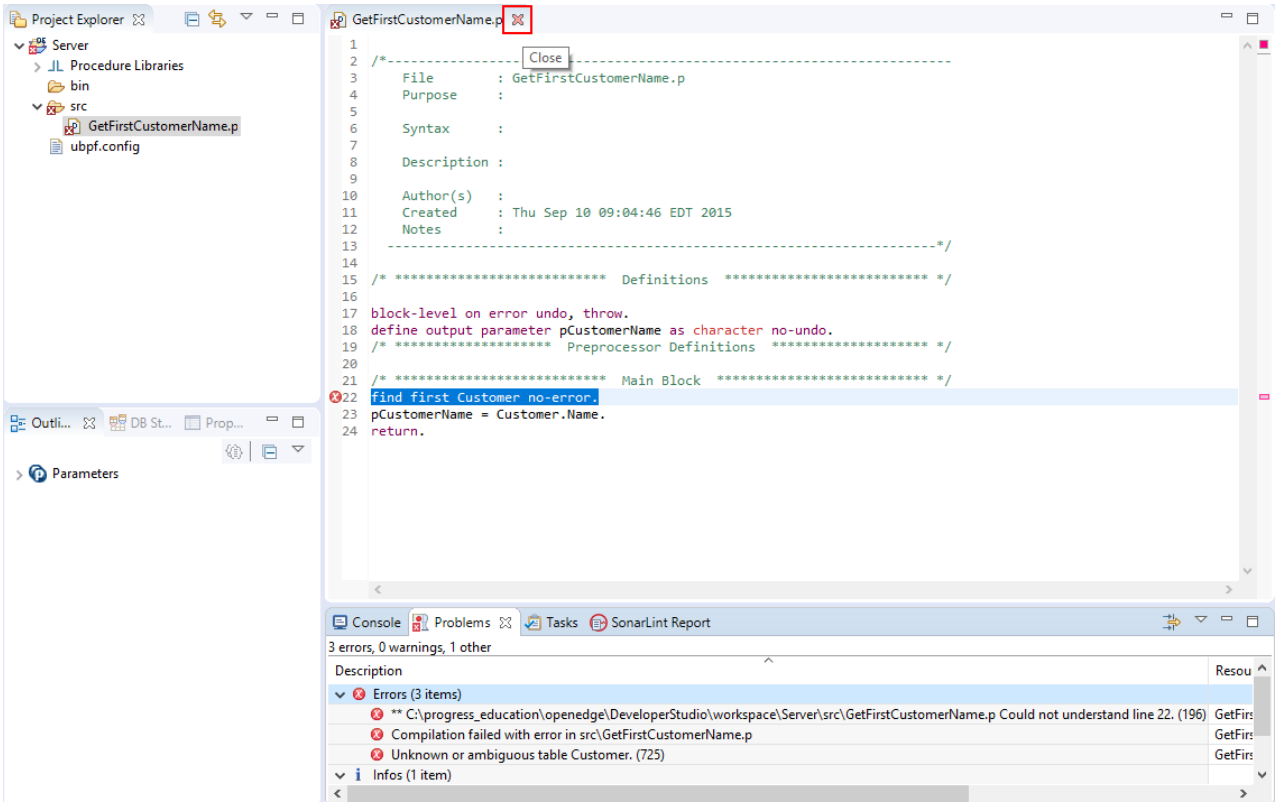


You will correct this error later in this Guided Exercise.

6. Set your workspace so that whatever file you currently have selected in OpenEdge Editor will be selected in Project Explorer by selecting the Link with Editor icon in Project Explorer:



7. Close the procedure file, **GetFirstCustomerName.p**, by clicking the “X’ in the tab for the file.

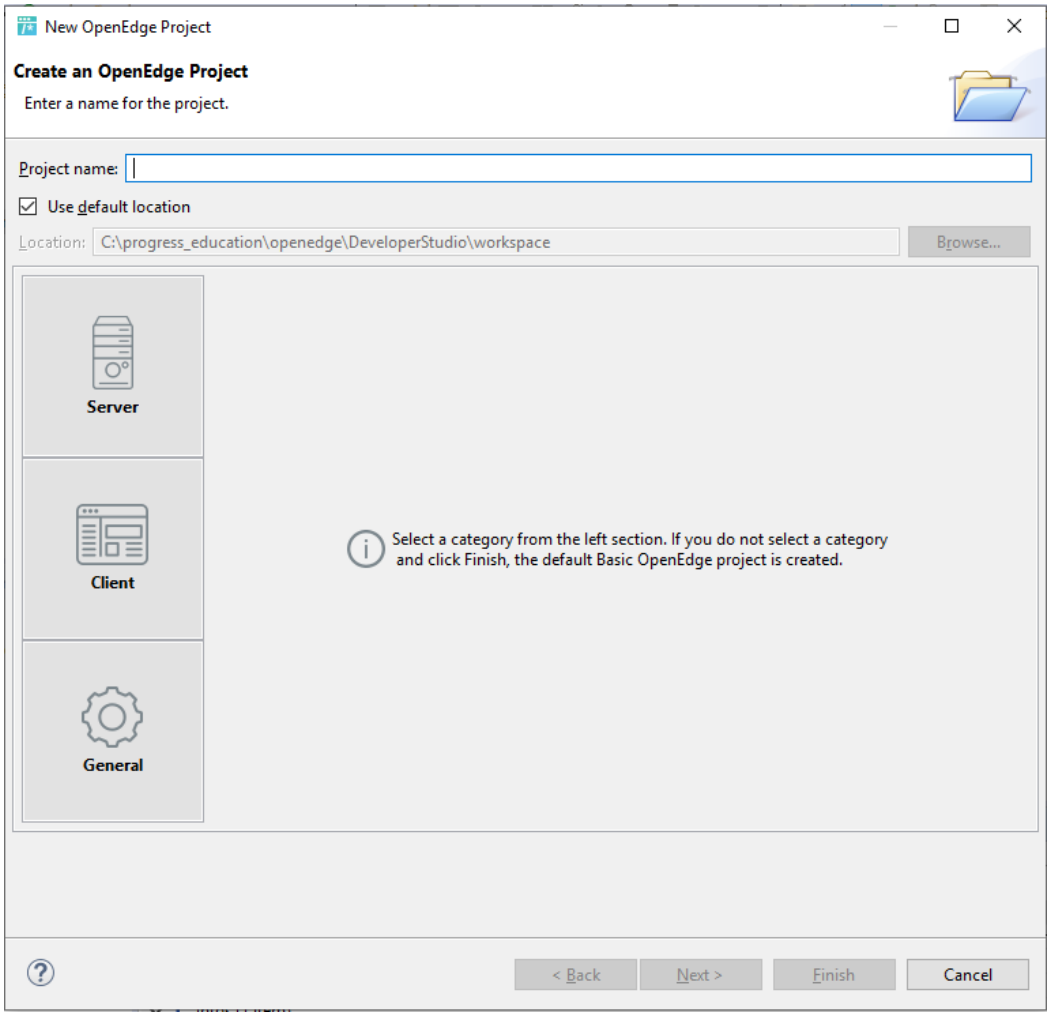


Part 2—Creating a new project and importing a file

Now you will create a project for the client side of the sample application. Then you will import a file into the project you create.

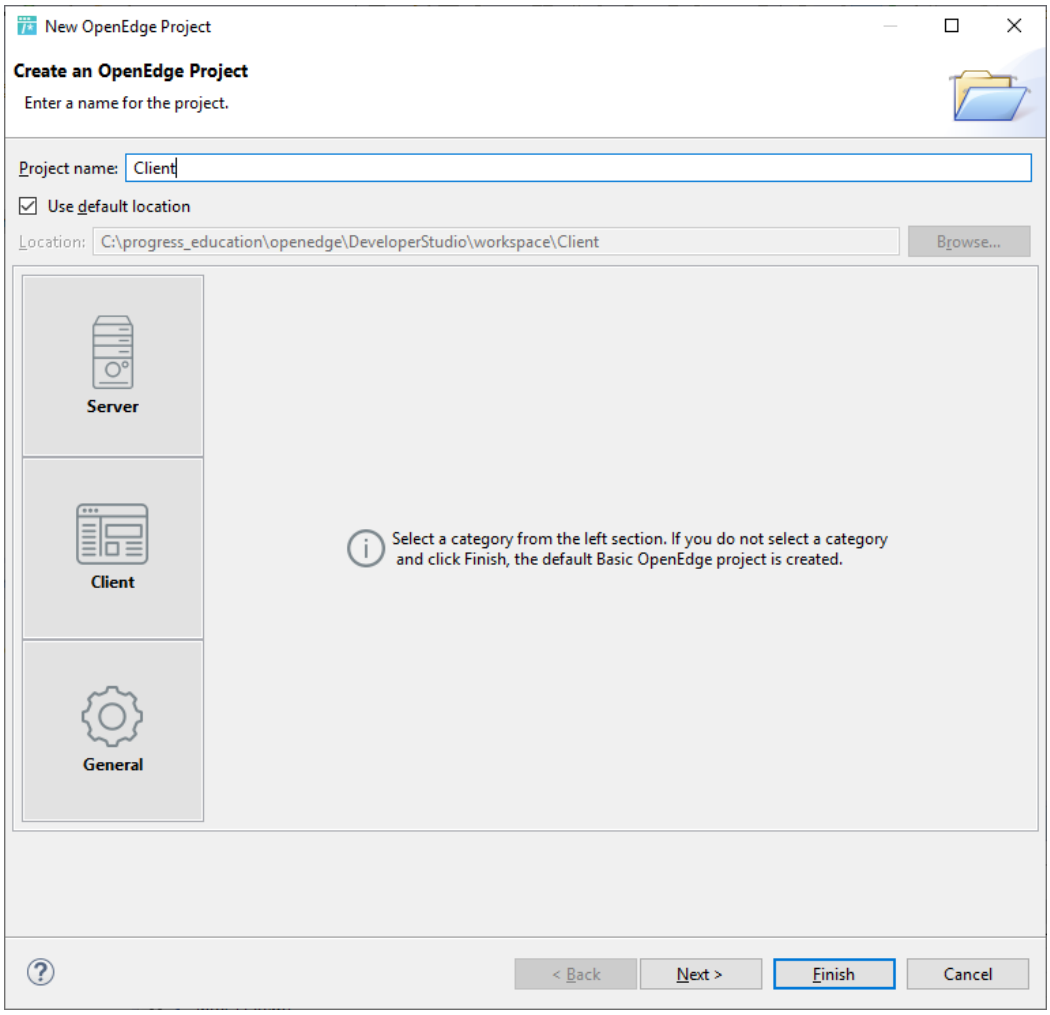
1. Create an OpenEdge project with this information:
   a. Name: **Client**
   b. Ensure 'Use default location' is checked, which is the workspace folder.
   c. Source and r-code directories will be named **src** and **bin**.
      a. Navigate to **File > New**.
      b. Notice that in Developer Studio you can create a number of resources that are OpenEdge-specific for ABL development tasks. In this case, select **OpenEdge Project**.
      c. Notice that you can create server, client, or general OpenEdge projects. Each type of OpenEdge project contains its own specific facets. Here, create a General project.

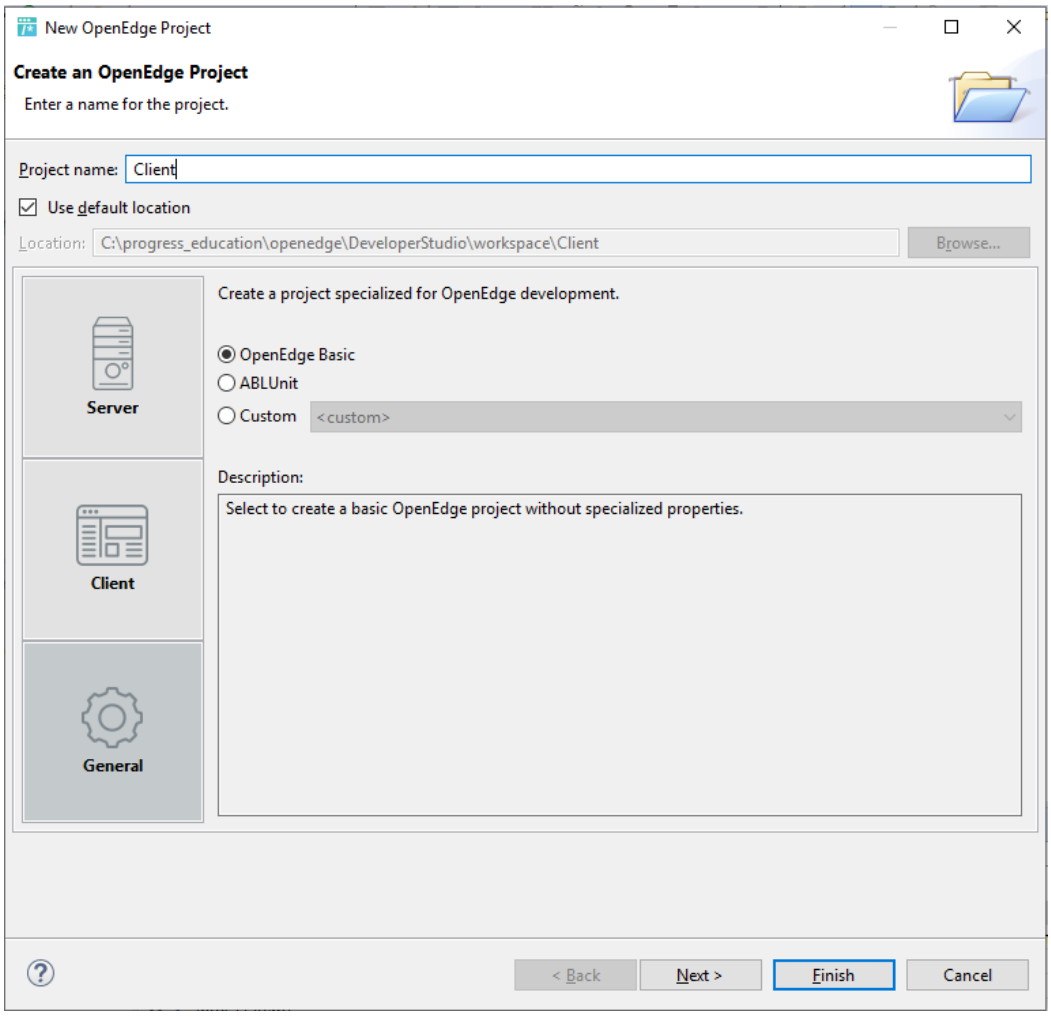
         Notice also that when you create a project, Developer Studio sets the location of the project to the same location as the workspace.



      d. Enter **Client** as the project name and click the **General** icon to create an **OpenEdge Basic** project.

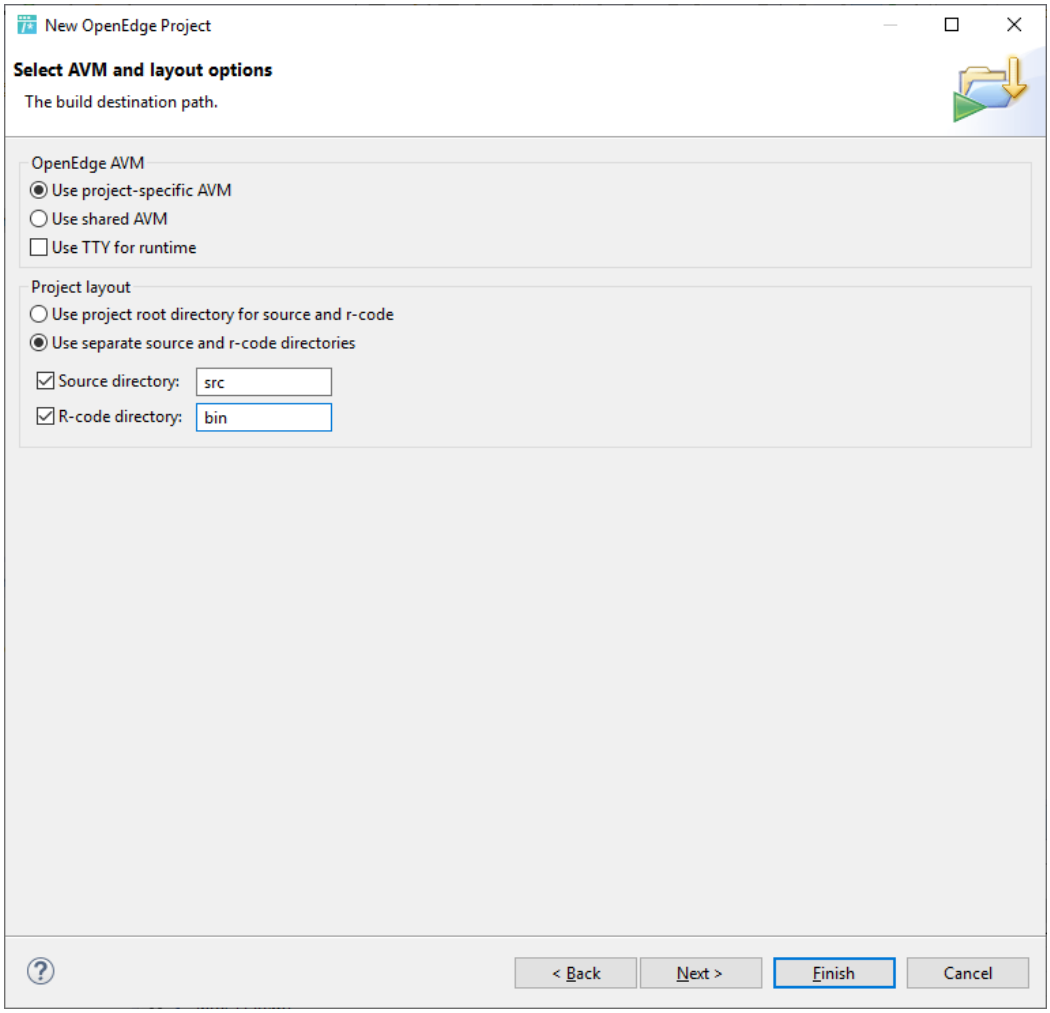
         Click **Next**.


New OpenEdge Project
Create an OpenEdge Project
Enter a name for the project.
Project name: Client
Use default location
Location: C:\progress_education\openedge\DeveloperStudio\workspace\Client
Browse...
Server
Client
General
Select a category from the left section. If you do not select a category and click Finish, the default Basic OpenEdge project is created.
< Back
Next >
Finish
Cancel


New OpenEdge Project
Create an OpenEdge Project
Enter a name for the project.
Project name: Client
Use default location
Location: C:\progress_education\openedge\DeveloperStudio\workspace\Client
Browse...
Server
Client
General
Create a project specialized for OpenEdge development.
OpenEdge Basic
ABLUnit
Custom
<custom>
Description:
Select to create a basic OpenEdge project without specialized properties.
< Back
Next >
Finish
Cancel

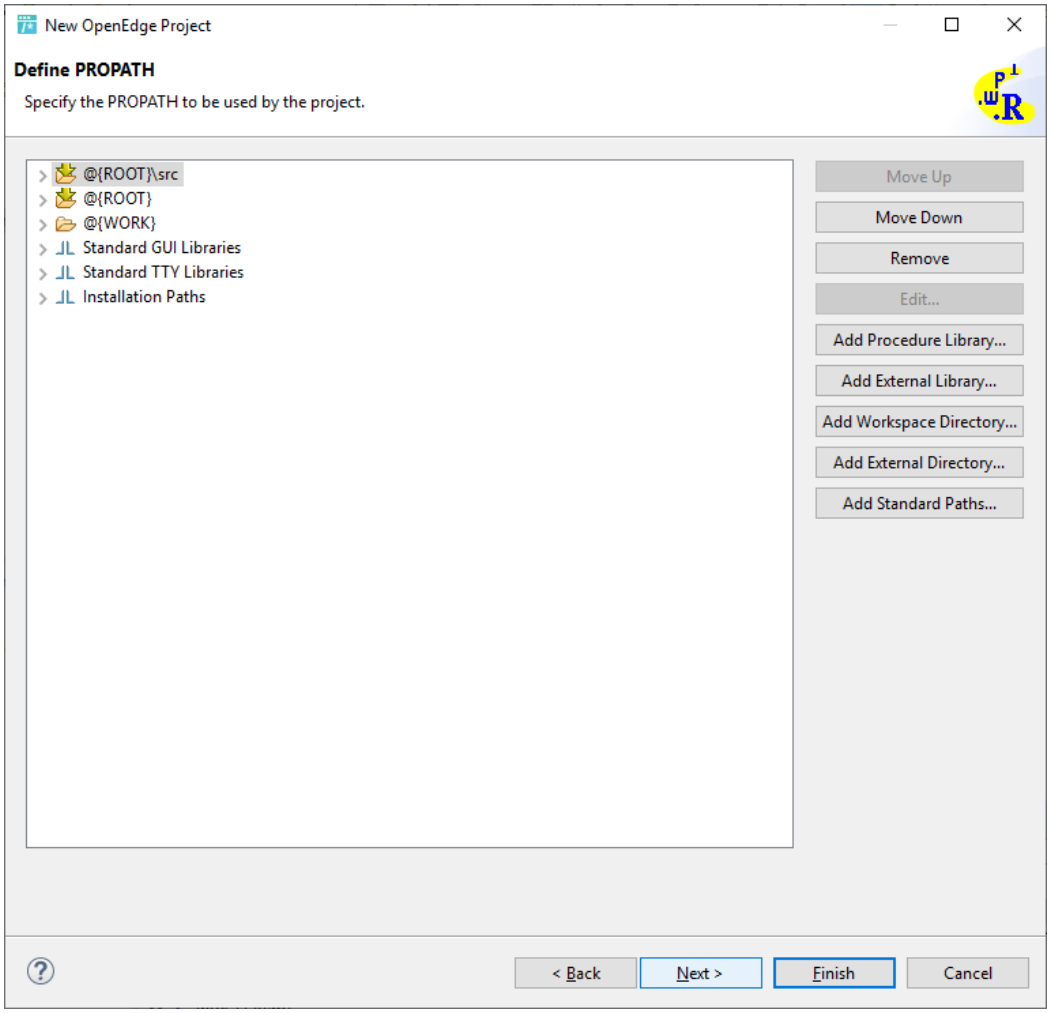
e. Next, specify **Use project-specific AVM** and **Use separate source and r-code directories** and then type **src** in the **Source directory** and **bin** in the **R-code directory** field.

Click **Next**.



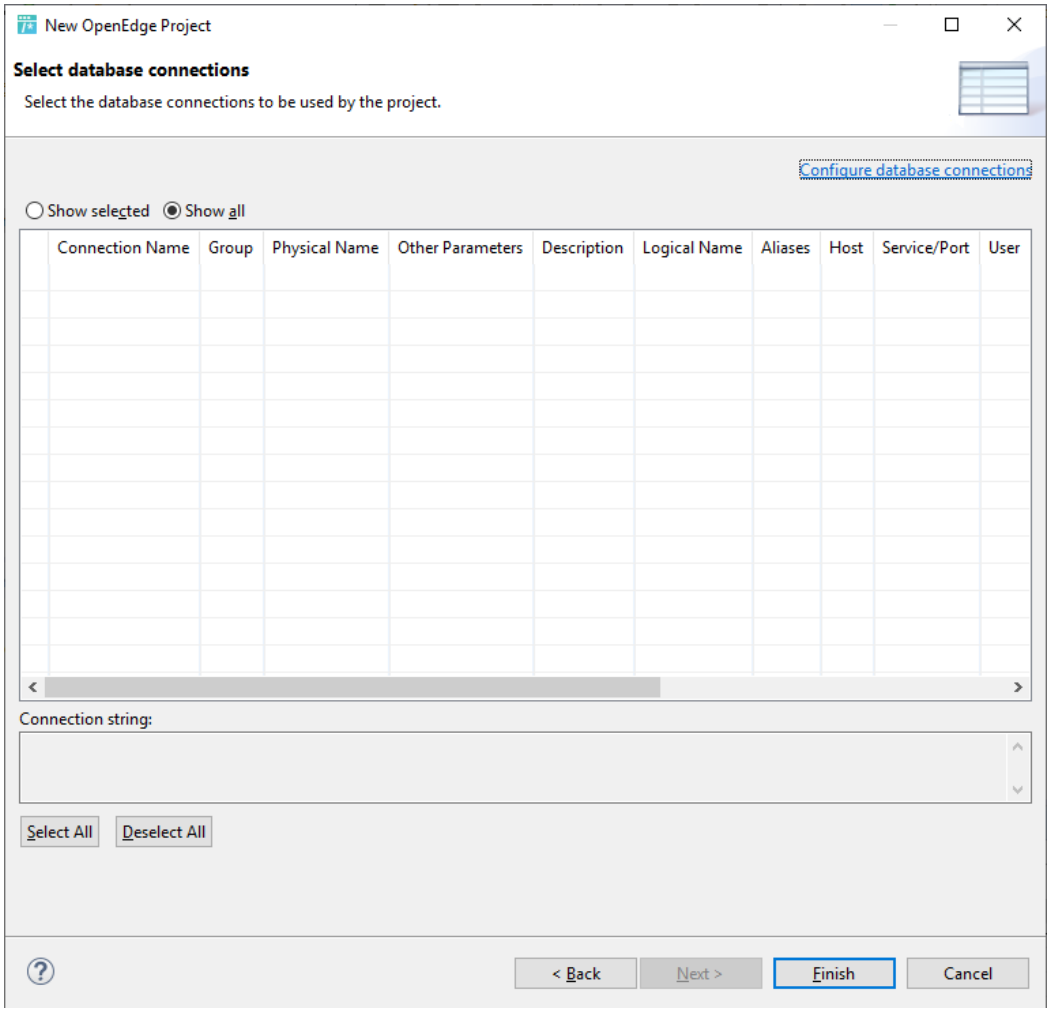
f. This is a new project, so keep the default PROPATH settings for now.

Click **Next**.

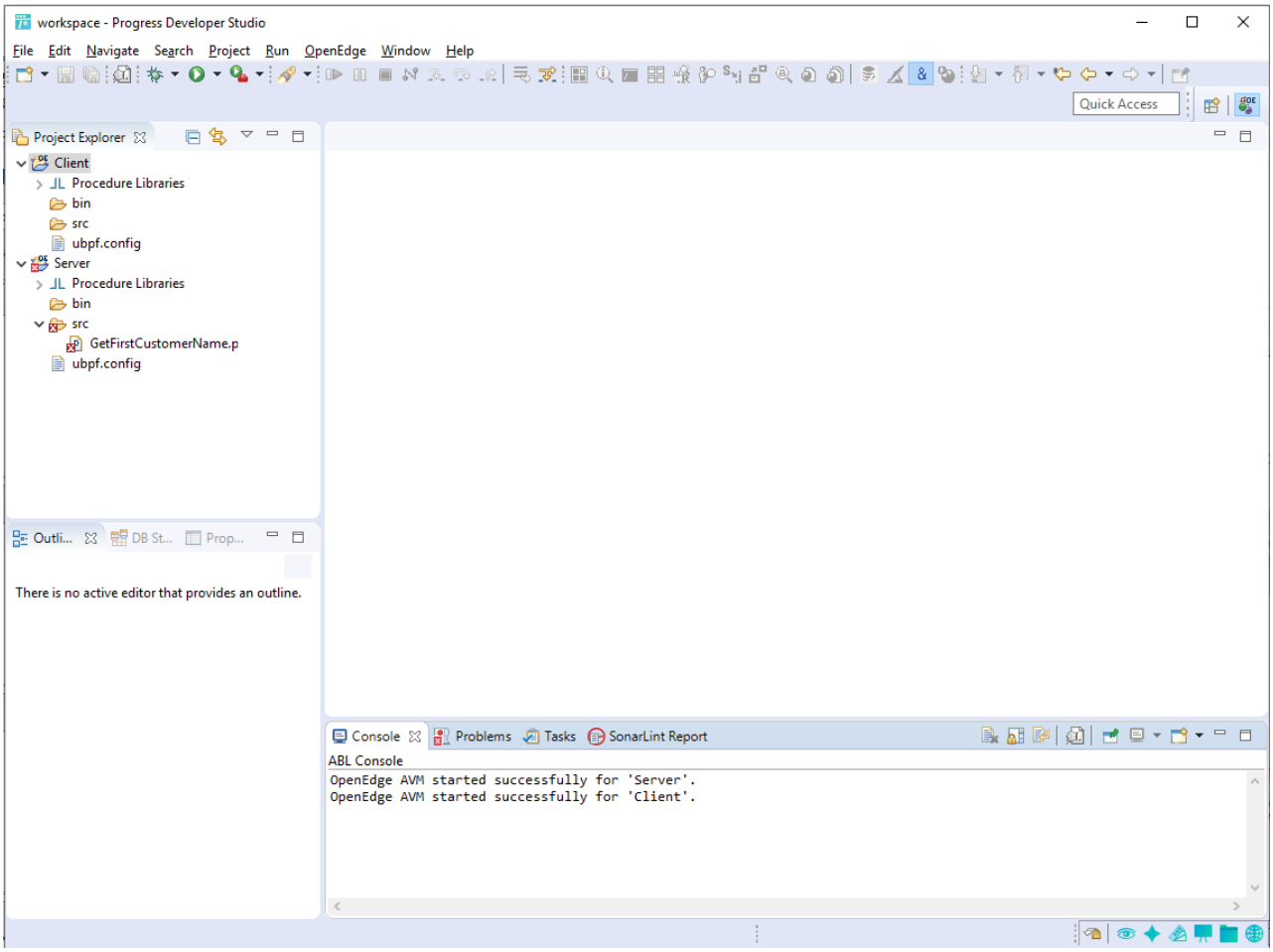


g. Since this project contains client code, do not specify a connection to the database.

Click **Finish**.



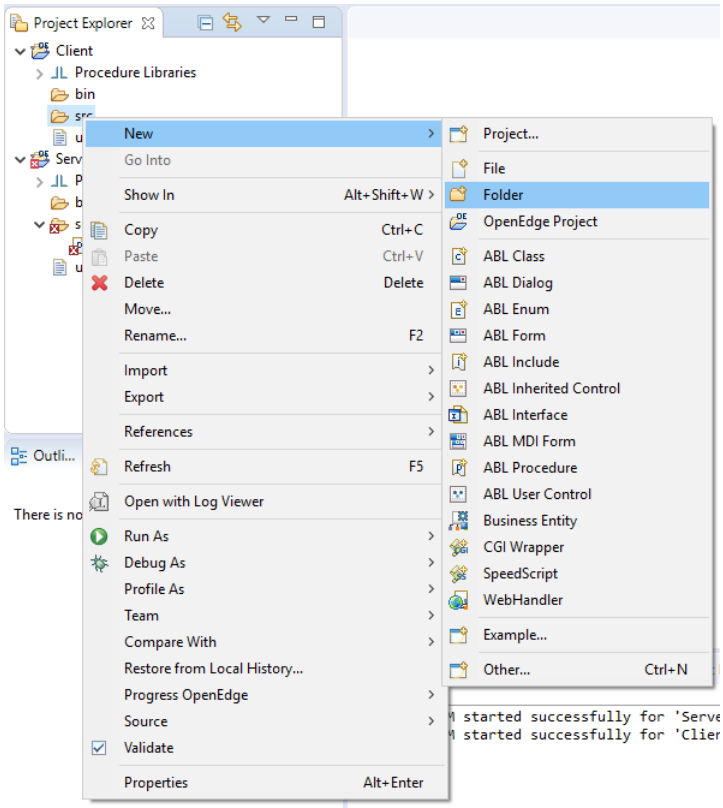
h. Here you should see your newly created project, with **src** and **bin** folders, in Project Explorer The **src** folder will contain ABL source code.



2. Add a subfolder to the **src** folder of the **Client** project as follows:

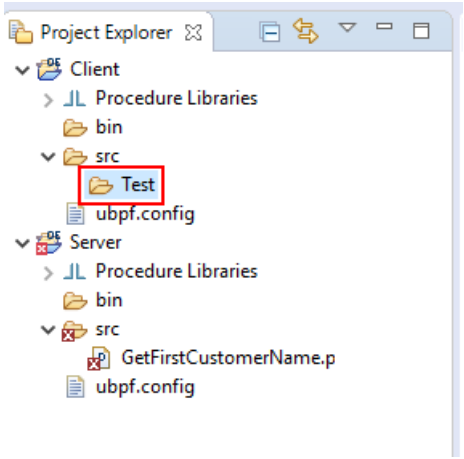
   a. In Project Explorer, navigate to the **src** folder under the **Client** project.

   b. Right-click **src** and then select **New > Folder**.



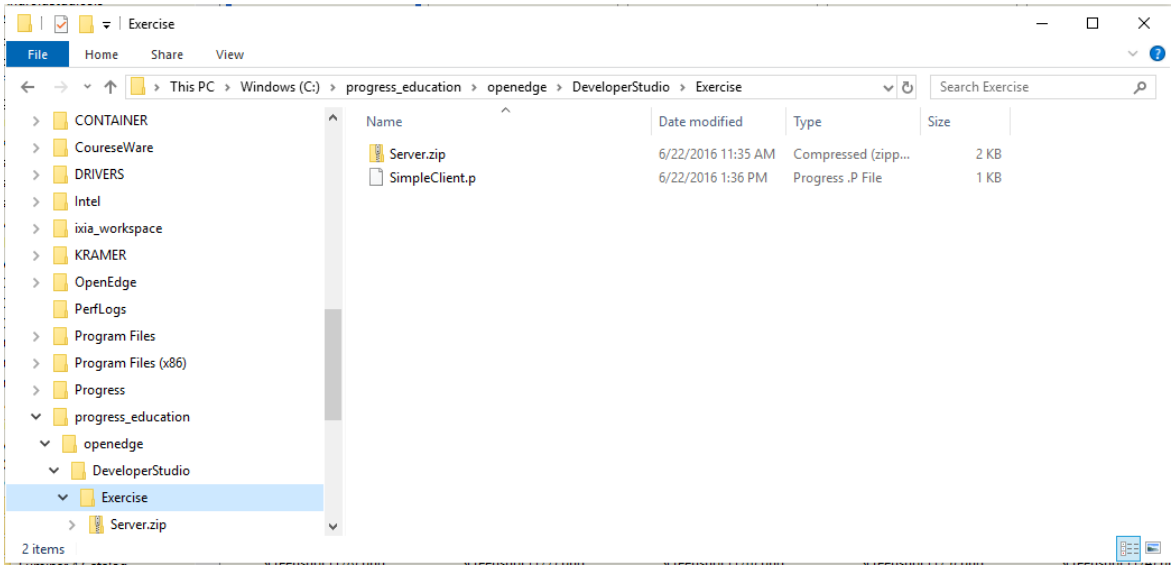
   c. Enter **Test** as the name of the new folder.

   d. Click **Finish**.

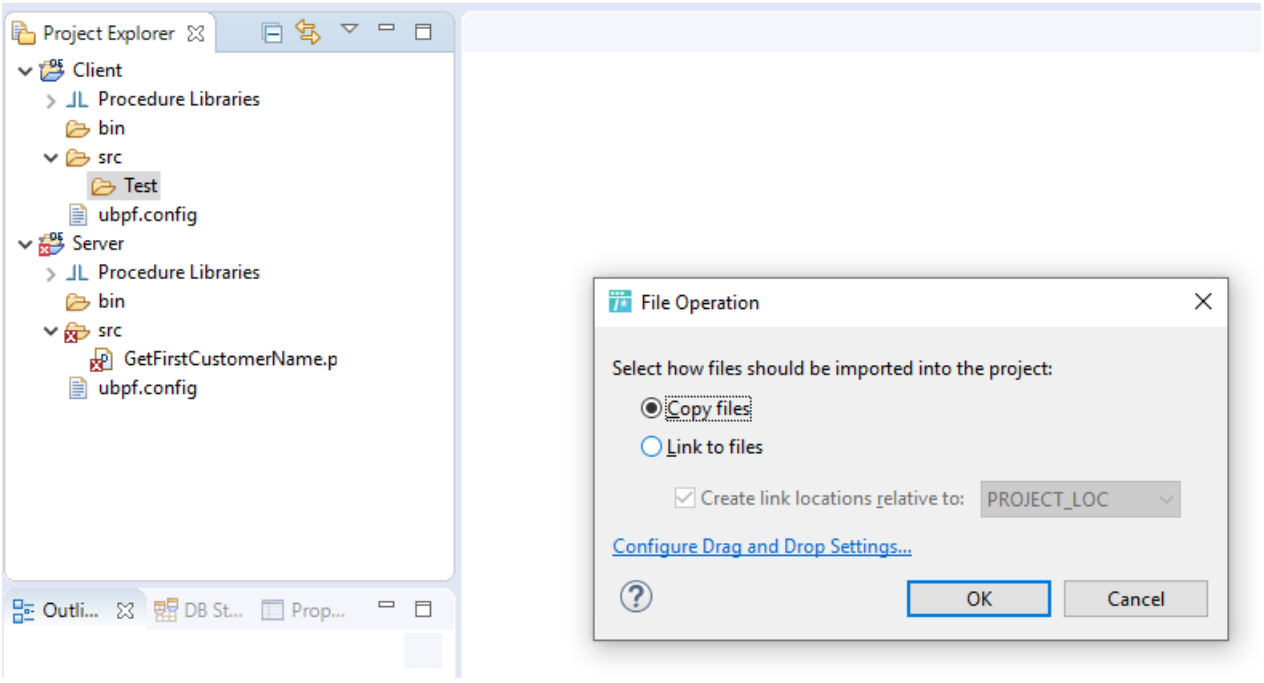


3. Import an ABL source file into your project from `C:\progress_education\openedge\DeveloperStudio\Exercise`. The source file is: **SimpleClient.p**.

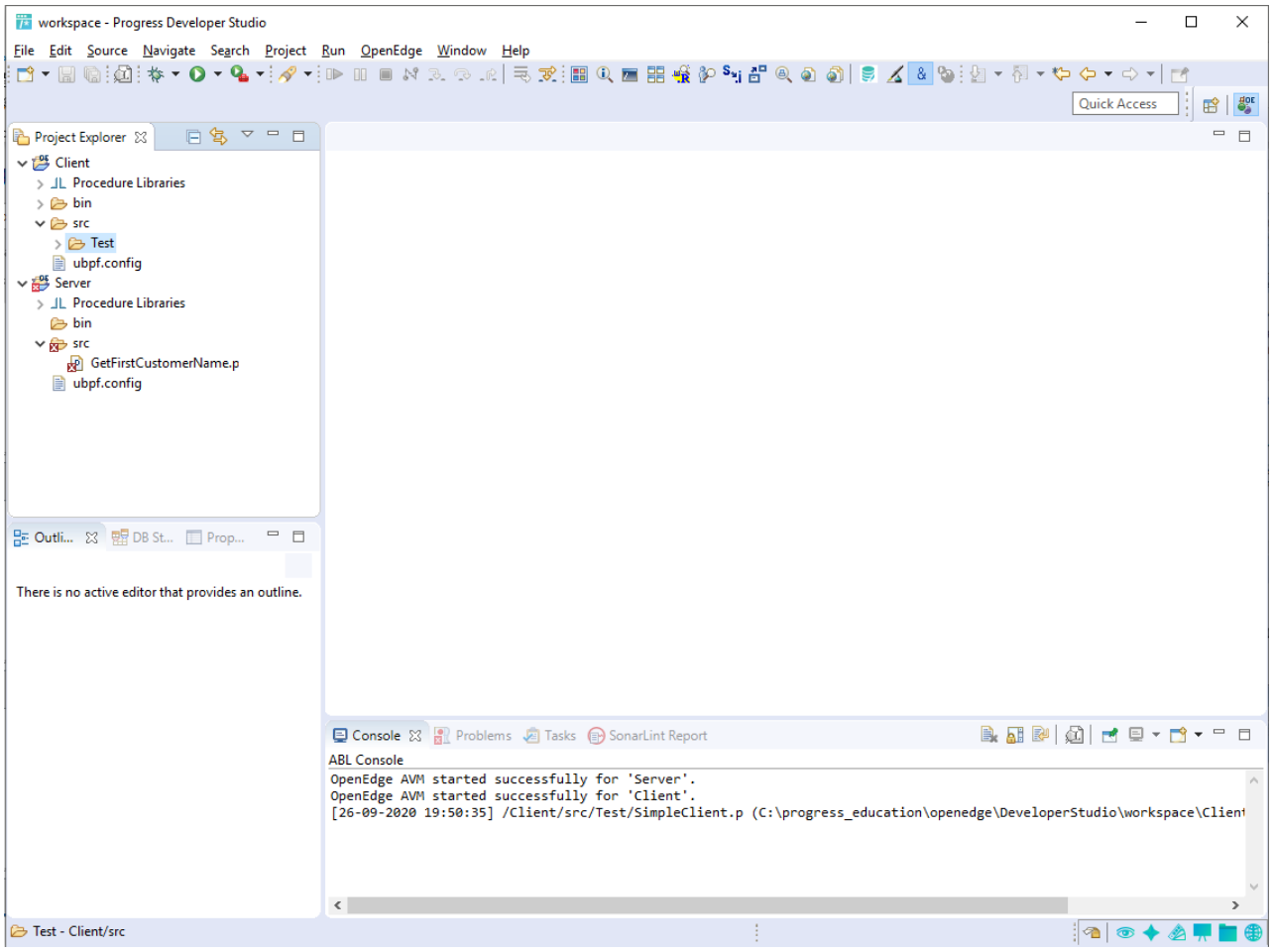
   a. Launch Windows Explorer and browse to `C:\progress_education\openedge\DeveloperStudio\Exercise`.



   b. Select **SimpleClient.p**. Then drag and drop the file onto the **Client > src > Test** folder in the Project Explorer view of Developer Studio.

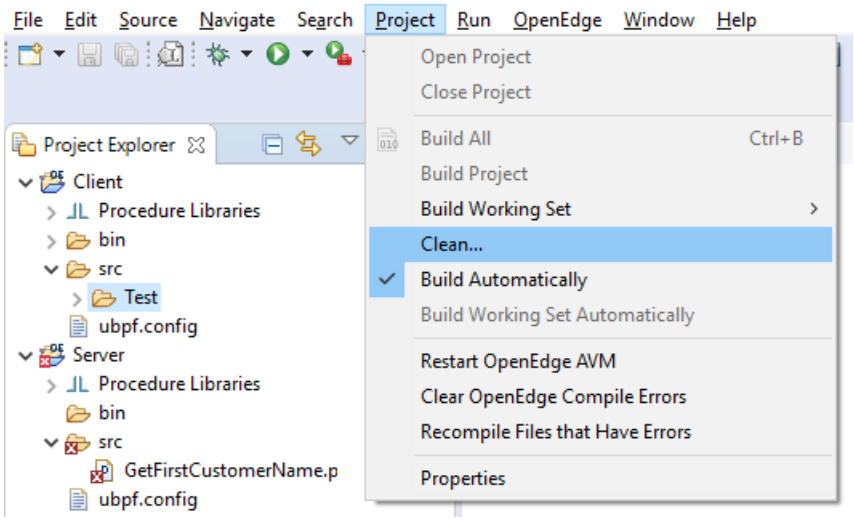


   c. Select **Copy files** and then click **OK**.

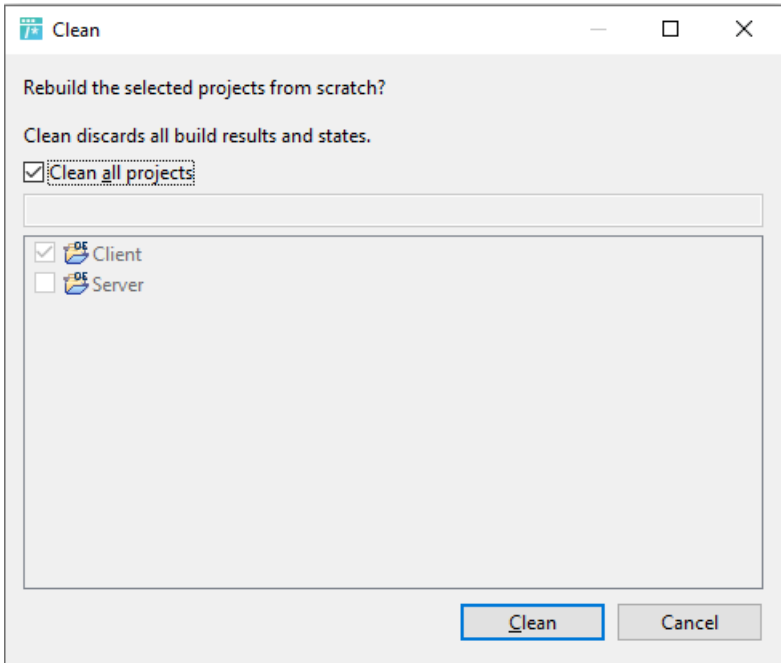

workspace - Progress Developer Studio
File Edit Source Navigate Search Project Run OpenEdge Window Help
Quick Access
Project Explorer
Client
Procedure Libraries
bin
src
Test
ubpf.config
Server
Procedure Libraries
bin
src
GetFirstCustomerName.p
ubpf.config
Outli...
DB St...
Prop...
There is no active editor that provides an outline.
Console
Problems
Tasks
SonarLint Report
ABL Console
OpenEdge AVM started successfully for 'Server'.
OpenEdge AVM started successfully for 'Client'.
[26-09-2020 19:50:35] /Client/src/Test/SimpleClient.p (C:\progress_education\openedge\DeveloperStudio\workspace\Client
Test - Client/src

4. Clean and recompile the entire workspace:

   a. Select **Project > Clean…**.



   b. Click **Clean**.



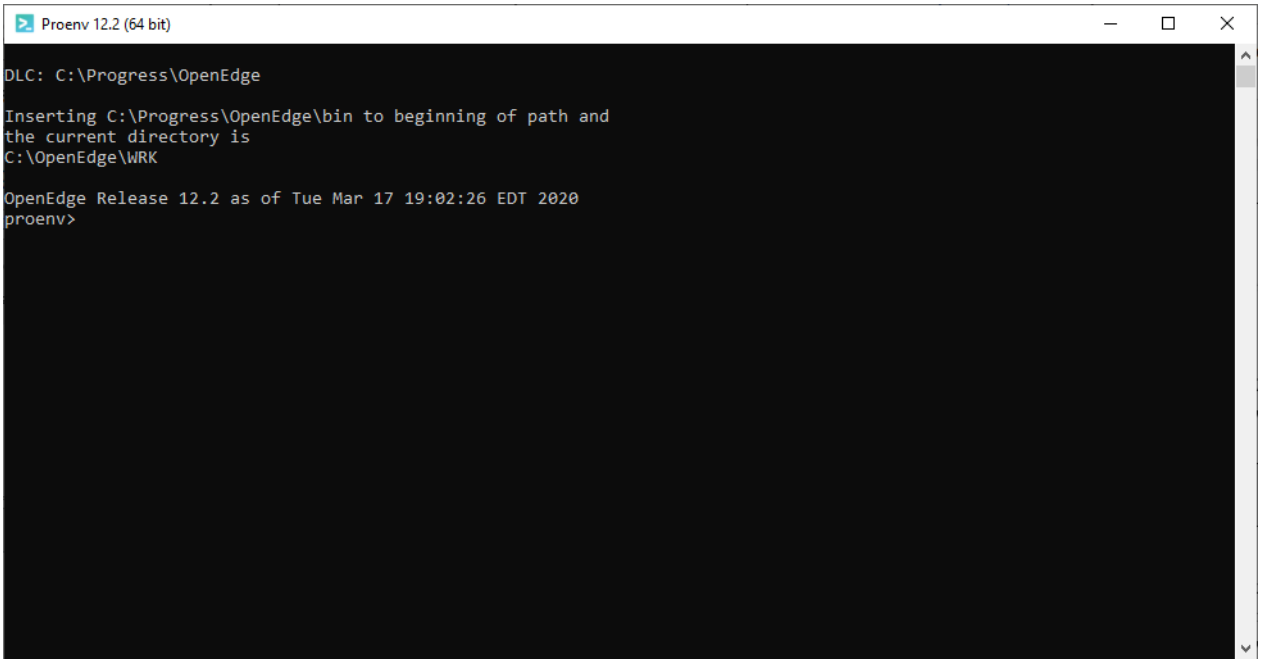
Note: You should always clean your workspace after importing projects and files.

Part 3—Starting the Database Server and configuring the Server project that uses it

In this part of the exercise you start the Database Server for the sports2020 database and configure the Server project to use a connection to this database.

1. Select **Start > Progress > Proenv 12.2 (64 bit)**.

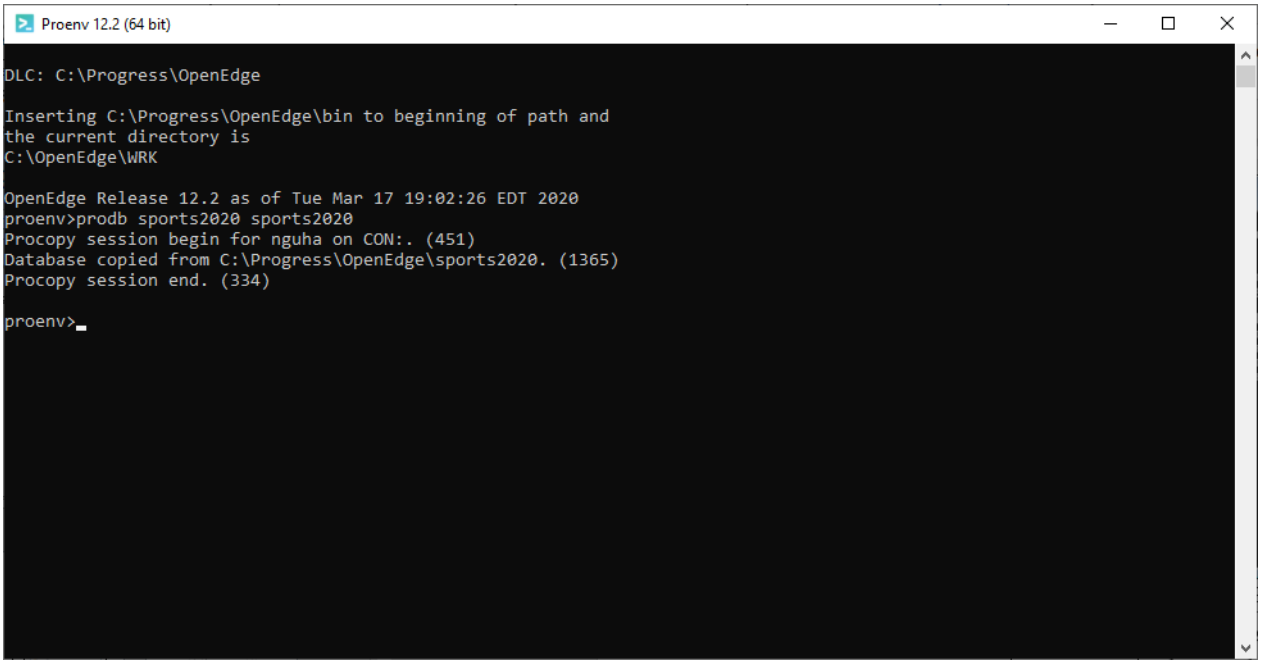
   A new window that is set up for a Progress OpenEdge session opens. By default, `C:\OpenEdge\WRK` is the OpenEdge working directory where the database will be created.



2. Type the following command to create the sports2020 database:

   prodb sports2020 sports2020

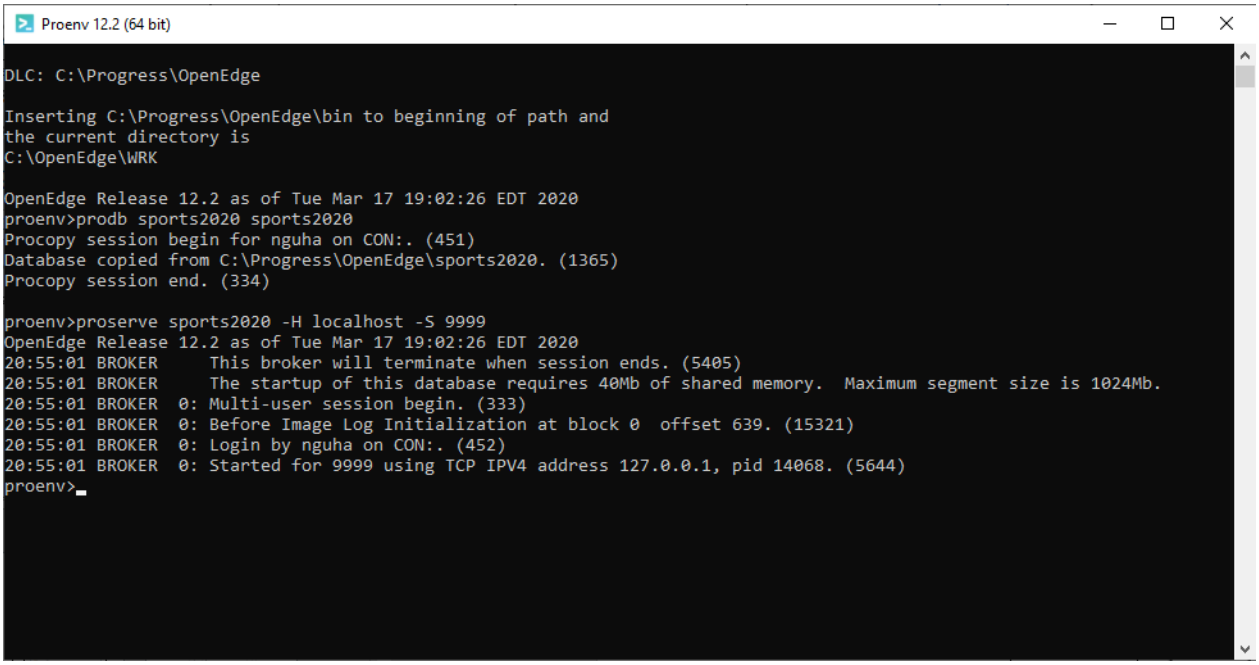
   The database files are now added to the `C:\OpenEdge\WRK` directory and the database is ready for starting. All the database-related file names begin with *sports2020*.



3. Type the following command:

proserve sports2020 -H localhost –S 9999

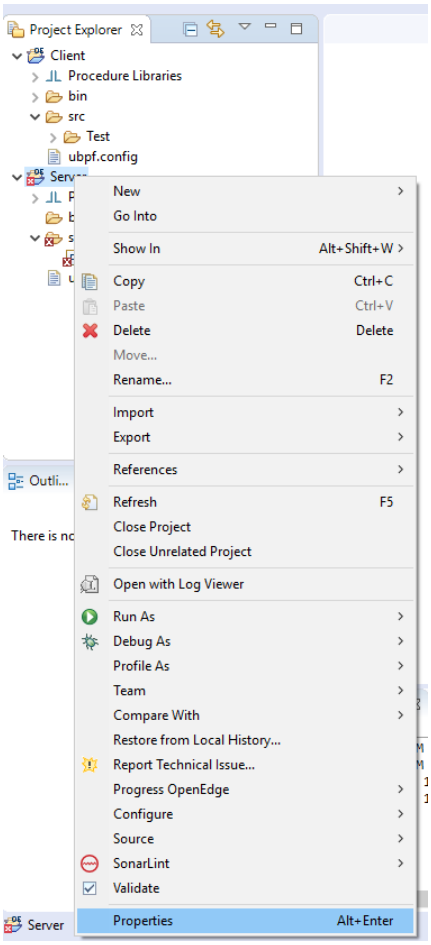
The Database Server for this database will start. When the Database Server is started, a **sports2020.lk** file is created; it remains in the directory with the other database files as long as the Database Server runs.



Note: If port 9999 is unavailable on your system, use a different port number.

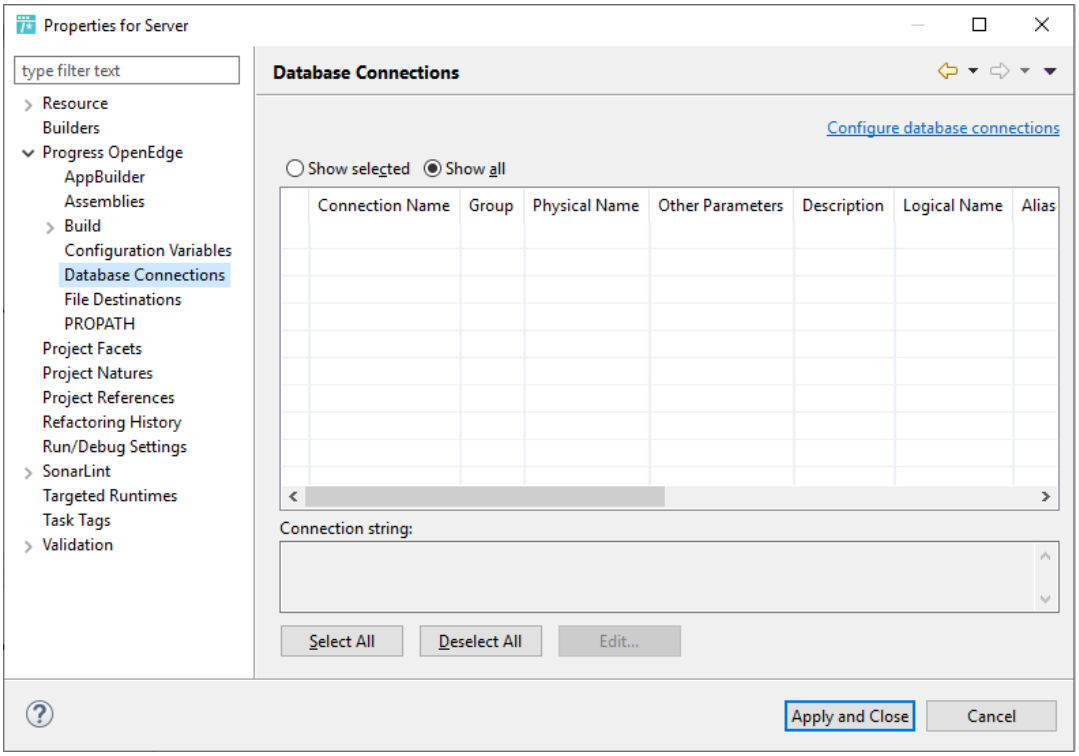
4. Close the Proenv window by clicking the “X” in the top right corner or by typing **exit** in the window. The Database Server will continue to run.

5. In Project Explorer, right-click the **Server** project and then select **Properties**.

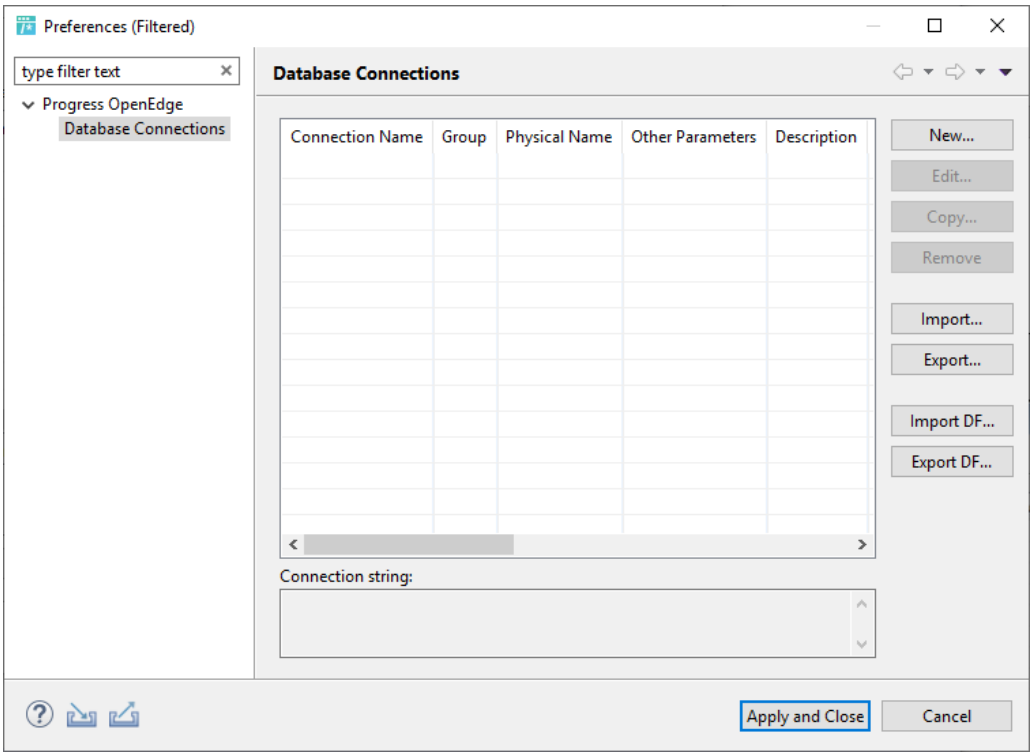


6. Navigate to **Progress OpenEdge > Database Connections**.

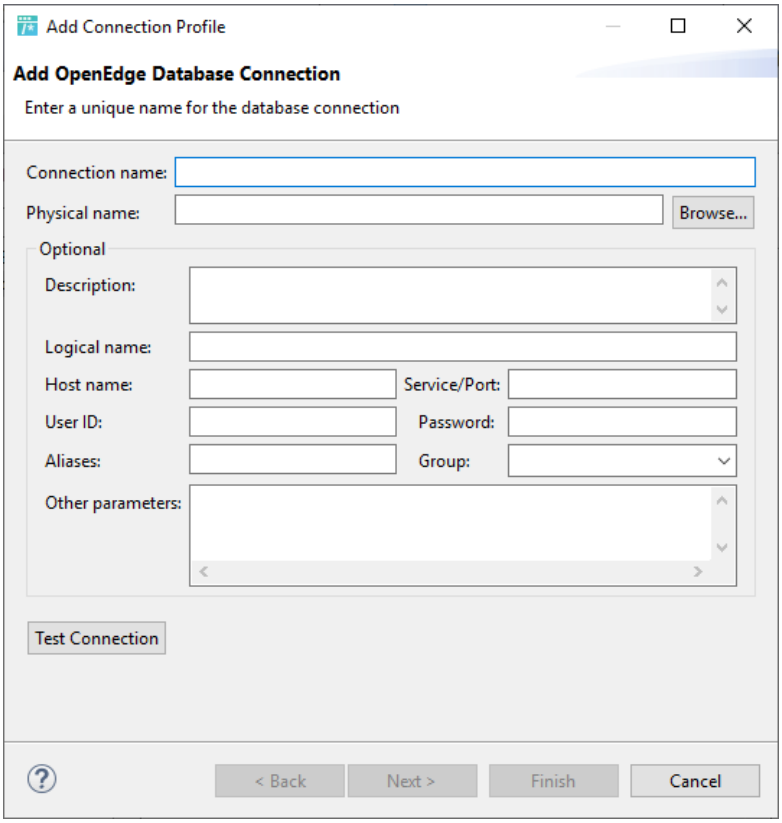
   Click the link **Configure database connections**.



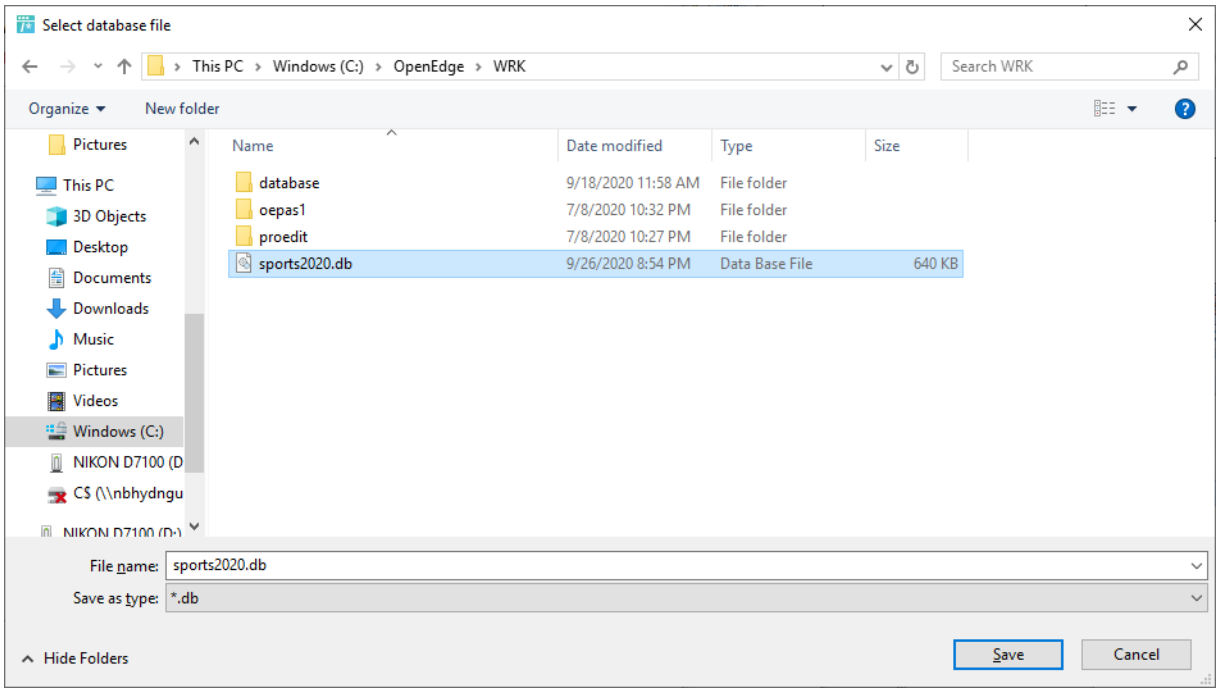
7. Click **New…** to open the *Add Connection Profile* wizard.



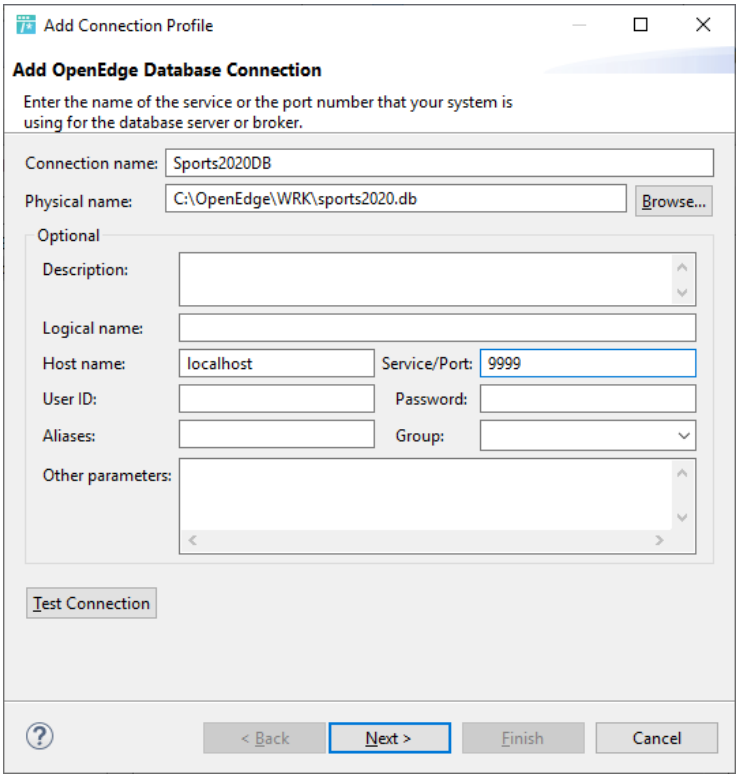
8. Enter the connection name, **Sports2020DB**.



9. In the Physical name area:
   a. Click the **Browse** button and then navigate to and select `C:\OpenEdge\WRK\sports2020.db`.
   b. Click **Save**.

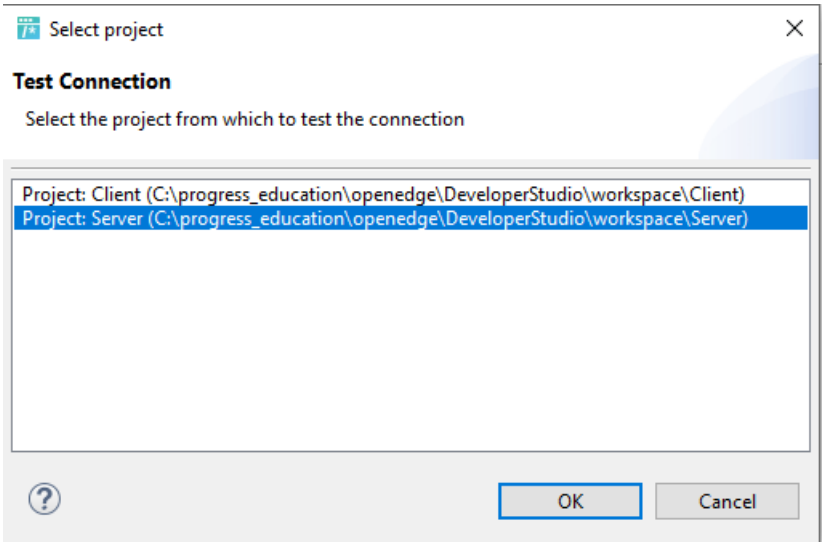


10. Specify the host name as **localhost**.
11. Specify **9999** as the port number.



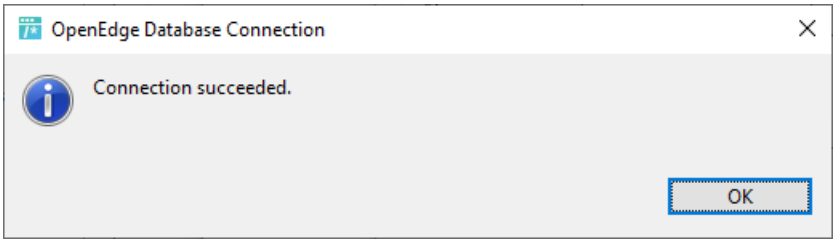
If port 9999 is unavailable on your system, use a different port number. It must be the same port number that you specified when you started the Database Server.

12. Click the **Test Connection** button.
13. The wizard prompts you about which project in your workspace to use for the test because, to run the code to test the connection, you need an AVM. Select the **Server** project and then click **OK**.

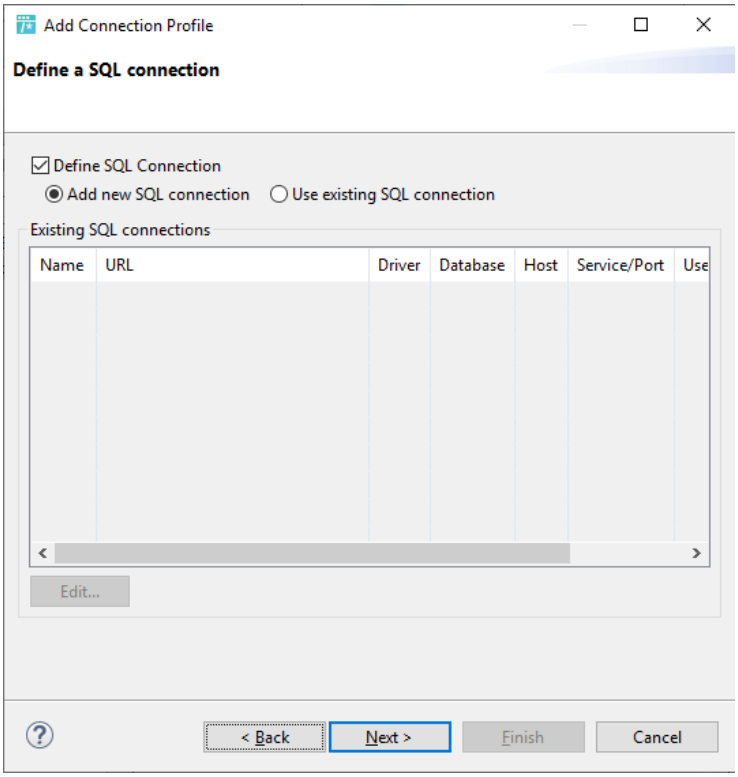


14. The test should complete successfully. If it does not complete successfully, you need to investigate. Is the Database Server running? Is the port number correct for the connection?

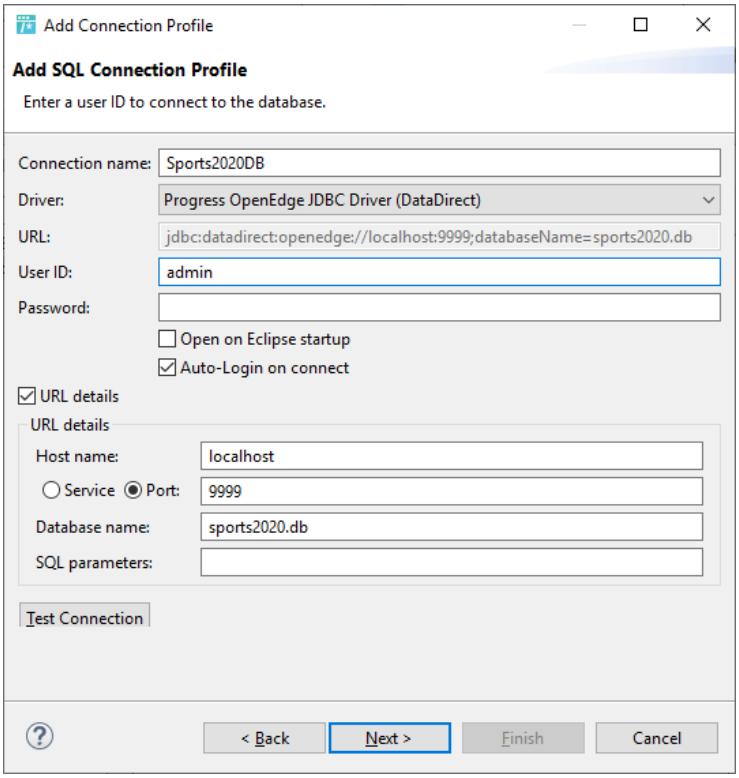
    After the test completes successfully, click **OK**.



15. Back in the Add OpenEdge Database Connection window, click **Next**. The Define a SQL connection window opens.



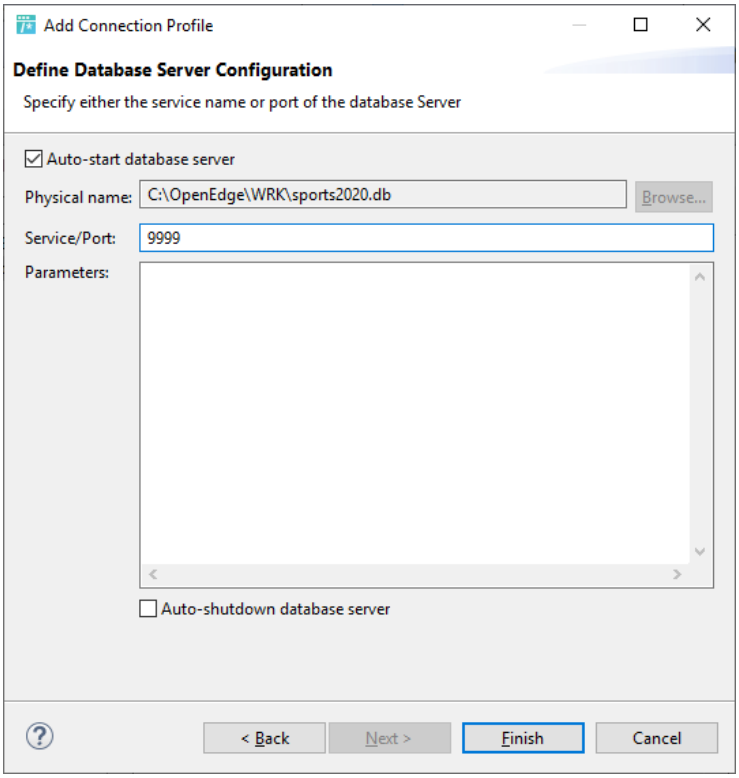
16. Click **Next**. The Add SQL Connection Profile window opens. Here you retain the values that have been automatically set for you by the wizard.



17. Click the **Test Connection** button. Then click **OK** when the connection succeeds. Again, if the connection does not succeed, you must investigate the problem.

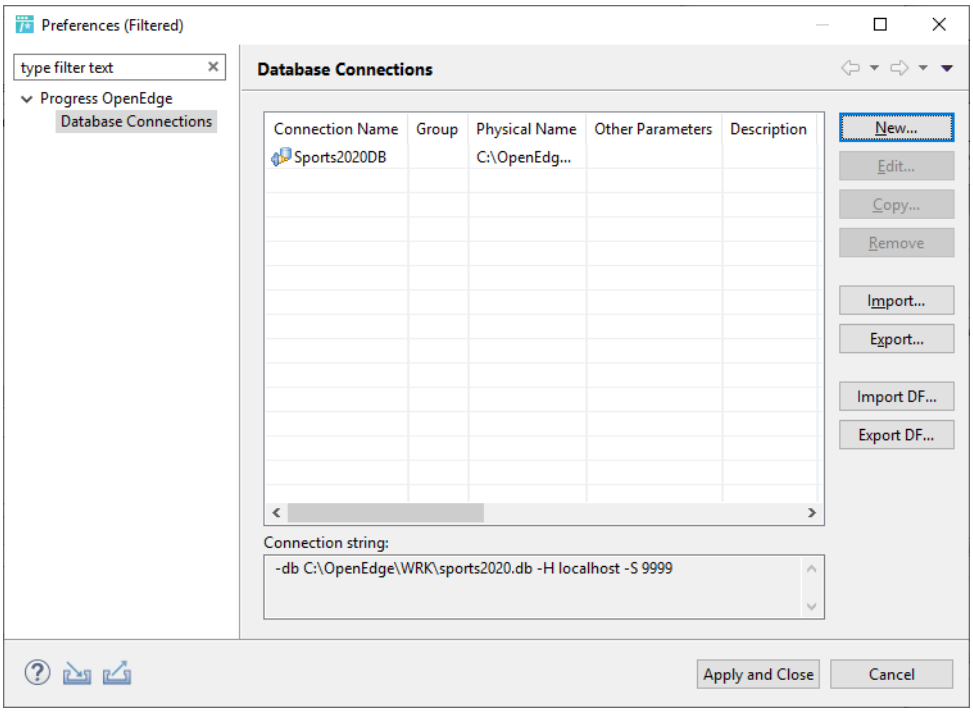
18. Click **Next**. The Define Database Server Configuration window opens. Notice that Auto-start database server is selected. Leave that box selected; Developer Studio will automatically start the Database Server for you. Do not select **Auto-shutdown database server**.

    Click **Finish**.

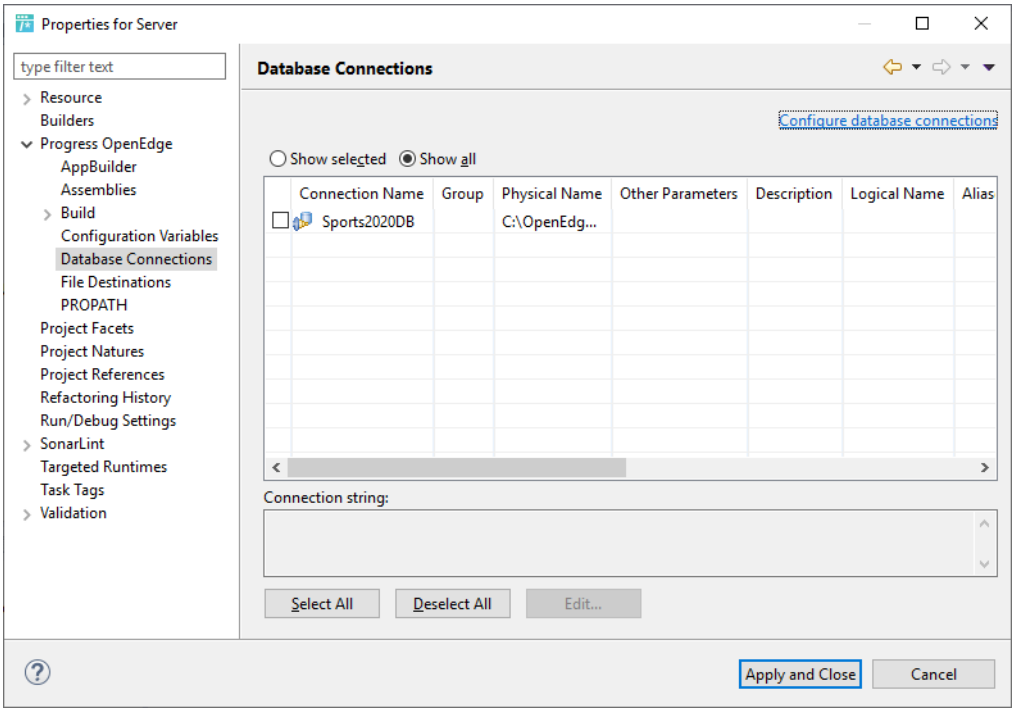


19. After the configuration is completed, you see the connection string in the lower part of the Database Connections window.

    Click **Apply and Close**.

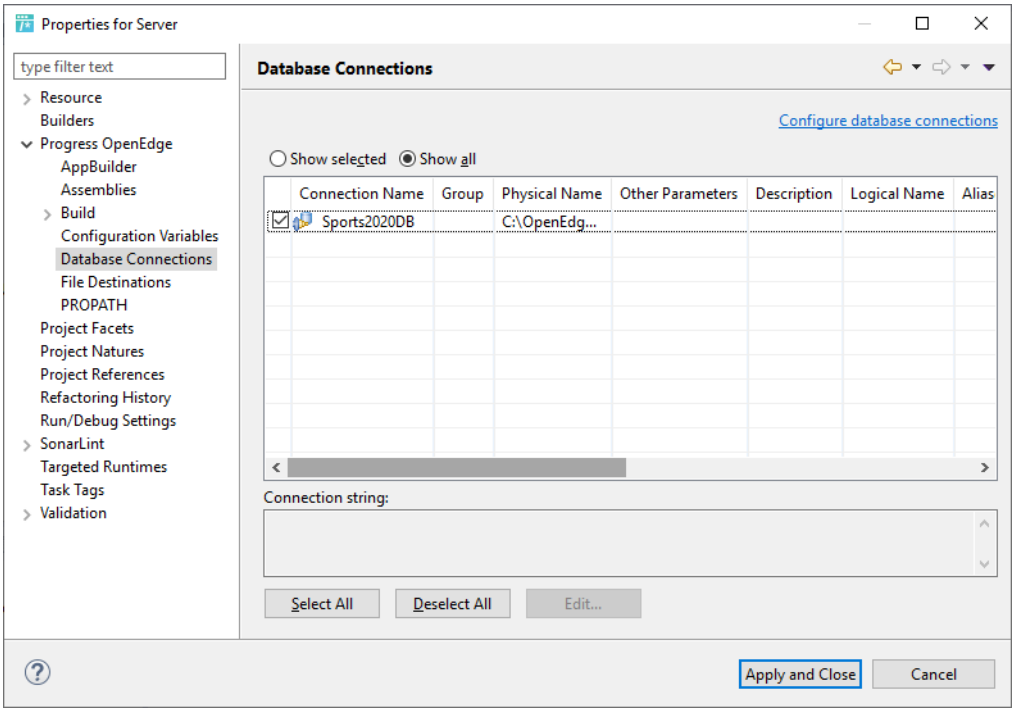


20. You are now back in the Database Connections window of the Server project.

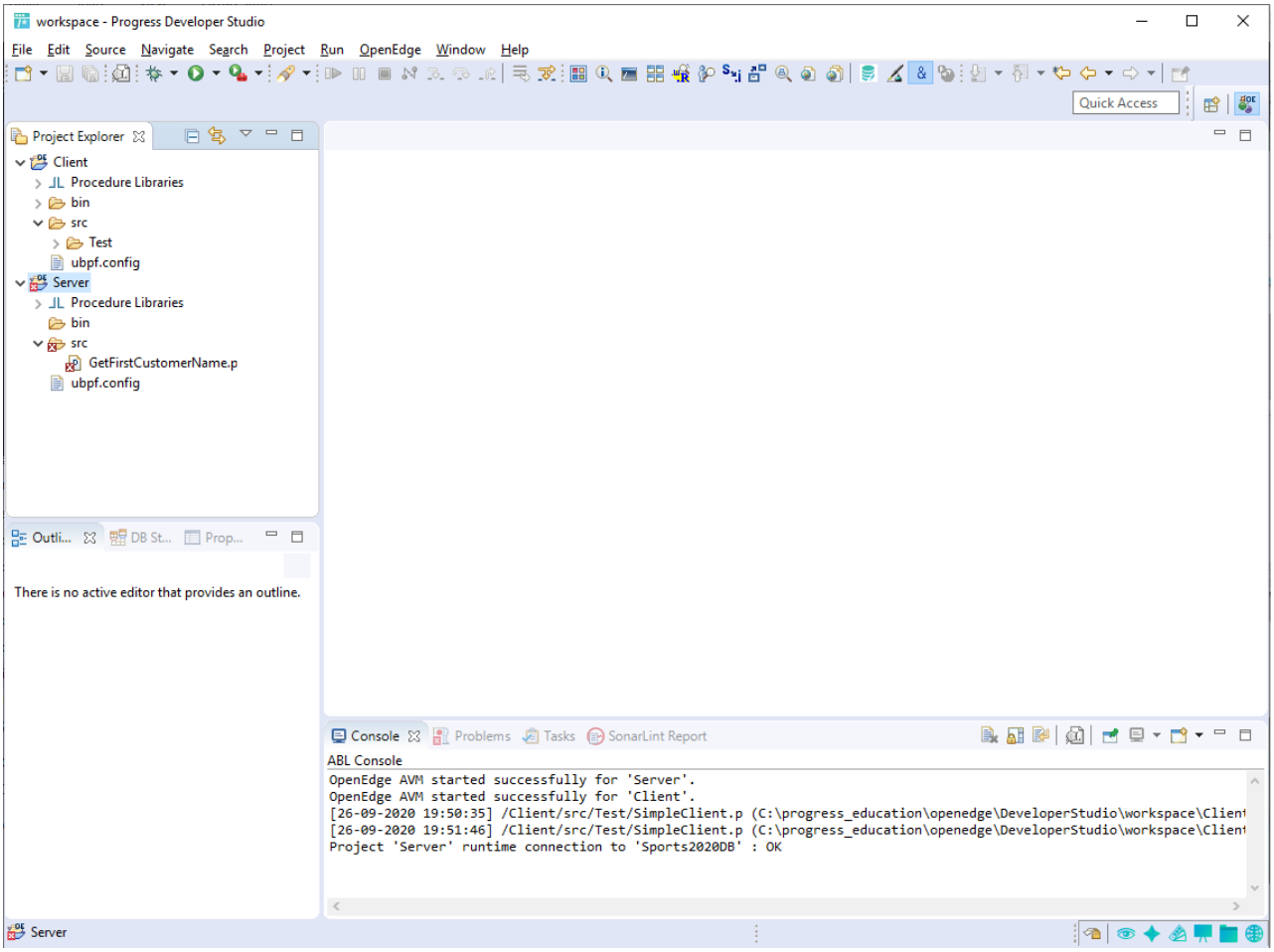


21. Select the connection profile you just configured.

Click **Apply and Close** to complete the configuration.



22. You will notice in the Console that the AVM for the Server project restarts. This is because it must start an AVM that has an active connection to the Database Server.

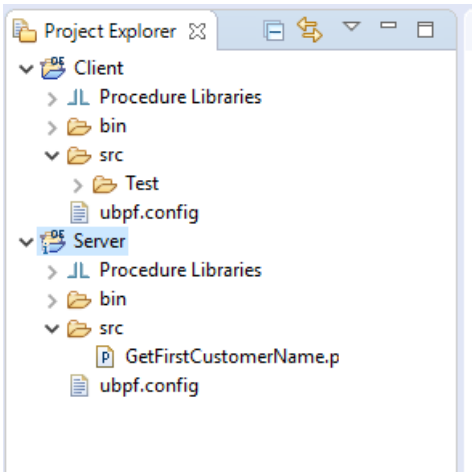


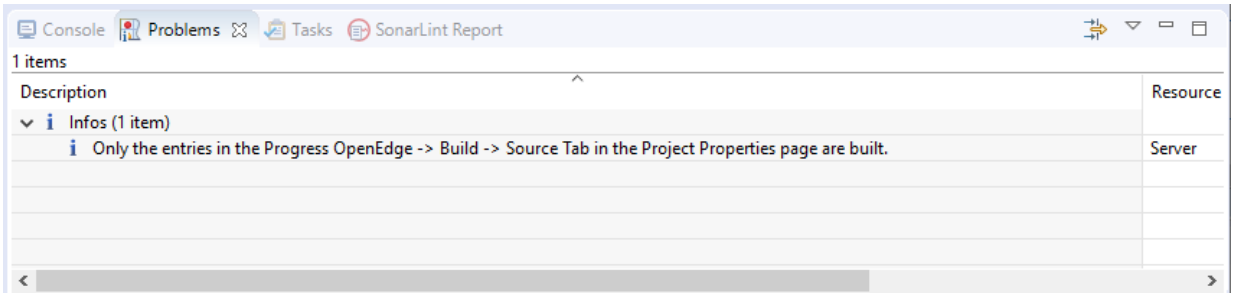
23. Now that the Server project has a connection to the sports2020 database, the code in this project should compile without errors. Clean the workspace to recompile all files:

    a. Select **Project > Clean…**.

    b. Click **OK**.

24. You should now see that all compilation errors have been cleared since the project now has a connection to the database.

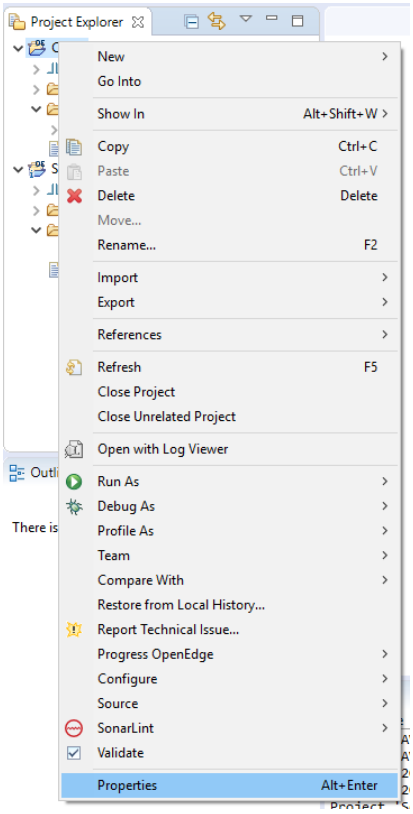




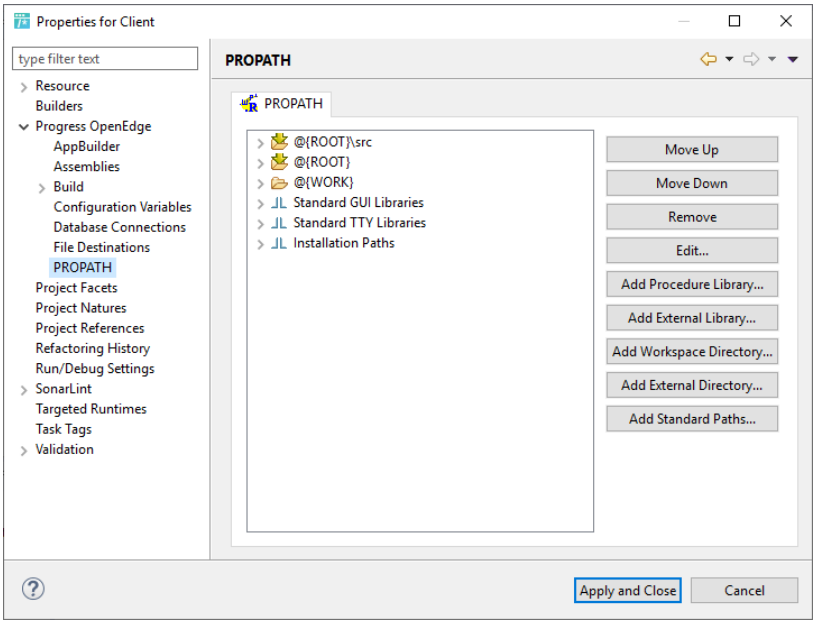
Part 4—Adding PROPATH to the Client project

The ABL code in the Client project references code in the Server project. For the ABL code to run successfully, it must find the code at runtime. Next, you will modify the properties for the Client project to use the Server project bin folder in its PROPATH.

1. In Project Explorer, right-click the **Client** project and then select **Properties**.

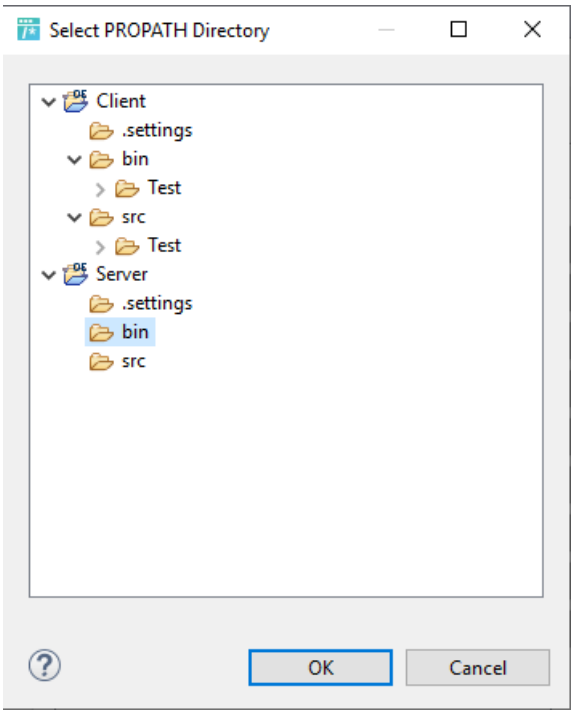


2. Navigate to **Progress OpenEdge > PROPATH**.

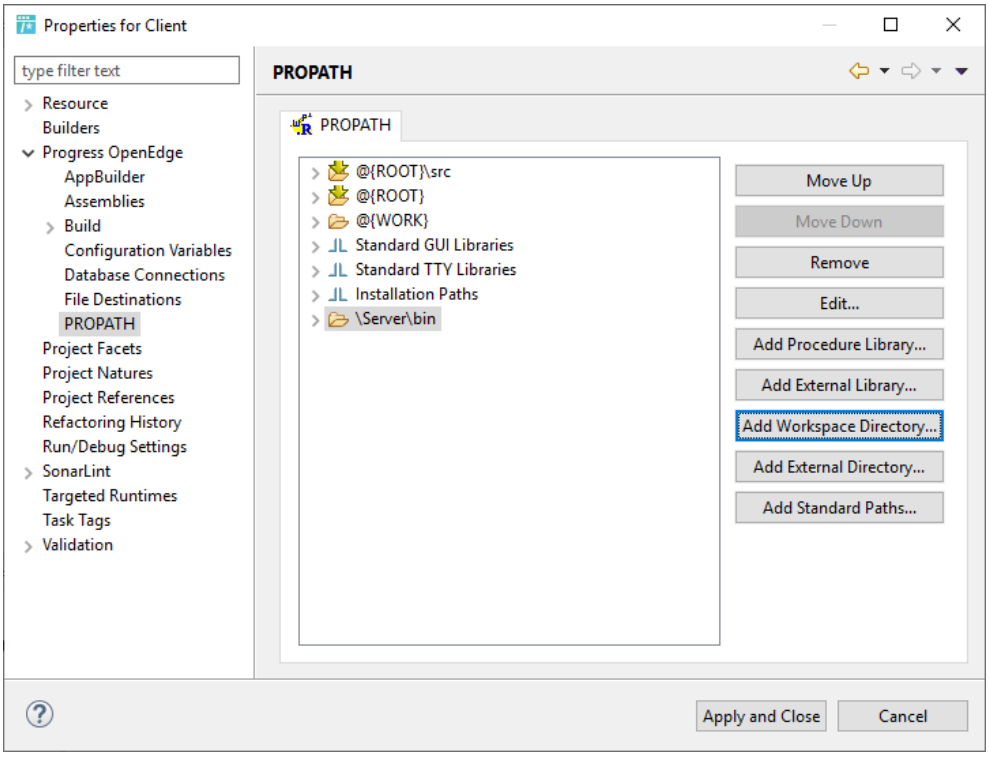


3. Click **Add Workspace Directory…**.

4. In the Select PROPATH Directory window, select the **bin** directory of the **Server** project.



5. Click **OK**.
6. Your PROPATH for the Client Project should now appear as like this:



7. Click **Apply and Close**.

Wrap-up

In this Guided Exercise, you imported an OpenEdge project named Server. Then you created an OpenEdge project named Client and added an ABL procedure to it. Then, you started a Database Server and configured the Server project to connect to the database. Finally, you modified PROPATH for the Client project to use resources in the Server project.