

# Building the program; -> Building the program

**1 Click the Build button to bring up the Project Build panel.**

**2 In the Project Build panel, click the Build button to start the build.**

When you build a program in Project Builder, you are really invoking the **make** utility to create an executable. The **make** utility invokes the compiler and linker using information from the project makefile.

\_MainWindowSyntaxError.eps ->

Before the build begins, Project Builder prompts you to save any unsaved source files, nib files, or the project itself. In this way, Project Builder ensures that you are always building the latest version of the project.

If the build fails because of link errors, look in the bottom browser for more information. The bottom browser shows the exact compiler and linker commands being executed, and it shows all messages produced by these commands. If the Project Build panel appears to only have one browser, drag the split view knob up until you can see the bottom browser.

Before you build, you might want to set specific build options for the project or modify its makefiles. The rest of this chapter describes these tasks.

If the build fails because of link errors, chances are you aren't linking against the correct library or framework. See <sup>a</sup>[Some OPENSTEP Libraries](#) in this chapter. [;BuildingConcepts.rtf](#); [SomeOPENSTEPLibraries](#); ->

**Related Concept:** [;BuildingConcepts.rtf](#); [AllAboutmakeandgnumake](#); All About make and gnumake