Tutorial

- How to Begin the Creation of a New Module
- Some Basic Regulation for the Beginner PHP Developer
- See also:

Note: This page is a work-in-progress. Feedback and suggested improvements are welcome. Please **join the discussion on moodle.org** or use the page comments.

How to Begin the Creation of a New Module

1. Choose a name for your module. Take care: no capital letter and no numbers in the name. In this example it will be choosen: "widget".
2. Use a good text editor to perform a multi-file find/replace to change all occurrences of "newmodule" in each file with the name of your own module.
3. Change the name of the folder containing your newly-created module to "widget".
4. Change the name of file "widget/lang/en_utf8/newmodule.php" to "widget.php".
5. Put your newly-created widget folder in moodle/mod/.
6. Visit the Moodle "Notifications" page (in the Moodle->Site Administration block) to trigger installation of your module.
7. Follow the instructions on the screen.

widget.gif

Some Basic Regulation for the Beginner PHP Developer

You will probably want to use this set of basic commands:

- require
- require_once
- include
- include_once

Take in mind the difference between them.

- The difference between require and include is: **require** produces a fatal E_ERROR level error which halts execution if the specified file cannot be accessed, while **include** produces a warning (E_WARNING) which does not halt execution if the specified file cannot be accessed
- The difference between **require** and **require_once** is: **require_once** checks if the file has been included and does not include it again if it has
- The difference between **include** and **include_once** is: **include_once** checks if the file has been included and does not include it again if it has

Use only xxx to call libraries and use only xxx to include snippets of code written to different files.

Remember:

- **redirect** moves the execution of the code to the target file and you are allowed to call the target file passing it some parameters, for instance: `redirect("./come_modello/edit.php?id=$cm->id");`
- **include** requires the name of the called file to be "clean" (i.e., without "?id=$cm->id"), and the target file is executed as it were written in the place of the calling row.

You need this information to correctly include library files into target files called through **redirect** or **include**.

See also:

- A Step-by-step Guide To Creating Blocks

Source: docs.moodle.org/503/en/Tutorial — MoodleDocs (GPL) · Imported 2026-11-10.

□□□ #3
□ cocobook-sync □□□ 2026-11-10 14:53:33 UTC
□ cocobook-sync □□□ 2026-11-11 00:03:38 UTC