

[Skip to content](#)

# Extensions Operators

`bpy.ops.extensions.dummy_progress()`

Undocumented, consider [contributing](#).

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:1497](#)

`bpy.ops.extensions.package_disable()`

Turn off this extension

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3568](#)

`bpy.ops.extensions.package_enable_not_installed()`

Turn on this extension

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:4040](#)

`bpy.ops.extensions.package_install(*, repo_directory="", repo_index=-1, pkg_id="", enable_on_install=True, url="", do_legacy_replace=False)`

Download and install the extension

**PARAMETERS:**

- **repo\_directory** (*string, (optional, never None)*) – Repo Directory
- **repo\_index** (*int in [-inf, inf], (optional)*) – Repo Index
- **pkg\_id** (*string, (optional, never None)*) – Package ID
- **enable\_on\_install** (*boolean, (optional)*) – Enable on Install, Enable after installing
- **url** (*string, (optional, never None)*) – URL
- **do\_legacy\_replace** (*boolean, (optional)*) – Do Legacy Replace

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:1497](#)

`bpy.ops.extensions.package_install_files(*, filter_glob='*.zip;*.py', directory="", files=None, filepath="", repo="", enable_on_install=True, target="", overwrite=True, url=")`

Install extensions from files into a locally managed repository

**PARAMETERS:**

- **filter\_glob** (*string, (optional, never None)*) – filter\_glob
- **directory** (*string, (optional, never None)*) – Directory
- **files** (*bpy\_prop\_collection of OperatorFileListElement, (optional)*) – files
- **filepath** (*string, (optional, never None)*) – filepath
- **repo** (*enum in [], (optional)*) – User Repository, The user repository to install extensions into
- **enable\_on\_install** (*boolean, (optional)*) – Enable on Install, Enable after installing
- **target** (*enum in [], (optional)*) – Legacy Target Path, Path to install legacy add-on packages to
- **overwrite** (*boolean, (optional)*) – Legacy Overwrite, Remove existing add-ons with the same ID
- **url** (*string, (optional, never None)*) – URL

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:1497](#)

`bpy.ops.extensions.package_install_marked(*, enable_on_install=True)`

Undocumented, consider [contributing](#).

**PARAMETERS:**

**enable\_on\_install** (*boolean, (optional)*) – Enable on Install, Enable after installing

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:1497](#)

bpy.ops.extensions.**package\_mark\_clear**(\*, pkg\_id="", repo\_index=-1)

Undocumented, consider [contributing](#).

**PARAMETERS:**

- **pkg\_id** (*string, (optional, never None)*) – Package ID
- **repo\_index** (*int in [-inf, inf], (optional)*) – Repo Index

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3655](#)

bpy.ops.extensions.**package\_mark\_clear\_all**()

Undocumented, consider [contributing](#).

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3702](#)

bpy.ops.extensions.**package\_mark\_set**(\*, pkg\_id="", repo\_index=-1)

Undocumented, consider [contributing](#).

**PARAMETERS:**

- **pkg\_id** (*string, (optional, never None)*) – Package ID
- **repo\_index** (*int in [-inf, inf], (optional)*) – Repo Index

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3641](#)

bpy.ops.extensions.**package\_mark\_set\_all**()

Undocumented, consider [contributing](#).

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3666](#)

bpy.ops.extensions.**package\_obsolete\_marked**()

Zeroes package versions, useful for development - to test upgrading

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3759](#)

bpy.ops.extensions.**package\_show\_clear**(\*, pkg\_id="", repo\_index=-1)

Undocumented, consider [contributing](#).

**PARAMETERS:**

- **pkg\_id** (*string, (optional, never None)*) – Package ID
- **repo\_index** (*int in [-inf, inf], (optional)*) – Repo Index

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3728](#)

bpy.ops.extensions.**package\_show\_set**(\*, pkg\_id="", repo\_index=-1)

Undocumented, consider [contributing](#).

**PARAMETERS:**

- **pkg\_id** (*string, (optional, never None)*) – Package ID
- **repo\_index** (*int in [-inf, inf], (optional)*) – Repo Index

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3714](#)

`bpy.ops.extensions.package_show_settings(*, pkg_id="", repo_index=-1)`

Undocumented, consider [contributing](#).

**PARAMETERS:**

- **pkg\_id** (*string, (optional, never None)*) – Package ID
- **repo\_index** (*int in [-inf, inf], (optional)*) – Repo Index

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3742](#)

`bpy.ops.extensions.package_theme_disable(*, pkg_id="", repo_index=-1)`

Turn off this theme

**PARAMETERS:**

- **pkg\_id** (*string, (optional, never None)*) – Package ID
- **repo\_index** (*int in [-inf, inf], (optional)*) – Repo Index

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3596](#)

`bpy.ops.extensions.package_theme_enable(*, pkg_id="", repo_index=-1)`

Turn off this theme

**PARAMETERS:**

- **pkg\_id** (*string, (optional, never None)*) – Package ID
- **repo\_index** (*int in [-inf, inf], (optional)*) – Repo Index

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3581](#)

`bpy.ops.extensions.package_uninstall(*, repo_directory="", repo_index=-1, pkg_id="")`

Disable and uninstall the extension

**PARAMETERS:**

- **repo\_directory** (*string, (optional, never None)*) – Repo Directory
- **repo\_index** (*int in [-inf, inf], (optional)*) – Repo Index
- **pkg\_id** (*string, (optional, never None)*) – Package ID

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:1497](#)

`bpy.ops.extensions.package_uninstall_marked()`

Undocumented, consider [contributing](#).

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:1497](#)

`bpy.ops.extensions.package_uninstall_system()`

Undocumented, consider [contributing](#).

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3559](#)

`bpy.ops.extensions.package_upgrade_all(*, use_active_only=False)`

Upgrade all the extensions to their latest version for all the remote repositories

**PARAMETERS:**

**use\_active\_only** (*boolean, (optional)*) – Active Only, Only sync the active repository

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:1497](#)

`bpy.ops.extensions.repo_enable_from_drop(*, repo_index=1)`

Undocumented, consider [contributing](#).

**PARAMETERS:**

**repo\_index** (*int in [-inf, inf], (optional)*) – Repo Index

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:1838](#)

`bpy.ops.extensions.repo_lock_all()`

Lock repositories - to test locking

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3828](#)

`bpy.ops.extensions.repo_refresh_all()`

Scan extension & legacy add-ons for changes to modules & meta-data (similar to restarting). Any issues are reported as warnings

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:1759](#)

`bpy.ops.extensions.repo_sync(*, repo_directory="", repo_index=1)`

Undocumented, consider [contributing](#).

**PARAMETERS:**

- **repo\_directory** (*string, (optional, never None)*) – Repo Directory
- **repo\_index** (*int in [-inf, inf], (optional)*) – Repo Index

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:1497](#)

`bpy.ops.extensions.repo_sync_all(*, use_active_only=False)`

Refresh the list of extensions for all the remote repositories

**PARAMETERS:**

**use\_active\_only** (*boolean, (optional)*) – Active Only, Only sync the active repository

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:1497](#)

`bpy.ops.extensions.repo_unlock()`

Remove the repository file-system lock

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:1925](#)

`bpy.ops.extensions.repo_unlock_all()`

Unlock repositories - to test unlocking

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3854](#)

`bpy.ops.extensions.status_clear()`

Undocumented, consider [contributing](#).

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3627](#)

`bpy.ops.extensions.status_clear_errors()`

Undocumented, consider [contributing](#).

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3616](#)

`bpy.ops.extensions.userpref_allow_online()`

Allow internet access. Blender may access configured online extension repositories. Installed third party add-ons may access the internet for their own functionality

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3976](#)

`bpy.ops.extensions.userpref_allow_online_popup()`

Allow internet access. Blender may access configured online extension repositories. Installed third party add-ons may access the internet for their own functionality

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3990](#)

`bpy.ops.extensions.userpref_show_for_update()`

Open extensions preferences

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3919](#)

`bpy.ops.extensions.userpref_show_online()`

Show system preferences “Network” panel to allow online access

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3956](#)

`bpy.ops.extensions.userpref_tags_set(*, value=False, data_path='')`

Set the value of all tags

**PARAMETERS:**

- **value** (*boolean, (optional)*) – Value, Enable or disable all tags
- **data\_path** (*string, (optional, never None)*) – Data Path

**FILE:**

[addons\\_core/bl\\_pkg/bl\\_extension\\_ops.py:3888](#)