Skip to content

File Operators

bpy.ops.file.**autopack_toggle()**

Automatically pack all external files into the .blend file

bpy.ops.file.**bookmark_add()**

Add a bookmark for the selected/active directory

bpy.ops.file.**bookmark_cleanup()**

Delete all invalid bookmarks

bpy.ops.file.**bookmark_delete(*, index=-1)**

Delete selected bookmark

PARAMETERS:

index (*int in [-1, 20000], (optional)*) – Index

bpy.ops.file.**bookmark_move(*, direction='TOP')**

Move the active bookmark up/down in the list

PARAMETERS:

direction (*enum in ['TOP', 'UP', 'DOWN', 'BOTTOM'], (optional)*) –

Direction, Direction to move the active bookmark towards

- `TOP` Top – Top of the list.
- `UP` Up.
- `DOWN` Down.
- `BOTTOM` Bottom – Bottom of the list.

bpy.ops.file.**cancel()**

Cancel file operation

bpy.ops.file.**delete()**

Move selected files to the trash or recycle bin

bpy.ops.file.**directory_new(*, directory='', open=False, confirm=True)**

Create a new directory

PARAMETERS:

- **directory** (*string, (optional, never None)*) – Directory, Name of new directory
- **open** (*boolean, (optional)*) – Open, Open new directory
- **confirm** (*boolean, (optional)*) – Confirm, Prompt for confirmation

bpy.ops.file.**edit_directory_path()**

Start editing directory field

bpy.ops.file.**execute()**

Execute selected file

bpy.ops.file.**external_operation(*, filepath='', operation='OPEN')**

Perform external operation on a file or folder

PARAMETERS:

- **filepath** (*string, (optional, never None)*) – File or folder path

- **filepath** (*string, (optional, never None)*) – File or folder path
- **operation** (*enum in ['OPEN', 'FOLDER_OPEN', 'EDIT', 'NEW', 'FIND', 'SHOW', 'PLAY', 'BROWSE', 'PREVIEW', 'PRINT', 'INSTALL 'RUNAS', 'PROPERTIES', 'FOLDER_FIND', 'CMD'], (optional)*) –

  Operation, Operation to perform on the file or path

  - `OPEN` Open – Open the file.
  - `FOLDER_OPEN` Open Folder – Open the folder.
  - `EDIT` Edit – Edit the file.
  - `NEW` New – Create a new file of this type.
  - `FIND` Find File – Search for files of this type.
  - `SHOW` Show – Show this file.
  - `PLAY` Play – Play this file.
  - `BROWSE` Browse – Browse this file.
  - `PREVIEW` Preview – Preview this file.
  - `PRINT` Print – Print this file.
  - `INSTALL` Install – Install this file.
  - `RUNAS` Run As User – Run as specific user.
  - `PROPERTIES` Properties – Show OS Properties for this item.
  - `FOLDER_FIND` Find in Folder – Search for items in this folder.
  - `CMD` Command Prompt Here – Open a command prompt here.

bpy.ops.file.**filenum(*, increment=1)**

Increment number in filename

PARAMETERS:

increment (*int in [-100, 100], (optional)*) – Increment

bpy.ops.file.**filepath_drop(*, filepath='Path')**

Undocumented, consider contributing.

bpy.ops.file.**find_missing_files(*, find_all=False, directory='', hide_props_region=True, check_existing=False, filter_blender=False, filter_backup=False, filter_image=False, filter_movie=False, filter_python=False, filter_font=False, filter_sound=False, filter_text=False, filter_archive=False, filter_btx=False, filter_collada=False, filter_alembic=False, filter_usd=False, filter_obj=False, filter_volume=False, filter_folder=False, filter_blenlib=False, filemode=9, display_type='DEFAULT', sort_method='')**

Try to find missing external files

PARAMETERS:

- **find_all** (*boolean, (optional)*) – Find All, Find all files in the search path (not just missing)
- **directory** (*string, (optional, never None)*) – Directory, Directory of the file
- **hide_props_region** (*boolean, (optional)*) – Hide Operator Properties, Collapse the region displaying the operator settings
- **check_existing** (*boolean, (optional)*) – Check Existing, Check and warn on overwriting existing files
- **filter_blender** (*boolean, (optional)*) – Filter .blend files
- **filter_backup** (*boolean, (optional)*) – Filter .blend files
- **filter_image** (*boolean, (optional)*) – Filter image files
- **filter_movie** (*boolean, (optional)*) – Filter movie files
- **filter_python** (*boolean, (optional)*) – Filter Python files
- **filter_font** (*boolean, (optional)*) – Filter font files
- **filter_sound** (*boolean, (optional)*) – Filter sound files
- **filter_text** (*boolean, (optional)*) – Filter text files
- **filter_archive** (*boolean, (optional)*) – Filter archive files
- **filter_btx** (*boolean, (optional)*) – Filter btx files
- **filter_collada** (*boolean, (optional)*) – Filter COLLADA files

- **filter_alembic** (*boolean, (optional)*) – Filter Alembic files
- **filter_usd** (*boolean, (optional)*) – Filter USD files
- **filter_obj** (*boolean, (optional)*) – Filter OBJ files
- **filter_volume** (*boolean, (optional)*) – Filter OpenVDB volume files
- **filter_folder** (*boolean, (optional)*) – Filter folders
- **filter_blenlib** (*boolean, (optional)*) – Filter Blender IDs
- **filemode** (*int in [1, 9], (optional)*) – File Browser Mode, The setting for the file browser mode to load a .blend file, a library or a special fil
- **display_type** (*enum in ['DEFAULT', 'LIST_VERTICAL', 'LIST_HORIZONTAL', 'THUMBNAIL'], (optional)*) – Display Type
    - `DEFAULT` Default – Automatically determine display type for files.
    - `LIST_VERTICAL` Short List – Display files as short list.
    - `LIST_HORIZONTAL` Long List – Display files as a detailed list.
    - `THUMBNAIL` Thumbnails – Display files as thumbnails.
- **sort_method** (*enum in [], (optional)*) – File sorting mode

bpy.ops.file.**hidedot()**

Toggle hide hidden dot files

bpy.ops.file.**highlight()**

Highlight selected file(s)

bpy.ops.file.**make_paths_absolute()**

Make all paths to external files absolute

bpy.ops.file.**make_paths_relative()**

Make all paths to external files relative to current .blend

bpy.ops.file.**mouse_execute()**

Perform the current execute action for the file under the cursor (e.g. open the file)

bpy.ops.file.**next()**

Move to next folder

bpy.ops.file.**pack_all()**

Pack all used external files into this .blend

bpy.ops.file.**pack_libraries()**

Store all data-blocks linked from other .blend files in the current .blend file. Library references are preserved so the linked data-blocks can be unpacked again

bpy.ops.file.**parent()**

Move to parent directory

bpy.ops.file.**previous()**

Move to previous folder

bpy.ops.file.**refresh()**

Refresh the file list

bpy.ops.file.**rename()**

Rename file or file directory

bpy.ops.file.**report_missing_files()**

bpy.ops.file.**report_missing_files()**

Report all missing external files

bpy.ops.file.**reset_recent()**

Reset recent files

bpy.ops.file.**select(*, wait_to_deselect_others=False, mouse_x=0, mouse_y=0, extend=False, fill=False, open=True, deselect_all=False, only_activate_if_selected=False, pass_through=False)**

Handle mouse clicks to select and activate items

PARAMETERS:

- **wait_to_deselect_others** (*boolean, (optional)*) – Wait to Deselect Others
- **mouse_x** (*int in [-inf, inf], (optional)*) – Mouse X
- **mouse_y** (*int in [-inf, inf], (optional)*) – Mouse Y
- **extend** (*boolean, (optional)*) – Extend, Extend selection instead of deselecting everything first
- **fill** (*boolean, (optional)*) – Fill, Select everything beginning with the last selection
- **open** (*boolean, (optional)*) – Open, Open a directory when selecting it
- **deselect_all** (*boolean, (optional)*) – Deselect On Nothing, Deselect all when nothing under the cursor
- **only_activate_if_selected** (*boolean, (optional)*) – Only Activate if Selected, Do not change selection if the item under the cursor is already selected, only activate it
- **pass_through** (*boolean, (optional)*) – Pass Through, Even on successful execution, pass the event on so other operators can execute on it a well

bpy.ops.file.**select_all(*, action='TOGGLE')**

Select or deselect all files

PARAMETERS:

action (*enum in ['TOGGLE', 'SELECT', 'DESELECT', 'INVERT'], (optional)*) –

Action, Selection action to execute

- `TOGGLE` Toggle – Toggle selection for all elements.
- `SELECT` Select – Select all elements.
- `DESELECT` Deselect – Deselect all elements.
- `INVERT` Invert – Invert selection of all elements.

bpy.ops.file.**select_bookmark(*, dir='')**

Select a bookmarked directory

PARAMETERS:

dir (*string, (optional, never None)*) – Directory

bpy.ops.file.**select_box(*, xmin=0, xmax=0, ymin=0, ymax=0, wait_for_input=True, mode='SET')**

Activate/select the file(s) contained in the border

PARAMETERS:

- **xmin** (*int in [-inf, inf], (optional)*) – X Min
- **xmax** (*int in [-inf, inf], (optional)*) – X Max
- **ymin** (*int in [-inf, inf], (optional)*) – Y Min
- **ymax** (*int in [-inf, inf], (optional)*) – Y Max
- **wait_for_input** (*boolean, (optional)*) – Wait for Input
- **mode** (*enum in ['SET', 'ADD', 'SUB'], (optional)*) –

  Mode

  - `SET` Set – Set a new selection.
  - `ADD` Extend – Extend existing selection.

- `ADD` Extend – Extend existing selection.
- `SUB` Subtract – Subtract existing selection.

bpy.ops.file.**select_walk(*, direction='UP', extend=False, fill=False)**

Select/Deselect files by walking through them

PARAMETERS:

- **direction** (*enum in ['UP', 'DOWN', 'LEFT', 'RIGHT'], (optional)*) – Walk Direction, Select/Deselect element in this direction
- **extend** (*boolean, (optional)*) – Extend, Extend selection instead of deselecting everything first
- **fill** (*boolean, (optional)*) – Fill, Select everything beginning with the last selection

bpy.ops.file.**smoothscroll()**

Smooth scroll to make editable file visible

bpy.ops.file.**sort_column_ui_context()**

Change sorting to use column under cursor

bpy.ops.file.**start_filter()**

Start entering filter text

bpy.ops.file.**unpack_all(*, method='USE_LOCAL')**

Unpack all files packed into this .blend to external ones

PARAMETERS:

method (*enum in ['USE_LOCAL', 'WRITE_LOCAL', 'USE_ORIGINAL', 'WRITE_ORIGINAL', 'KEEP', 'REMOVE'], (optional)*) – Method, How to unpack

bpy.ops.file.**unpack_item(*, method='USE_LOCAL', id_name='', id_type=19785)**

Unpack this file to an external file

PARAMETERS:

- **method** (*enum in ['USE_LOCAL', 'WRITE_LOCAL', 'USE_ORIGINAL', 'WRITE_ORIGINAL'], (optional)*) – Method, How to unpack
- **id_name** (*string, (optional, never None)*) – ID Name, Name of ID block to unpack
- **id_type** (*int in [0, inf], (optional)*) – ID Type, Identifier type of ID block

bpy.ops.file.**unpack_libraries()**

Restore all packed linked data-blocks to their original locations

bpy.ops.file.**view_selected()**

Scroll the selected files into view