

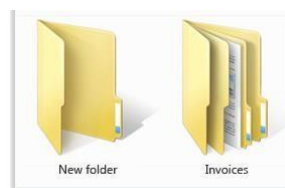
Working with files and folders

A **file** is an item that contains information, for example, text or images or music. When opened, a file can look very much like a text document or a picture that you might find on someone's desk or in a filing cabinet. On your computer, files are represented with **icons**, this makes it easy to recognize a type of file by looking at its icon.



There are hidden files, Although you usually can't see a hidden file, it's an ordinary file in all other respects. You can choose whether a file is hidden or visible by changing its attribute feature of its properties to become **hidden**. If a file is hidden and you want to display it, you'll need to show all hidden files to see it. You can hide rarely used files to reduce clutter, but remember that hidden files still take up space on your hard disk. Although you can hide confidential files so other people can't see them, you shouldn't rely on hidden files as your only means of security or privacy

A **folder** is a container you can use to store files in. If you had thousands of paper files on your desk, it would be nearly impossible to find any particular file when you needed it. That's why people often store paper files in folders inside a filing cabinet. On your computer, folders work the same way. Here are some typical folder icons:



An empty folder (left), a folder containing files (right)

Folders can also store other folders, a folder within a folder is usually called a **subfolder**. You can create any number of subfolders, and each can hold any number of files and additional subfolders.

From the [Computer folder](#), you can access locations such as hard disks, CD or DVD drives, and removable media. You can also access other devices that might be connected to your computer, such as external hard drives and USB flash drives. A common reason to open the Computer folder is to view the available space on your hard disks and removable media. If you right-click an item in the Computer folder, you can perform tasks such as ejecting a CD or DVD, viewing the properties of a hard disk, and formatting a disk.

[Using libraries to access files and folders](#)

[Library](#) is a collection of items, such as files and folders that are assembled from various locations, the locations might be on a computer, an external hard drive, etc. You can use *libraries* to access your files and folders, and arrange them in different ways. In libraries, you can include folders from different locations so that you can see the files in one place. This is a good way to keep your files organized so you don't have to search multiple locations to find what you're looking for. In some ways, a [library](#) is similar to a [folder](#). For example, when you open a library, you'll see one or more files. However, you can't store files or folders in a library. This is a subtle, but important, difference between **file and library**.



There are four default libraries:

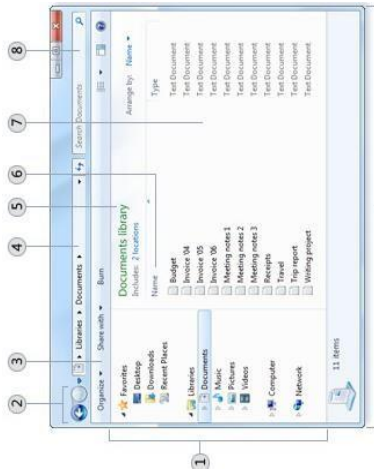
- **Documents library**: Use this library to organize and arrange word-processing documents, presentations, and other text-related files. By default, files that you move, copy, or save to the Documents library are stored in the My Documents folder.
- **Pictures library**: Use this library to organize and arrange your digital pictures, whether you get them from your camera, scanner, or in e-mail from other people. By default, files that you move, copy, or save to the Pictures library are stored in the My Pictures folder.
- **Music library**: Use this library to organize and arrange your digital music, by default, files that you move, copy, or save to the Music library are stored in the My Music folder.
- **Videos library**: Use this library to organize and arrange your videos, such as clips from your digital camera, or video files that you download from the Internet. By default, files that you move, copy, or save to the Videos library are stored in the My Videos folder.

Some things you can do with libraries:

- **Create a new library**. There are four default libraries (Documents, Music, Pictures, and Videos), but you can create new libraries for other collections.
- **Arrange items** by folder, date, and other properties. Items in a library can be arranged in different ways using the Arrange by menu, located in the library pane (above the file list) in any open library. For example, you can arrange your Music library by Artist to quickly find a song by a particular artist.
- **Include or remove a folder**. Libraries gather content from included folders, or library locations.
- **Change the default save location**. The default save location determines where an item is stored when it's copied, moved, or saved to the library.

Parts of Windows

When you open a folder or library, you see it in a [window](#). The various parts of this window are designed to help you navigate around Windows or work with files, folders, and libraries more easily. Here's a typical window and each of its parts:



- 1 Navigation pane
- 2 Back and Forward buttons
- 3 Toolbar
- 4 Address bar
- 5 Library pane
- 6 Column headings
- 7 File list
- 8 The search box
- 9 Details pane

What it's useful for

Navigation pane	Use the navigation pane to access libraries, folders, saved searches , and even entire hard disks . Use the Favorites section to open your most commonly used folders and searches; use the Libraries section to access your libraries. You can also use the Computer folder to browse folders and subfolders. For more information, see Working with the navigation pane .
Back and Forward buttons	Use the Back button and the Forward button to navigate to other folders or libraries you've already opened without closing the current window. These buttons work together with the address bar; after you use the address bar to change folders, for example, you can use the Back button to return to the previous folder.
Toolbar	Use the toolbar to perform common tasks, such as changing the appearance of your files and folders, burning files to a CD, or starting a digital picture slide show. The toolbar's buttons change to show only the tasks that are relevant. For example, if you click a picture file, the toolbar shows different buttons than it would if you clicked a music file.
Address bar	Use the address bar to navigate to a different folder or library or to go back to a previous one. For more information, see Navigate using the address bar .
Library pane	The library pane appears only when you are in a library (such as the Documents library). Use the library pane to customize the library or to arrange the files by different properties. For more information, see Working with libraries .
Column headings	Use the column headings to change how the files in the file list are organized. For example, you can click the left side of a column heading to change the order the files and folders are displayed in, or you can click the right side to filter the files in different ways. (Note that column headings are available only in Details view. To learn how to switch to Details view, see Viewing and arranging files and folders later in this topic.)
File list	This is where the contents of the current folder or library are displayed. If you type in the search box to find a file, only the files that match your current view (including files in subfolders) will appear.
The search box	Type a word or phrase in the search box to look for an item in the current folder or library. The search begins as soon as you begin typing—so if you type "B," for example, all the files with names starting with the letter B will appear in the file list. For more information, see Find a file or folder .
Details pane	Use the details pane to see the most common properties associated with the selected file. File properties are information about a file, such as the author, the date you last changed the file, and any descriptive tags you might have added to the file. For more information, see Add tags and other properties to files .
Preview pane	Use the preview pane to see the contents of most files. If you select an e-mail message, text file, or picture, for example, you can see its contents without opening it in a program. If you don't see the preview pane, click the Preview pane button in the toolbar to turn it on.

File Concept

Since main memory is usually too small to contain all the data and programs permanently, the computer system must provide secondary storage to back up main memory. The file system provides the mechanism for on-line storage of and access to both data and programs residing on the disks.

The files are mapped by the operating system onto physical devices. The devices that attach to a computer vary in many aspects. Some devices transfer a character or a block of characters at a time. Some can be accessed only sequentially, others randomly. Some transfer data synchronously, others asynchronously. They can be read-only or read-write. They vary greatly in speed. In many ways, they are also the slowest major component of the computer. Because of all this device variation, the operating system needs to provide a wide range of functionality to applications, to allow them to control all aspects of the devices.

Computers can store information on various storage media, such as magnetic disks, magnetic tapes, and optical disks. So that the computer system will be convenient to use, the operating system provides a uniform logical view of information storage. The operating system abstracts from the physical properties of its storage devices to define a logical storage unit, the file. Files are mapped by the operating system onto physical devices. These storage devices are usually nonvolatile, so the contents are persistent through power failures and system reboots. A file is a named collection of related information that is recorded on secondary storage. From a user's perspective, a file is the smallest allotment of logical secondary storage; that is, data cannot be written to secondary storage unless they are within a file.

Commonly, files represent programs (both source and object forms) and data. Data files may be numeric, alphabetic, alphanumeric, or binary, files may be free form, such as text files. In general, a file is a sequence of bits, bytes, lines, or records.

Types of Files

Windows O.S. recognizes many different types of files. The type depends on in which application the file has been created. Below are some of the examples of common file types identified by their extension or by the icon.

Many different types of information may be stored in a file-source programs, object programs, executable programs, numeric data, text, payroll records, graphic images, sound recordings, and so on. A file has a certain defined which depends on its type. A text file is a sequence of characters organized into lines (and possibly pages). A source file is a sequence of subroutines and functions, each of which is further organized as declarations followed by executable statements. An object file is a sequence of bytes organized into blocks understandable by the system's linker. An executable file is a series of code sections that the loader can bring into memory and execute.

File Extension	File Type
.txt	A text file
.bmp	Bit Map File
.docx	MS-Word file
.xlsx	MS-Excel file
.Zip or .rar	A compressed file
.pdf	A portable Document Format file
.htm	A web page file

File Attributes

A file is named, for the convenience of its human users, and is referred to by its name. A name is usually a string of characters, such as *example.cpp*. Some systems differentiate between uppercase and lowercase characters in names, whereas other systems do not. When a file is named, it becomes independent of the process, the user, and even the system that created it. For instance, one user might create the file *example.cpp*, and another user might edit that file by specifying its name.

The file's owner might write the file to a flash memory, send it in an e-mail, or copy it across a network, and it could still be called *example.cpp* on the destination system.

A file's attributes vary from one operating system to another but typically consist of these:

Name. The symbolic file name is the only information kept in human readable form. **Identifier.** This unique tag, usually a number, identifies the file within the file system; it is the non-human-readable name for the file.

Type. This information is needed for systems that support different types of files. **Location.** This information is a pointer to a device and to the location of the file on that device.

Size. The current size of the file (in bytes, words, or blocks) and possibly the maximum allowed size are included in this attribute.

Protection. Access-control information determines who can do reading, writing, executing, and so on.

Time, date, and user identification. This information may be kept for creation, last modification, and last use. These data can be useful for protection, security, and usage monitoring.

File Operations

To define a file properly, we need to consider the operations that can be performed on files. The operating system can provide system calls to create, write, read, reposition, delete, and truncate files. Let's examine **what the operating system must do to perform** each of these six basic file operations.

Creating a file. Two steps are necessary to create a file. First, space in the file system must be found for the file. Second, an entry for the new file must be made in the directory.

Writing a file. To write a file, we make a system call specifying both the name of the file and the information to be written to the file. Given the name of the file, the system searches the directory to find the file's location.

The system must keep a *write* pointer to the location in the file where the next write is to take place. The write pointer must be updated whenever a write occurs.

Reading a file. To read from a file, we use a system call that specifies the name of the file and where (in memory) the next block of the file should be put. Again, the directory is searched for the associated entry, and the system needs to keep a *read* pointer to the location in the file where the next read is to take place. Once the read has taken place, the read pointer is updated. Because a process is usually either reading from or writing to a file, the current operation location can be kept as a per-process . Both the read and write operations use this same pointer, saving space and reducing system complexity.

Repositioning within a file. The directory is searched for the appropriate entry, and the current-file-position pointer is repositioned to a given value. Repositioning within a file need not involve any actual I/O. This file operation is also known as a **file seek**.

Deleting a file. To delete a file, we search the directory for the named file. Having found the associated directory entry, we release all file space, so that it can be reused by other files, and erase the directory entry.

Truncating a file. The user may want to erase the contents of a file but keep its attributes. Rather than forcing the user to delete the file and then recreate it, this function allows all attributes to remain unchanged –except for file length-but lets the file be reset to length zero and its file space released.

These six basic operations comprise the minimal set of required file operations. Other common operations include ***appending*** new information to the end of an existing file , ***renaming*** an existing file and ***execute*** (Load the file into memory and execute it). These primitive operations can then be combined to perform other file operations. For instance, we can create a ***copy*** of a file, or copy the file to another I/O device, such as a printer or a display, by creating a new file and then reading from the old and writing to the new.