Digital modeling is based on the XYZ coordinate system, with the color red assigned to X, green to Y, and blue to Z (think of the rhyme RGB=XYZ). In front view X is always side to side, but Z may be up and down or into and out of the screen depending on the program. In Blender it’s always up and down, with Y representing movement into and out of the screen when seen from the front view.

What sometimes causes confusion is that not only is there a world coordinate system (often shortened to WCS) but also every object in that world has one, known as its local coordinate system (LCS). These are identical except for location when the object is created but differ as the object is later moved and rotated.

This is a new cube in Blender. Blender’s default WCS tripod is in the lower left corner while the cube’s control widget is centered on its pivot point (the orange dot). You can see they are aligned (bear in mind this is a perspective view). Now let’s rotate the cube 45° around the Z axis by typing R-Z-45-Enter. The
cube’s widget is still aligned to the WCS. This is because by default the movement of the cube is based on the WCS. We can see this by looking at the control bar at the bottom of the 3D Window:

The word Global means that any selected object(s) will move and rotate according to the WCS. If we click on that box and select Local, however, the control widget now shows the cube’s local axes:

By holding down the left mouse button on the head of any of the control widget arrows we can now move and rotate the cube along or around its local axes. Another way to do this, without having to change the automatic selection mode using the pop-up menu, is to add an extra axis letter to the typed commands. So moving along the world X axis is G-X, and moving along the selected object’s local X axis is G-X-X. Rotating around an axis parallel to the world Y axis and through the object’s orange pivot point is R-Y; rotating around the local Y axis (which is always through the object’s pivot point) is R-Y-Y.

Movement, rotation, and scaling are called transform operations because they don’t change the relative locations of the object’s vertices (though scaling in one or two dimensions will change the proportions of the object itself). Blender uses the key shortcuts G for movement, R for rotation, and S for scaling.

You can also activate one or more (with Shift-click) the widgets in the 3D Window button bar and then drag them in the 3D window. Generally only the default Move arrows are needed, though, and it keeps the screen simpler.
Or use the pop-up menu options in Object-Transform. Though this is a slow way it does show the wide variety of commands (not all of which have key shortcuts) available through the menus.

And finally you can manually enter changes in the Transform panel at the top of the right fly-in.

Let’s look at the keyboard shortcuts first. We’ve seen that hitting the axis key twice changes that axis from world to local coordinates. Suppose we want to scale our cube along the X and Y axes but keep it the same height (Z dimension)? Hit S-Shift-Z. Then moving the mouse keeps the Z dimension frozen but scales X and Y dimensions. Is the change in size too fast and uncontrollable? Then hold down the Shift key as you drag the mouse: this will reduce the sensitivity to mouse movements ten times. If you want to scale in precise increments? Hold down the Control key as you drag (the amount of increment is set and can be changed in User Preferences). If you want a precise scale? Just type it in rather than moving the mouse and then hit Enter.

Chris Yonge - 20180716