Fundamental idea:

* Take a sphere object (or even a cube) and texture it from a cubmap “skybox” texture.
* Should be smoothly shaded
* Place camera at the centre of the sphere object

Simple way:

* Draw a model with inverted normals (so that it will actually be drawn)   
  OR  
  Turn off back-face culling and draw the object
* Use the normal to lookup the cube map texture value
* Make the sphere REALLY, REALLY large so it holds the entire scene.  
  BEWARE about your “far plane” that it doesn’t cut off the back of this sphere
* Good Things: It’s simple
* Bad Things: Need a larger depth buffer range (far – near plane) 🡪 might lead to z fighting

More complicated (“more correct”) way

* This avoids the “Bad Things” above:
* Draw a tiny sphere at the camera location
* Turn off the depth buffer writing
* Draw the sphere skybox first (before anything else)

Even fancier way:

* Perform a calculation to generate the skybox within the shader
* Frank Luna’s DirectX 11 and 12 books have an example of this