```haskell
-- At the beginning, we specify the module name and the functions that it will export.
-- It exports functions because when you import a module, you get the functions from it.
module Geometry (
        sphereVolume,
        sphereArea,
        cubeVolume,
        cubeArea,
        cuboidArea,
        cuboidVolume
) where
-- After the "where" is where we start writing our functions.

sphereVolume :: Float -> Float
sphereVolume radius = (4.0 / 3.0) * pi * (radius ^ 3)

sphereArea :: Float -> Float
sphereArea radius = 4 * pi * (radius ^ 2)

cubeVolume :: Float -> Float
cubeVolume side = cuboidVolume side side side

cubeArea :: Float -> Float
cubeArea side = cuboidArea side side side

cuboidArea :: Float -> Float -> Float -> Float
cuboidArea a b c = rectangleArea a b * 2 + rectangleArea a c * 2 + rectangleArea b c * 2

cuboidVolume :: Float -> Float -> Float -> Float
cuboidVolume a b c = a*b*c

rectangleArea :: Float -> Float -> Float
rectangleArea a b = a*b
-- Notice how we don't export the rectangleArea function because we only want it to be a "helper" function. We don't want it to be accessed
-- outside of this module.

-- Now we just have to 'import Geometry' from a file that's in the same directory as Geometry.
```