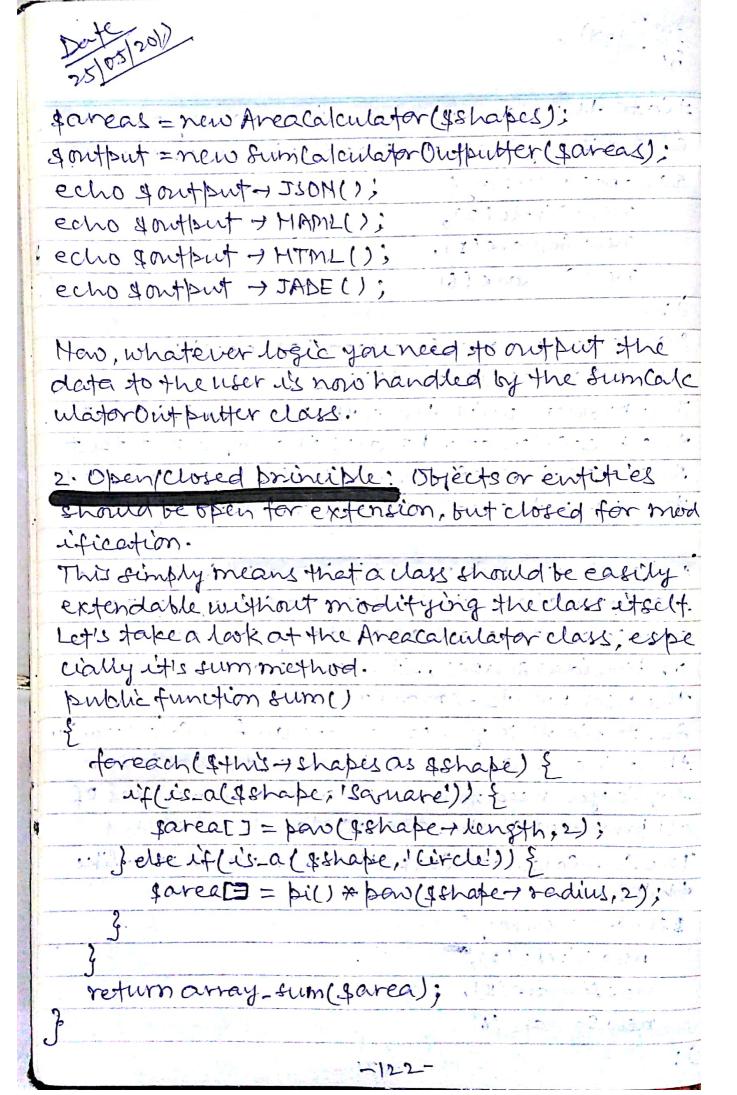
(120) tiate the class and bass in an array of shapes, and display the output at the bottom of the page. \$shapes = array(new Circle (2), new Square(5), in the water to be all the province new Sernate (6) , gareas = new Area (alculator (4shapes); echo saveas + output(); The problem with the output method is that the AreaCalculator handles the logic to output the data. Therefore, what if the user wanted to output the data as Jean or semething letses All of that logic would be handled by the Areac alculator class, this is what spragainst; the Areacalculator class should only sum the areas of provided shapes, it should not care whether the user wants ison or HTML. So, to fix this you can create an Sum Calculator Ou Aputter class and use this to hariable whatever logic you need to handle how the sum areas of all provided shapes are obsplayed, The Sumcalculator Dutputter class would wask like thits: LAW and Laboured A i and a Class wall \$ shapes = array (new Circle (2), new square (5), new Square (6) -121-





If we wanted the sum method to beable to sum the areas of more shapes, we would have to add more iffelse blocks and that goes against the. open/closed principle. A way we can make this sum method betteras . to remove the logic to calculate the area of each shape out of the sum method and attach it to the shape's class. class samare Frok Endals on Froti bublic slength; public function _ construct (& longth) 4this - length = slongth; public function areal) ; will said sold return son (4this, length, 2); The same thing should be done for the wicle class, an area method should be added. Now, to calculate the sum of any shape provided should be as simple as: Public function sum() toreach (4this - shapes as fehape); sarea[] = sshape-area(); -123-

Scanned by CamScanner

Maio me c

Now we can create another shape class and pass it in when calculating the fum neithbout break ing our code. However, now another problem arises, how do we know that the object passed into the Areacalculator is actually a shape or if the shape has a method named area?

Coding to an interface is an integral part of S.O.L. I. D., a quick example is the create an integral example is the create an interface, that every shape implements!

interface shape Interface

bublic function areal);

class lircle impléments shape Interface

bublic gradius;

public function_construct (\$radius)

Sthis-radius = gradius;

public function areact

return pic + pow (8this -, raidius, 2);

In our Areacalculator sum method we can check if the shapes provided are actually instances of the shape siterface, otherwise we throw an exception:

-124-

. To Man 2 30 24

1 1 the property to enable to be the state.

V5[05[20]) public function sum () 10 00 00 100 100 foreach (sthis - shapes as ashabe) f if (is-alshape, 'Shepemterface')) & in sareal] = 48hape - areaw; of out in 1100 continue; throw new Areacalculator Invalid shape Exception. 3. Liskov substitution principle: Every subclass/ derived class should be substitutable for their base/parent class. class A : Dublic function fine () { } class B extends A public function fire(i & for hit toil toil unction do something (A/40ts) Modo something with it Lip says that it we start use of B instead of -125-