

```

class Circle
  include Comparable

  def initialize(r)
    @radius=r
  end

  def area
    3.14159*@radius*@radius
  end

  def <=>(cir)
    if cir.area == self.area then
      0
    else
      1
    end
  end
end

```

```

class Shapes
  def initialize(n)
    @shapeArray = []
  end

  def addCircle(cir)
    @shapeArray << cir
  end
end

```

```

c1 = Circle.new(3)
c2 = Circle.new(4)
c3 = Circle.new(3)

```

```

if c1 == c2 then
  print "c1 is the same as c2;"
else
  print "c1 is different than c2;"
end

```

```

if c1 == c3 then
  print "c1 is the same as c3;"
else
  print "c1 is different than c3;"
end

```

```

if c2 == c3 then
  print "c2 is the same as c3;"
else

```

```
    print "c2 is different than c3;"  
end
```

```
shape1 = Shapes.new(3)  
shape1.addCircle(Circle.new(rand(1)))  
shape1.addCircle(Circle.new(rand(2)))  
shape1.addCircle(Circle.new(rand(3)))
```

```
puts shape1.min()  
puts shape1.sort()
```