

Quiz 1: SRP Question

Question: Given code on the board, construct a new version of the enhanced Gear class to allow it to more closely adhere to SRP.

Drag from here

Construct your solution here

```
def ratio
  chainring / cog.to_f
end
```

```
@wheel = Wheel.new(26, 1.5)
puts @wheel.circumference
# -> 91.106186954104
```

```
end
```

```
puts Gear.new(52, 11, @wheel).gear_inches
# -> 137.09090909090909
puts Gear.new(52, 11).ratio
# -> 4.72727272727273
```

```
attr_reader :rim, :tire
```

```
end
```

```
def diameter
  rim + (tire * 2)
end
```

```
def gear_inches
  ratio * wheel.diameter
end
```

```
class Gear
```

```
attr_reader :chainring, :cog, :wheel
```

```
def circumference  
  diameter * Math::PI  
end
```

```
def initialize(chainring, cog, wheel=nil)  
  @chainring = chainring  
  @cog = cog  
  @wheel = wheel  
end
```

```
def initialize(rim, tire)  
  @rim = rim  
  @tire = tire  
end
```

```
class Wheel
```

Reset

Submit test

Quiz 1: SRP (/tests/1) **Solution**

Marks Obtained : 1

Question: Given code on the board, construct a new version of the enhanced Gear class to allow it to more closely adhere to SRP.

Drag from here Construct your solution here

```
class Gear
```

```
  attr_reader :chainring, :cog, :wheel
```

```
  def initialize(chainring, cog, wheel=nil)
    @chainring = chainring
    @cog = cog
    @wheel = wheel
  end
```

```
  def ratio
    chainring / cog.to_f
  end
```

```
  def gear_inches
    ratio * wheel.diameter
  end
```

```
end
```

```
class Wheel
```

```
  attr_reader :rim, :tire
```

```
def initialize(rim, tire)
  @rim = rim
  @tire = tire
end
```

```
def diameter
  rim + (tire * 2)
end
```

```
def circumference
  diameter * Math::PI
end
```

```
end
```

```
@wheel = Wheel.new(26, 1.5)
puts @wheel.circumference
# -> 91.106186954104
```

```
puts Gear.new(52, 11, @wheel).gear_inches
# -> 137.090909090909
puts Gear.new(52, 11).ratio
# -> 4.72727272727273
```

Reset

Get feedback

Correct!

Quiz 2: Coupling **Question**

Question: The original Gear class refers to the Wheel class by its name in the gear_inches method, creating a dependency. Create a new version of the Gear class that isn't glued to the Wheel class and is more loosely coupled.

Drag from here

Construct your solution here

```
# ...
```

```
attr_reader :chainring, :cog, :wheel
```

```
def initialize(chainring, cog, wheel)
  @chainring = chainring
  @cog = cog
  @wheel = wheel
end
```

```
end
```

```
class Gear
```

```
  Gear.new(
    :chainring => 52,
    :cog => 11,
    : wheel => Wheel.new(26, 1.5)).gear_inches
```

```
  Gear.new(52, 11, Wheel.new(26, 1.5)).gear_inches
```

```
  def gear_inches
    ratio * wheel.diameter
  end
```

Reset

Submit test

Quiz 2: Coupling (/tests/2)

Solution

Marks Obtained : 1

Question: The original Gear class refers to the Wheel class by its name in the gear_inches method, creating a dependency. Create a new version of the Gear class that isn't glued to the Wheel class and is more loosely coupled.

Drag from here

```
Gear.new(  
  :chainring => 52,  
  :cog => 11,  
  : wheel => Wheel.new(26, 1.5)).gear_inches
```

Construct your solution here

```
class Gear
```

```
  attr_reader :chainring, :cog, :wheel
```

```
  def initialize(chainring, cog, wheel)  
    @chainring = chainring  
    @cog = cog  
    @wheel = wheel  
  end
```

```
  def gear_inches  
    ratio * wheel.diameter  
  end
```

```
# ...
```

end

```
Gear.new(52, 11, Wheel.new(26, 1.5)).gear_inches
```

Reset

Get feedback

Correct!