```
# Instance variables are more powerful when you can guarantee a rigidity to the data the object is holding
# the self keyword refers to the object and not the class being called

# In Circle's constructor, set the instance variable self.radius to equal half the diameter that gets passed in.
class Circle:                                                    #create the class
    pi = 3.14                                                    # class variable

    def __init__(self, diameter):                                #create the object with 2 arguments
        print("Creating circle with diameter {d}".format(d=diameter))
        # Add assignment for self.radius here:
        self.radius = diameter / 2

    # Define a new method .circumference() for your circle object that takes only one argument, self,
    # and returns the circumference of a circle with the given radius by this formula:
    def circumference(self):                                     # create method with single argument
        circumference = 2 * self.pi * self.radius                    # note the reference to self
        return circumference


# Define three Circles with three different diameters.
# A medium pizza, medium_pizza, that is 12 inches across.
# Your teaching table, teaching_table, which is 36 inches across.
# The Round Room auditorium, round_room, which is 11,460 inches across.
medium_pizza = Circle(12)
teaching_table = Circle(36)
round_room = Circle(11460)

# Print out the circumferences of medium_pizza, teaching_table, and round_room
print(medium_pizza.circumference())
print(teaching_table.circumference())
print(round_room.circumference())
```