class SingletonObject {

      private static SingletonObject instance = new SingletonObject();

      private SingletonObject(){}

   public static SingletonObject getInstance(){

      return instance;

   }

   public void printMessage(){

      System.out.println("Hello from Singleton object!!!");

   }

}

public class Main {

   public static void main(String[] args) {

      SingletonObject object = SingletonObject.getInstance();

      object.printMessage();

   }

}

Top of Forminterface Geometric\_shape {

   void draw\_shape();

}

class Rectangle implements Geometric\_shape {

   public void draw\_shape() {

      System.out.println("Rectangle class::draw\_shape() method.");

   }

}

class Square implements Geometric\_shape {

   public void draw\_shape() {

      System.out.println("Square class::draw\_shape() method.");

   }

}

class Circle implements Geometric\_shape {

   public void draw\_shape() {

      System.out.println("Circle class::draw\_shape() method.");

   }

}

class ShapeFactory {

      public Geometric\_shape shapeObject(String shapeType){

      if(shapeType == null){

         return null;

      }

      if(shapeType.equalsIgnoreCase("Circle")){

         return new Circle();

      } else if(shapeType.equalsIgnoreCase("Rectangle")){

         return new Rectangle();

      } else if(shapeType.equalsIgnoreCase("Square")){

         return new Square();

      }

      return null;

   }

}

public class Main {

   public static void main(String[] args) {

            ShapeFactory shapeFactory = new ShapeFactory();

      Geometric\_shape shape\_Circle = shapeFactory.shapeObject("CIRCLE");

            shape\_Circle.draw\_shape();

            Geometric\_shape shape\_Rectangle = shapeFactory.shapeObject("RECTANGLE");

      shape\_Rectangle.draw\_shape();

      Geometric\_shape shape\_Square = shapeFactory.shapeObject("SQUARE");

            shape\_Square.draw\_shape();

   }

}

Bottom of Form