Solution to Lab week 2

1) Exercise P5.4, complete the main method below to ask the user to decide whether to continue. A YesNoChecker class is provided in the next page.

```java
import java.util.Scanner;
//This program asks the user to decide whether to continue.

class Continue
{
    public static void main(String[] args)
    {
        Scanner in = new Scanner(System.in);
        System.out.println("Do you want to continue?\n");
        String input = in.nextLine();
        YesNoChecker c = new YesNoChecker(input);

        // fill you code here
        if(c.isYes())
            System.out.println("OK");
        else if (c.isNo())
            System.out.println("Terminating");
        else
            System.out.println("Bad Input");
    }
}
```
// This class checks if the user wants to continue.
public class YesNoChecker
{
    // Constructs an input checker to determine if user wants to continue.
    public YesNoChecker(String input)
    {
        if (input.equalsIgnoreCase("Y") ||
            input.equalsIgnoreCase("Yes") ||
            input.equalsIgnoreCase("Sure") ||
            input.equalsIgnoreCase("Why not?") ||
            input.equalsIgnoreCase("OK"))
        {
            type = YES;
        }
        else if (input.equalsIgnoreCase("N") ||
            input.equalsIgnoreCase("No"))
        {
            type = NO;
        }
        else
            type = UNKNOWN;
    }

    // Test if the user wants to continue.
    public boolean isYes()
    {
        return type == YES;
    }

    // Test if the user does not want to continue.
    public boolean isNo()
    {
        return type == NO;
    }
}

private int type;
private static final int YES = 1;
private static final int NO = 0;
private static final int UNKNOWN = -1;
2) Exercise P6.1

```
import java.util.Scanner;
// This program converts dollar to euros.
public class CurrencyConverter
{
    public static void main (String[] args)
    {
        Scanner in = new Scanner(System.in);
        System.out.print("How many euros is one dollar: ");
        double rate = in.nextDouble();

        boolean done = false;
        while (!done)
        {
            System.out.print("Dollar value (Q to quit): ");
            String input = in.next();
            if (input.equalsIgnoreCase("Q"))
                done = true;
            else
            {
                double amount = Double.parseDouble(input);
                double exchange = amount * rate;
                System.out.printf("%.2f dollar = %.2f euro\n", amount, exchange);
            }
        }
    }
}
```
3) Exercise P8.5, refer to the class Geometry.java class in the next page, and complete the GeometryCalculator.java class.

```
//GeometryCalculator.java
import java.util.Scanner;
/**
   * This program calculate the volume
   * and surface area of a sphere, a cylinder, and a cone.
   */
public class GeometryCalculator
{
    public static void main(String[] args)
    {
        Scanner in = new Scanner(System.in);
        System.out.println("Please enter the radius: ");
        double r = in.nextDouble();
        System.out.println("Please enter the height: ");
        double h = in.nextDouble();
        System.out.println("The volume of the sphere is: 
            + Geometry.sphereVolume(r));
        System.out.println("The surface area of the sphere is: 
            + Geometry.sphereSurface(r));
        System.out.println("The volume of the cylinder is: 
            + Geometry.cylinderVolume(r, h));
        System.out.println("The surface area of the cylinder is: 
            + Geometry.cylinderSurface(r, h));
        System.out.println("The volume of the cone is: 
            + Geometry.coneVolume(r, h));
        System.out.println("The surface area of the cone is: 
            + Geometry.coneSurface(r, h));
    }
```
//Geometry.java

/**
 * This class has methods to calculate the volume and surface area of
geometric shapes.
 */
public class Geometry
{

    //Computes the volume of a sphere.
    public static double sphereVolume(double r)
    {
        return 4.0 * Math.PI * r * r * r / 3.0;
    }

    //Computes the surface area of a sphere.
    public static double sphereSurface(double r)
    {
        return 4.0 * Math.PI * r * r;
    }

    //Computes the volume of a cylinder.
    public static double cylinderVolume(double r, double h)
    {
        return Math.PI * r * r * h;
    }

    // Computes the surface area of a cylinder.
    public static double cylinderSurface(double r, double h)
    {
        return 2.0 * r * Math.PI * (r + h);
    }

    //Computes the volume of a cone.
    public static double coneVolume(double r, double h)
    {
        return Math.PI * r * r * h / 3.0;
    }

    //Computes the surface area of a cone.
    public static double coneSurface(double r, double h)
    {
        return Math.PI * r * (r + h);
    }
}
Exercise P8.6, based on the GeometryCalculator class given below, implement class Sphere, Cylinder and Cone to solve the Exercise P8.5 problem

```java
import java.util.Scanner;

/**
 * This is a test driver for the Sphere, Cylinder, and Cone class
 * This approach is more object-oriented than the previous one because it clearly separates the duties of each class.
 */
public class GeometryCalculator
{
    public static void main(String[] args)
    {
        Scanner in = new Scanner(System.in);

        System.out.println("Please enter the radius: ");
        double r = in.nextDouble();

        System.out.println("Please enter the height: ");
        double h = in.nextDouble();

        Sphere sp = new Sphere(r);
        double v = sp.getVolume();
        double s = sp.getSurfaceArea();
        System.out.println("The volume of the sphere is: " + v);
        System.out.println("The surface area of the sphere is: " + s);

        Cylinder cy = new Cylinder(r, h);
        v = cy.getVolume();
        s = cy.getSurfaceArea();
        System.out.println("The volume of the cylinder is: " + v);
        System.out.println("The surface area of the cylinder is: " + s);

        Cone co = new Cone(r, h);
        v = co.getVolume();
        s = co.getSurfaceArea();
        System.out.println("The volume of the cone is: " + v);
        System.out.println("The surface area of the cone is: " + s);
    }
}
```
/**
   * This class provides methods to compute the volume and surface area of a sphere.
   */
   
   public class Sphere
   {
   /**
   * Constructs a Sphere object with input: radius and height.
   * @param aRadius the radius
   * @param aHeight the height
   */
   public Sphere(double aRadius)
   {
       r = aRadius;
   }

   /**
    * Computes the volume of a sphere.
    * @return volume of sphere
    */
   public double getVolume()
   {
       return (4.0 / 3.0) * Math.PI * r * r * r;
   }

   /**
    * Computes the surface area of a sphere.
    * @return surface area of a sphere
    */
   public double getSurfaceArea()
   {
       return 4.0 * Math.PI * r * r;
   }

   private double r;
   }
public class Cylinder
{
    /**
     * Constructs a Cylinder object with input: radius and height.
     * @param aRadius the radius
     * @param aHeight the height
     */
    public Cylinder(double aRadius, double aHeight)
    {
        r = aRadius;
        h = aHeight;
    }

    /**
     * Computes the volume of a cylinder.
     * @return volume of a cylinder
     */
    public double getVolume()
    {
        return h * Math.PI * r * r;
    }

    /**
     * Computes the surface area of a cylinder.
     * @return surface area of a cylinder
     */
    public double getSurfaceArea()
    {
        return (2.0 * r * Math.PI * h) + (2.0 * Math.PI * r * r);
    }

    private double r;
    private double h;
}
/**
   * This class provides methods to compute the volume and surface area of a cone.
   */

public class Cone {

  /**
   * Constructs a Cone object with input: radius and height.
   * @param aRadius the radius
   * @param aHeight the height
   */
  public Cone(double aRadius, double aHeight) {
    r = aRadius;
    h = aHeight;
  }

  /**
   * Computes the volume of a cone.
   * @return volume of a cone
   */
  public double getVolume() {
    return (1.0 / 3.0) * Math.PI * r * r * h;
  }

  /**
   * Computes the surface area of a cone.
   * @return surface area of a cone
   */
  public double getSurfaceArea() {
    return Math.PI * r * (h + r);
  }

  private double r;
  private double h;
}