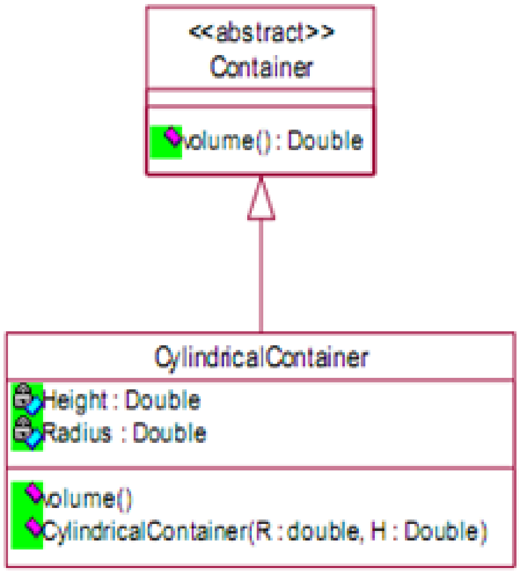
1.



The Volume of a Cylinder can be found with the following formula:

Volume = PI * Radius*Radius*Height   where PI=3.14159

It is required to map the above class diagram to Java code.

Note: Container is an abstract class.

Height & Radius are private variables

All the methods are public

(i) Write down the Java definition of class container

(ii) Write the Java Definition of class CylindricalContainer. (Implement the Methods)

(iii) Create an object from CylindricalContainer and display the volume

```
package com.mycompany.containerobj;

public class Containerobj

{

   public static void main(String[] args)

   {
       CylinderContainer c=new CylinderContainer(7,5);

       double calculatevolume=c.calculatevolume();

        System.out.println("Volume is :"+calculatevolume);

    }

}
```

```
package com.mycompany.containerobj;

public abstract class Container

{

    private double Height;

    private double Radius;

    public Container(double Height,double Radius)

    {

        this.Height = Height;

        this.Radius = Radius;

    }

    public void setHeight(double Height) {

        this.Height = Height;

    }

    public void setRadius(double Radius) {

        this.Radius = Radius;

    }

    public double getHeight() {
     return Height;
    }
    public double getRadius() {
      return Radius;
    }
    public abstract double calculatevolume();
}
```

Extends….

```
package com.mycompany.containerobj;
public class CylinderContainer extends Container
{
    public CylinderContainer(double Height, double Radius)
    {
        super(Height, Radius);
    }
    @Override
    public double calculatevolume()
    {
        double calculatevolume = Math.PI *getRadius()*getRadius()*getHeight();
        return calculatevolume;
    }
}
```

2.

A Student wants to create a game called “Life”, ‘life’ is a RPG game in which a player can move up, down, left & Right. In order to implement this game assume that you need to create an abstraction of the player controllers. Make sure to print the directions of the player when keys are pressed.

```
package com.mycompany.Test;
public class Test {
    public static void main(String[] args) {
        Player player = new Player(0, 0);
        player.moveUp();
        player.moveRight();
        player.moveDown();
        player.moveLeft();
    }
}

public class Player implements PlayerController {
    private int x;
    private int y;
    public Player(int x, int y) {
        this.x = x;
        this.y = y;
    }
    @Override
    public void moveUp() {
   y--;
```

```
    System.out.println("Moved up. Current position: (" + x + ", " + y + ")");
  }
  @Override
  public void moveDown() {
    y++;
    System.out.println("Moved down. Current position: (" + x + ", " + y + ")");
  }
  @Override
  public void moveLeft() {
    x--;
    System.out.println("Moved left. Current position: (" + x + ", " + y + ")");
  }
  @Override
  public void moveRight() {
    x++;
    System.out.println("Moved right. Current position: (" + x + ", " + y + ")");
  }
}

public interface PlayerController {
  void moveUp();
  void moveDown();
  void moveLeft();
  void moveRight();
}
```