

```

import java.awt.*;
import java.applet.*;
import java.awt.event.*;
public class africa extends Applet implements ActionListener {

    game ga;
    TextField in ;
    Button restart;
    Label l;
    Panel p,p1;
    public void init() {
        resize(400,330);
        ga = new game();
        p = new Panel();
        p1 = new Panel();
        in=new TextField(3);
        p.add(in);
        restart = new Button("restart game");
        p.add(restart);
        l = new Label("-----");
        p1.add(l);
        this.setLayout(new BorderLayout());
        add(p,"North");
        add(p1,"South");
        in.addActionListener(this);
        restart.addActionListener(this);
    }

    public void paint(Graphics g){

        ga.print_table(g,100,100);
        g.drawString("score of player 1 is :"+ga.get_score1(),50,220);
        g.drawString("score of player 2 is :"+ga.get_score2(),220,220);

        if (ga.get_score2()>20)g.drawString("player 2 won ,press restart" ,100,200);
        else if (ga.get_score1()>20)g.drawString("player 1 won , press restart" ,100,200);
        else g.drawString("turn is now for player :"+ga.get_player(),20,180);

    }

    public void actionPerformed(ActionEvent e){

        if (e.getSource() == restart ){
            ga=new game();
        }
        else{
            l.setText("");
            int hole = Integer.parseInt(in.getText());
            in.setText("");
            int res = ga.play(hole);

            if (res == -1 ) l.setText("you dont have beens to move from here !");
            else if (res == -2 ) l.setText( "you are not playing from your side !!");
            else if (res > 0 ) l.setText ("player got "+res+ " more beens");
        }
        repaint();
    }

}

```

```

import java.awt.*;

public class game{

    int[] holes;
    int current_player;
    int[] score;
    int left;

    public game(){

        holes = new int[12];
        score = new int[3];
        score[1]=0;
        score[2]=0;
        current_player=1;
        int left = 48;
        for(int i=0;i<12;i++) holes[i]=4;

    }

    public void print_table(Graphics g , int x , int y){
        g.drawRect(x+10,y-40,200,80);
        g.drawString("Player 1",45,y-20);
        g.drawString("Player 2",45,y+20);
        for(int i=0;i<6;i++) g.drawString(""+i+"="+holes[i],x+30+i*30,y+20);
        for(int i=6;i<12;i++) g.drawString(""+i+"="+holes[i],x+30+(11-i)*30,y-20);
    }

    public int get_score1(){
        return score[1];
    }
    public int get_score2(){
        return score[2];
    }
    public int get_player(){
        return current_player;
    }

    public int play(int hole){

        if (current_player == 1 && (hole<0 || hole >5) )return -2;
        if (current_player == 2 && (hole<6 || hole >11) )return -2;
        if (holes[hole]==0) return -1;

        int move = holes[hole];
        holes[hole]=0;
        while(move > 0){
            hole++;
            if(hole == 12 ) hole =0;
            holes[hole]++;
            move--;
        }

        int current_score =0;

        if ( current_player == 1 && hole>5 && (holes[hole] == 2 || holes[hole] == 3 ) ){
            score[1]=score[1]+holes[hole];
            current_score = holes[hole];
            holes[hole] = 0;
        }

        if ( current_player == 2 && hole<6 && (holes[hole] == 2 || holes[hole] == 3 ) ){
            score[2]=score[2]+holes[hole];
            current_score= holes[hole];
            holes[hole] = 0;
        }

        boolean found_bean=false;

        if ( current_player == 1 ){
            for(int i=0;i<6;i++) if (holes[i] !=0) found_bean=true;
            if (found_bean) current_player =2;
        }
        else{
            found_bean=false;
            for(int i=6;i<12;i++) if (holes[i] !=0) found_bean=true;
            if (found_bean) current_player =1;
        }

        return current_score;
    }

}

```