

জাভায় স্ক্রলবার সংযোজন

মো: আবদুল কাদের

স্ক্রলবার উইডো অ্যাপ্লিকেশনের একটি গুরুত্বপূর্ণ কম্পোনেন্ট। অ্যাপ্লিকেশন উইডোর দৃশ্যমান অংশ আমরা সাধারণত দেখতে পাই। তবে দৃশ্যমান অংশ ছাড়াও এর অংশ থাকতে পারে, যা প্রথমেই দেখা যায় না। এই অদৃশ্য অংশটুকু দেখার জন্য স্ক্রলবার প্রয়োজন হয়। দুই ধরনের স্ক্রলবার রয়েছে। একটি হরাইজন্টাল বা আনুভূমিক এবং আরেকটি ভার্টিক্যাল বা খাড়া আকৃতির। হরাইজন্টাল স্ক্রলবার দিয়ে উইডোর ডানে ও বামে এবং ভার্টিক্যাল স্ক্রলবার দিয়ে উপরে ও নিচের অদৃশ্য অংশ দেখা যায়।



এ প্রোগ্রামে আমরা তিনটি হরাইজন্টাল এবং একটি ভার্টিক্যাল স্ক্রলবার ব্যবহার করব এবং এর সাথে ইভেন্ট সংযোজন করব। ফলে স্ক্রলবারের স্ক্রল মুভমেন্ট করলে তার সাথে সাথে কিছু কাজ সংঘটিত হবে। প্রোগ্রামটি রান করার জন্য আমরা সফটওয়্যারটির Jdk1.4 ভার্সন ব্যবহার করব এবং প্রোগ্রামগুলো D:\ড্রাইভের java ফোল্ডারে সেভ করব।

নিচের প্রোগ্রামটি নোটপ্যাডে টাইপ করে ScrollProg.java নামে সেভ করতে হবে।

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class ScrollProg extends JFrame implements AdjustmentListener
{
    JLabel redLabel, greenLabel, blueLabel, label;
    JScrollBar red, green, blue, vs;
    JPanel a;
    public ScrollProg()
    {
        setTitle("ScrollBar Program");
        setSize(300, 200);
        addWindowListener(new WindowAdapter()
        { public void windowClosing(WindowEvent e)
        { System.exit(0);
        }
        });
        Container contentPane = getContentPane();
        JPanel p = new JPanel(); //1
        p.setLayout(new GridLayout(3, 2));
        p.add(redLabel = new JLabel("Red 0"));
        p.add(red = new JScrollBar(Adjustable.HORIZONTAL,
        100, 0, 0, 255)); //2
        red.setBlockIncrement(16);
        red.addAdjustmentListener(this);
        p.add(greenLabel = new JLabel("Green 0"));
        p.add(green = new
        JScrollBar(Adjustable.HORIZONTAL,
        0, 100, 0, 255)); //3
        green.setBlockIncrement(16);
        green.addAdjustmentListener(this); //5
        p.add(blueLabel = new JLabel("Blue 0"));
        p.add(blue = new JScrollBar(Adjustable.HORIZONTAL,
        0, 0, 0, 255)); //4
        blue.setBlockIncrement(16);
        blue.addAdjustmentListener(this);
        JPanel q = new JPanel();
        q.setLayout(new GridLayout(1, 2));
        q.add(vs = new JScrollBar(Adjustable.VERTICAL,
        0, 0, 0, 100)); //2
        q.add(label= new JLabel());
```

```
vs.addAdjustmentListener(this);
contentPane.add(p, "South");
contentPane.add(q, "East");
a = new JPanel();
a.setBackground(new Color(0, 0, 0));
contentPane.add(a, "Center");
}
public void
adjustmentValueChanged(AdjustmentEvent evt)
{ redLabel.setText("Red " + red.getValue());
greenLabel.setText("Green " + green.getValue());
blueLabel.setText("Blue " + blue.getValue());
a.setBackground(new Color(red.getValue(),
green.getValue(), blue.getValue()));
a.repaint();
label.setText(vs.getValue() + "");
}
public static void main(String[] args)
{JFrame f = new ScrollProg();
f.show();
}
```

কোড বিশ্লেষণ

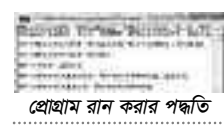
প্রোগ্রামটিতে JFrame-কে extends করা হয়েছে। ফলে একটি উইডো তৈরি হবে। উইডোতে ব্যবহৃত স্ক্রলবারের জন্য AdjustmentListener ব্যবহার করা হয়েছে। প্রোগ্রামের শুরুতেই চারটি লেবেল, চারটি স্ক্রলবার ও একটি প্যানেল ডিক্লেয়ার করা হয়েছে। এরপর কনস্ট্রাক্টরের মাধ্যমে টাইটেল বারে ScrollBar Program লেখাটি দেখানোর জন্য এর superclassে একটি স্ট্রিং 'ScrollBar Program' পাঠানো হচ্ছে, যাতে ফ্রেমের টাইটেল বারে তা প্রদর্শিত হয়। এরপর উইডোর সাইজ এবং উইডোর ক্লোজ বাটনে ক্লিক করলে উইডোটি যাতে বন্ধ হয়, সেজন্য প্রোগ্রাম লেখা হয়েছে।



একটি কন্টেইনার contentPane নেয়া হয়েছে, যেখানে আমরা কম্পোনেন্টগুলোকে একটির পর একটি সংযুক্ত করব। কন্টেইনার পাঁচটি ভাগে বিভক্ত থাকে- পূর্ব, পশ্চিম, উত্তর, দক্ষিণ ও মধ্য। আমরা কম্পোনেন্টগুলোকে যেখানে রাখতে চাই তা উল্লেখ করে দিতে হবে। ১নং লাইনে একটি প্যানেল নেয়া

হয়েছে, যাতে পরবর্তী লাইনে গ্রিড লেআউট সেট করা হয়েছে। ফলে তিনটি রো এবং দুটি কলাম আকারে কম্পোনেন্ট রাখা যাবে। ২, ৩, ৪নং লাইনে তিনটি হরাইজন্টাল স্ক্রলবার তৈরি করা হয়েছে। স্ক্রলবারগুলো setBlockIncrement-এর মাধ্যমে ইনক্রিমেন্ট ভেল্যু নির্দিষ্ট করা হয়েছে। স্ক্রলবার তৈরির সময় চারটি সংখ্যা দেয়া হয়েছে। এর প্রথমটি হলো স্ক্রলের পজিশন কোথায় থেকে হবে। যেমন- ২নং লাইনে প্রথম সংখ্যাটি ১০০ দেয়া হয়েছে। ফলে স্ক্রলটি বাই ডিফল্ট ১০০ পয়েন্টে অবস্থান করবে। দ্বিতীয় সংখ্যাটি হলো স্ক্রলের সাইজ কতটুকু হবে। যেমন- ৩নং লাইনে দ্বিতীয় সংখ্যার মান ১০০ দেয়া হয়েছে। ফলে স্ক্রলের সাইজটি ১০০ পয়েন্টের সমান হবে। তৃতীয় এবং চতুর্থ সংখ্যাটি স্ক্রলবারের দৈর্ঘ্য প্রকাশ করে। যেমন- আমরা ২৫৫ সেট করেছি। স্ক্রলবারের ডান/বাম পাশের অ্যারোতে ক্লিক করলে এটি ১৬ পয়েন্ট করে বাড়তে/কমতে থাকবে। স্ক্রলবারের স্ক্রল মুভ করলে এর সাথে সাথে যাতে অন্য একটি কাজ হয়, সেজন্য ইভেন্ট সংযোগ করা হয়েছে (৫নং লাইন)। তিনটি লেবেলের পাশাপাশি তিনটি স্ক্রলবার প্যানেল p-তে যুক্ত করে কন্টেইনারের দক্ষিণ অংশ সংযুক্ত করেছি। তিনটি স্ক্রলবার থেকে সংখ্যা নিয়ে একটি রং তৈরি করবে, যা কন্টেইনারের মধ্যখানে সংযুক্ত প্যানেলের ব্যাকগ্রাউন্ড কালার পরিবর্তন করবে।

কন্টেইনারের পূর্ব পাশে একটি ভার্টিক্যাল স্ক্রলবার সংযুক্ত করার জন্য একটি প্যানেল q নেয়া হয়েছে। প্যানেলে একটি ভার্টিক্যাল স্ক্রলবার এবং একটি লেবেল সংযুক্ত করা হয়েছে, যার মাধ্যমে স্ক্রল মুভ করলে স্ক্রলের মুভের পরিমাণ সংখ্যায় প্রকাশ করবে। এই স্ক্রলবারটির সর্বোচ্চ মান ১০০ সেট করা হয়েছে। তাই এটি ১০০ পর্যন্ত মুভ করবে।



প্রোগ্রাম রান করা
জাভার আগের পোর্টাম গুলোর মতো কমান্ড প্রম্পট ওপেন করে নিচের চিত্রের মতো করে রান করতে হবে।



প্রোগ্রাম রান করার পর আউটপুট ফিডব্যাক : balaith@gmail.com

