



জাভাতে সুইং প্রোগ্রামিং

মো: আবদুল কাদের

গত পর্বে জাভা ল্যাঙ্গুয়েজে উইন্ডোনির্ভর জাভার নতুন টেকনোলজি সুইং (Swing)-এর ওপর একটি ধারণা দেয়া হয়েছে। এ পর্বে সুইংয়ের ওপর একটি প্রোগ্রাম দেখানো হয়েছে। সুইং দিয়ে একটি লেখাকে উইন্ডোর বিভিন্ন পজিশনে তার ব্যাকগ্রাউন্ডসহ বিভিন্ন উপায়ে উপস্থাপন করার পদ্ধতি দেখানো হয়েছে এ লেখায়। জাভা প্রোগ্রামে সুইংয়ের ক্লাস, মেথড, ইন্টারফেস ব্যবহার করার জন্য জাভা লাইব্রেরি থেকে সুইং প্যাকেজটিকে ইম্পোর্ট করে নিতে হবে। সেই সাথে এটি উইন্ডো সংক্রান্ত জাভার মূল ক্লাসগুলো থেকে অনেক আপগ্রেডেড এবং এক্সটেন্ডেড হওয়ায় javax ফোল্ডারের সুইং ফোল্ডারে এর ক্লাসগুলো রাখা আছে। এজন্য সুইং প্যাকেজটিকে ইম্পোর্ট করার জন্য প্রোগ্রামের শুরুতে নিম্নোক্ত লাইনটি লিখতে হবে :

```
import javax.swing.*;
```

প্রোগ্রামটি রান করার পদ্ধতি অন্যান্য জাভা প্রোগ্রামের মতোই। এজন্য অবশ্যই আপনার কমপিউটারে Jdk সফটওয়্যার ইনস্টল থাকতে হবে। এখানে সফটওয়্যারটির Jdk1.4 ভার্সন ব্যবহার করা হয়েছে এবং প্রোগ্রামগুলো D:\ড্রাইভের java ফোল্ডারে সেভ করা হয়েছে।

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

```
import java.awt.*;
import javax.swing.*;
class SwingLabel extends JFrame {
public SwingLabel(){
super("Swing Label");
setSize(600, 100);
JPanel content = (JPanel)
getContentPane(); content.setLayout(new
GridLayout(1, 4, 4, 4));
JLabel label = new JLabel();
label.setText("You");
label.setBackground(Color.red);
content.add(label);
label = new JLabel("Are",
SwingConstants.CENTER);
label.setOpaque(true);
cground(Color.blue);
content.add(label);
label = new JLabel("Using");
label.setFont(new Font("Helvetica",
Font.BOLD, 18));
label.setOpaque(true);
label.setBackground(Color.red);
content.add(label);
```

```
label = new JLabel("Swing Label",
SwingConstants.RIGHT);
label.setVerticalTextPosition(SwingConsta
nts.TOP);
label.setOpaque(true);
label.setBackground(Color.yellow);
content.add(label);
public static void main(String args[]) {
SwingLabel frame = new
SwingLabel();
frame.setDefaultCloseOperation(JFrame
e.EXIT_ON_CLOSE);
frame.setVisible(true);
}
```

কোড বিশ্লেষণ

প্রোগ্রামটিতে JFrame কে extends করা হয়েছে। ফলে একটি উইন্ডো তৈরি হবে। উইন্ডোর টাইটেল বারে SwingLabel লেখাটি দেখানোর জন্য এর super ক্লাসে একটি স্ট্রিং 'SwingLabel' পাঠানো হয়েছে, যাতে ফ্রেমের টাইটেল বারে তা প্রদর্শিত হয়। প্রোগ্রামে ব্যবহার হওয়া ফ্রেমে বিভিন্ন কনটেন্ট বসানোর জন্য একটি প্যানেল নেয়া হয়েছে। প্যানেলে কনটেন্টগুলো সাজানোর জন্য একটি লেআউট নেয়া হয়েছে, যার নাম GridLayout। জাভাতে কাজের প্রকৃতি অনুসারে বিভিন্ন ধরনের লেআউট ব্যবহার করা যায়। যেমন- GridBagLayout, BorderLayout ইত্যাদি। গ্রিড লেআউটের মাধ্যমে এই প্রোগ্রামে একটি রো ও চারটি কলাম তৈরি করা হবে।

এবার লেবেল তৈরির পর্ব। লেবেল তৈরির জন্য label নামে একটি লেবেল অবজেক্ট তৈরি করা হয়েছে। এই অবজেক্টের মাধ্যমে কী ধরনের লেবেল আমরা দেখাতে চাই, সেটি setText মেথডে উল্লেখ করা হচ্ছে। এরপর লেবেলের ব্যাকগ্রাউন্ডে সাদা রং সেট করে add মেথডের মাধ্যমে তা প্যানেলে সংযুক্ত করা হচ্ছে, কিন্তু এখানে setOpaqu মেথড ব্যবহার না করায় এর ব্যাকগ্রাউন্ডের কোনো তারতম্য দেখা যাবে না। বাই ডিফল্ট লেবেলটি প্রথম কলামের বামদিকে থাকবে।

প্রোগ্রামটিতে মোট চারটি লেবেল তৈরি করা হয়েছে, যা একটি প্যানেলে জাভার গ্রিড লেআউটের মাধ্যমে ভিন্নভাবে সাজানো হবে।

দ্বিতীয় লেবেলটি ('Are') label অবজেক্ট তৈরির সাথে সাথেই দেয়া হচ্ছে। এই লেবেলটি দ্বিতীয় কলামের মধ্যস্থলে প্রদর্শনের কোড ব্যবহার করা হয়েছে। এরপর লেবেলের

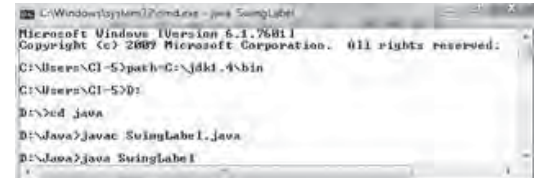
ব্যাকগ্রাউন্ড নীল রং সেট করে add মেথডের মাধ্যমে তা প্যানেলে সংযুক্ত করা হচ্ছে। এখানে setOpaque মেথড ব্যবহার করায় এর ব্যাকগ্রাউন্ড নীল রংয়ের দেখা যাবে।

তৃতীয় লেবেলটি ('Using') একইভাবে add মেথডের মাধ্যমে প্যানেলে সংযুক্ত করা হয়েছে। এর ব্যাকগ্রাউন্ড দেয়া হয়েছে লাল রংয়ের। উপরের দুটি লেবেলের সাথে এর একটি পার্থক্য হলো, এতে 'Using' লেখাটি নির্দিষ্ট ফন্ট Helvetica, বোল্ড আকারে এবং ফন্ট সাইজ ১৮-তে প্রদর্শিত হবে। চতুর্থ লেবেলটি একইভাবে চতুর্থ কলামের ডান পাশে হলুদ ব্যাকগ্রাউন্ডে প্রদর্শন করা হচ্ছে।

সবশেষে মেইন মেথডে SwingLabel ক্লাসের অবজেক্ট তৈরি হবে। অবজেক্ট তৈরি হওয়ার সাথে সাথেই উপরের বর্ণনা মোতাবেক সব কাজ সংঘটিত হবে। মূলত মেইন মেথড থেকেই প্রোগ্রাম রান করে। প্রোগ্রামটি রান করার পর উইন্ডোর ক্লোজ বাটনে ক্লিক করলে যাতে তা পুরোপুরি বন্ধ হয়ে যায় সেজন্য setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE) ব্যবহার করা হয়েছে। এই মেথডটি ব্যবহার না করা হলে ক্লোজ বাটনে ক্লিক করলে উইন্ডোটি দৃশ্যমান না থাকলেও রানিং অবস্থায় থাকে এবং মেমরি ব্যবহার করতে থাকে। এরপর প্রোগ্রামটি রান করার পর যাতে উইন্ডোটি দেখা যায়, সেজন্য setVisible(true) কোড ব্যবহার করা হয়েছে। এটি ব্যবহার না করলে উইন্ডো দেখা যাবে না, যদিও তা তৈরি হয়।

প্রোগ্রাম রান করা

জাভার আগের প্রোগ্রামগুলোর মতো কমান্ড প্রম্পট ওপেন করে নিচের চিত্রের মতো করে রান করতে হবে।



চিত্র-১ : প্রোগ্রাম রান করার পদ্ধতি



চিত্র-২ : প্রোগ্রাম রান করার পর আউটপুট

সামনের পর্বগুলোতে সুইংনির্ভর প্রয়োজনীয় কিছু প্রোগ্রাম নিয়ে আলোচনা করা হবে [ক্লিক](mailto:balaith@gmail.com)

ফিডব্যাক : balaith@gmail.com