

# জাভাতে থ্রেডিং প্রোগ্রাম তৈরি

পর্ব ২

মো: আবদুল কাদের

**জা**ভার তাৎপর্যমণ্ডিত বৈশিষ্ট্যের কারণেই জাভা অন্যান্য প্রোগ্রামিং ল্যাঙ্গুয়েজ থেকে আলাদা এবং জনপ্রিয়। জাভা দিয়ে লেখা প্রোগ্রাম যেকোনো মেশিনে রান করানো যায় বলেই প্লাটফর্ম ইনডিপেন্ডেন্ট ল্যাঙ্গুয়েজ হিসেবে এর গ্রহণযোগ্যতা ব্যাপক। অধুনা অ্যানড্রয়েড নির্মিত স্মার্টফোনগুলোতে যে অ্যাপস ব্যবহার করা হচ্ছে তাও জাভা দিয়ে করা সম্ভব। জাভা অ্যাপ্লিকেশন বা প্রোগ্রামে থ্রেডিংয়ের ব্যবহার খুবই গুরুত্বপূর্ণ।

গত পর্বে জাভা দিয়ে থ্রেড তৈরির প্রোগ্রাম দেখানো হয়েছে। আজকের পর্বে থ্রেড ক্লাস ব্যবহার করে আরেকটি মজার প্রোগ্রাম দেখানো হবে। প্রোগ্রামটি ছোট হলেও এখানে কোডিং কম ব্যবহার করা হয়েছে। তবে একই ধরনের কাজ বা রিপটিটেড কাজগুলোকে আমরা কোডিংয়ে কিছু পদ্ধতি যেমন লুপ ব্যবহার করে বারবার কাজগুলো করতে পারি। এতে কোডিংয়ের পরিমাণ কমে যায় এবং সেই সাথে কোডিং এরর কম হয় ও এরর খুঁজে বের করাও সহজ হয়। কম কোড লেখার আরেকটি গুরুত্বপূর্ণ সুবিধা হলো থার্ড পার্টি বা অন্যান্য প্রোগ্রামার এ প্রোগ্রামটি দেখে সহজেই কোডিং বুঝতে পারবেন। ফলে এতে সময় ও ভুলের পরিমাণ দুটোই কমেবে।

এ পর্বে আমরা জাভা দিয়ে নৌকা বানানোর প্রোগ্রাম দেখব। প্রোগ্রামটি রান করার জন্য অবশ্যই আপনার কমপিউটারে Jdk সফটওয়্যার ইনস্টল থাকতে হবে। আমরা সফটওয়্যারটির Jdk1.4 ভার্সন ব্যবহার করব এবং প্রোগ্রামগুলো D:\ড্রাইভের java ফোল্ডারে সেভ করব। নিম্নের এই প্রোগ্রামটি নোটপ্যাডে টাইপ করে Boat.java নামে সেভ করুন।

```
import java.awt.*;
import java.applet.Applet;
```

```
/*<applet code="Boat.class" width=300 height=300></
applet>*/
public class Boat extends Applet implements Runnable
{
    int x1[]={100,160,400,100,170,390,140,140,150,115,11
5,110,140,185};//1
    int y1[]={200,270,270,200,230,230,205,205,210,285,28
5,315,310,150};//2
    int x2[]={160,400,460,170,390,460,120,150,130,140,11
0,135,135,380};//3
    int y2[]={270,270,200,230,230,200,290,210,300,310,31
5,330,330,150};//4
    int j=0, k=0, red=0, green=0, blue=0;
```

```
public void init()
{
    new Thread (this).start();
}
public void update (Graphics g)
{
    //g.setColor(new Color(red, green, blue));
    for(k=0;k<=13;k++)
    {
        g.drawLine (x1[k],y1[k],x2[k],y2[k]);
    }
    g.drawOval(175,150,20,80);
    g.drawOval(370,150,20,80);
}
```

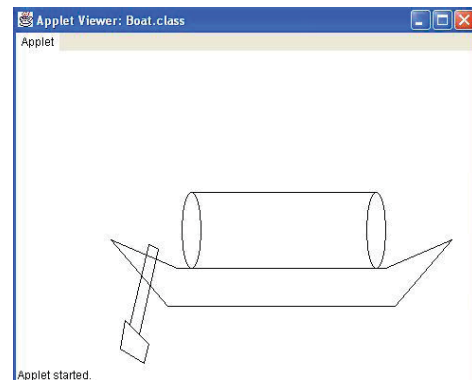
```
public void run()
```

```
{
    for (j=0; ;j++)
    {
        try
        {
            Thread.sleep (1000);
        }
        catch(Exception e){}
        if (j==14)j=0;
        //red=(int)(Math.random()*255.0);
        //green=(int)(Math.random()*255.0);
        //blue=(int)(Math.random()*255.0);
        repaint();
    }
}
```

## কোড বিশ্লেষণ ও প্রোগ্রাম রান করা

প্রোগ্রামটিতে <applet code> ব্যবহার করা হয়েছে উইন্ডো তৈরি করার জন্য, যার দৈর্ঘ্য ও প্রস্থ হবে ৩০০ ও ৩০০। 1, 2, 3 ও 4 চিহ্নিত লাইনে ইন্টিজার টাইপের চারটি অ্যারে নেয়া হয়েছে। অ্যারে হলো অনেকগুলো ভেরিয়েবল রাখার জায়গা। এই অ্যারেগুলোতে আমরা ১৪টি ভেরিয়েবল রেখেছি, যা প্রয়োজনমতো প্রোগ্রামে ব্যবহার করা হবে। এরপর init() মেথড ব্যবহার করা হয়েছে, যার মাধ্যমে অ্যাপলেট চালু হবে। ফলে উইন্ডো ওপেন হবে। এরপর আপডেট মেথড ব্যবহার করা হয়েছে, যা মূলত রান মেথডের মধ্য থেকে রান করবে। গ্রাফিক্স মেথডের মধ্যে গ্রাফিক্স সংক্রান্ত কোডগুলো লেখা হয়েছে। যেমন চতুর্ভুজ, সরলরেখা এবং বৃত্ত আঁকা হয়েছে। সবগুলোর সমন্বয়ে একটি নৌকা নির্মিত হয়েছে।

চিত্র : প্রোগ্রাম রান করার পদ্ধতি



চিত্র : প্রোগ্রাম রান করার পর আউটপুট

(বাকি অংশ ৩৭ পাতায়)