APACHE POI PPT - SLIDE MANAGEMENT

After completing this chapter, you will be able to delete, reorder, and perform read and write operations on a slide.

Changing a Slide

We can change the page size of a slide using the **setPageSize** method of the **XMLSlideShow** class.

Initially create a presentation as shown below:

```
File file=new File("C://POIPPT//Examples// TitleAndContentLayout.pptx");

//create presentation
XMLSlideShow ppt = new XMLSlideShow(new FileInputStream(file));
```

Get the size of the current slide using the **getPageSize** method of the **XMLSlideShow** class.

```
java.awt.Dimension pgsize = ppt.getPageSize();
```

Set the size of the page using the **setPageSize** method.

```
ppt.setPageSize(new java.awt.Dimension(1024, 768));
```

The complete program for changing the size of a slide is given below:

```
import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
import org.apache.poi.xslf.usermodel.XMLSlideShow;

public class ChangingSlide {

   public static void main(String args[]) throws IOException{

      //create file object
      File file = new File("TitleAndContentLayout.pptx");

      //create presentation
      XMLSlideShow ppt = new XMLSlideShow();

      //getting the current page size
      java.awt.Dimension pgsize = ppt.getPageSize();
      int pgw = pgsize.width; //slide width in points
      int pgh = pgsize.height; //slide height in points

      System.out.println("current page size of the PPT is:");
      System.out.println("width :" + pgw);
      System.out.println("height :" + pgh);

      //set new page size
      ppt.setPageSize(new java.awt.Dimension(2048,1536));

      //creating file object
      FileOutputStream out = new FileOutputStream(file);

      //saving the changes to a file
      ppt.write(out);
      System.out.println("slide size changed to given dimentions ");
      out.close();
```

```
    }
}
```

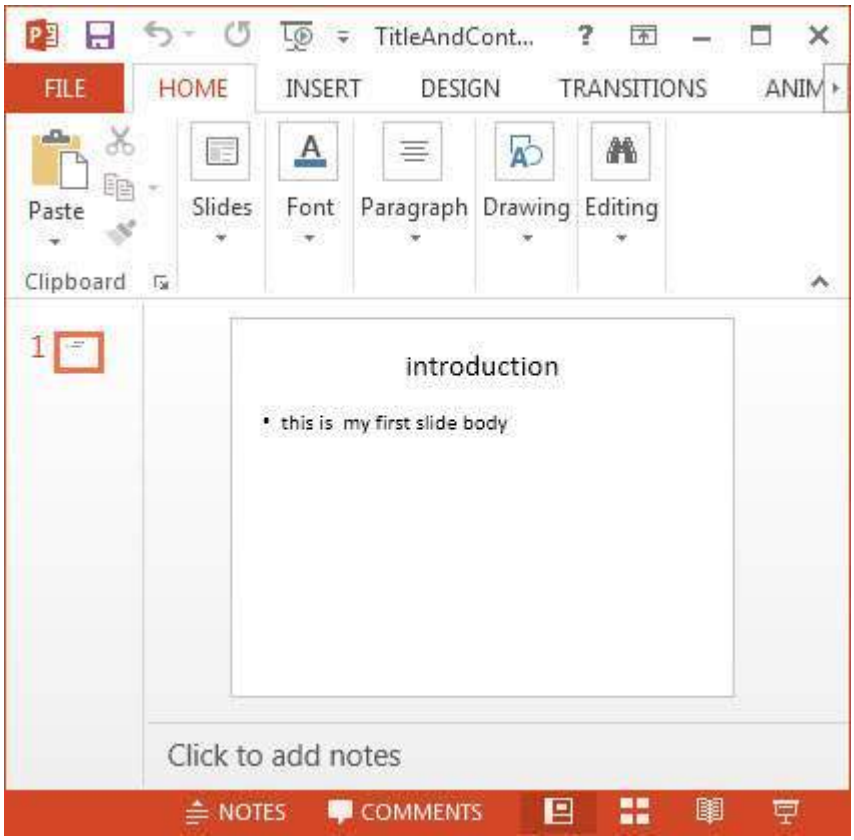
Save the above Java code as **ChangingSlide.java**, and then compile and execute it from the command prompt as follows:

```
$javac ChangingSlide.java
$java ChangingSlide
```

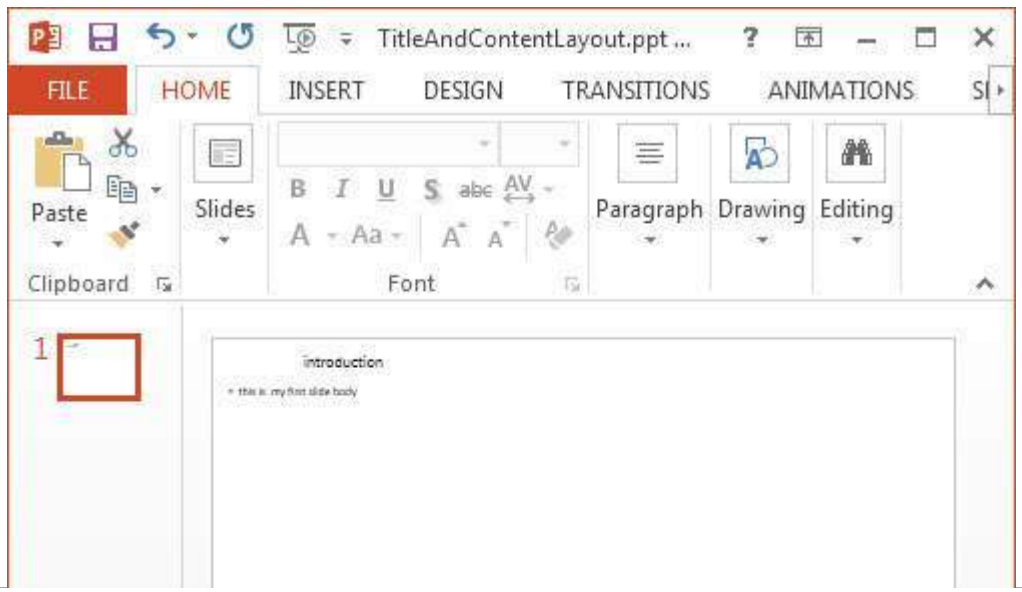
It will compile and execute to generate the following output.

```
current page size of the presentation is :
width :720
height :540
slide size changed to given dimensions
```

Given below is the snapshot of the presentation before changing the slide size:



The slide appears as follows after changing its size:



Reordering Slides

You can set the slide order using the **setSlideOrder** method. Given below is the procedure to set the order of the slides.

Open an existing PPT document as shown below:

```
File file=new File("C://POIPPT//Examples//example1.pptx");
XMLSlideShow ppt = new XMLSlideShow(new FileInputStream(file));
```

Get the slides using the **getSlides** method as shown below:

```
XSLFSlide[] slides =ppt.getSlides();
```

Select a slide from the array of the slides, and change the order using the **setSlideOrder** method as shown below:

```
//selecting the fourth slide
XSLFSlide selectesdslide= slides[4];

//bringing it to the top
ppt.setSlideOrder(selectesdslide, 1);
```

Given below is the complete program to reorder the slides in a presentation:

```
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;
import org.apache.poi.xslf.usermodel.XMLSlideShow;
import org.apache.poi.xslf.usermodel.XSLFSlide;

public class ReorderSlide {

   public static void main(String args[]) throws IOException{

      //opening an existing presentation
      File file=new File("example1.pptx");
      XMLSlideShow ppt = new XMLSlideShow(new FileInputStream(file));

      //get the slides
      XSLFSlide[] slides = ppt.getSlides();

      //selecting the fourth slide
      XSLFSlide selectesdslide = slides[13];

      //bringing it to the top
      ppt.setSlideOrder(selectesdslide, 0);

      //creating an file object
```

```
        FileOutputStream out = new FileOutputStream(file);

        //saving the changes to a file
        ppt.write(out);
        out.close();
    }
}
```

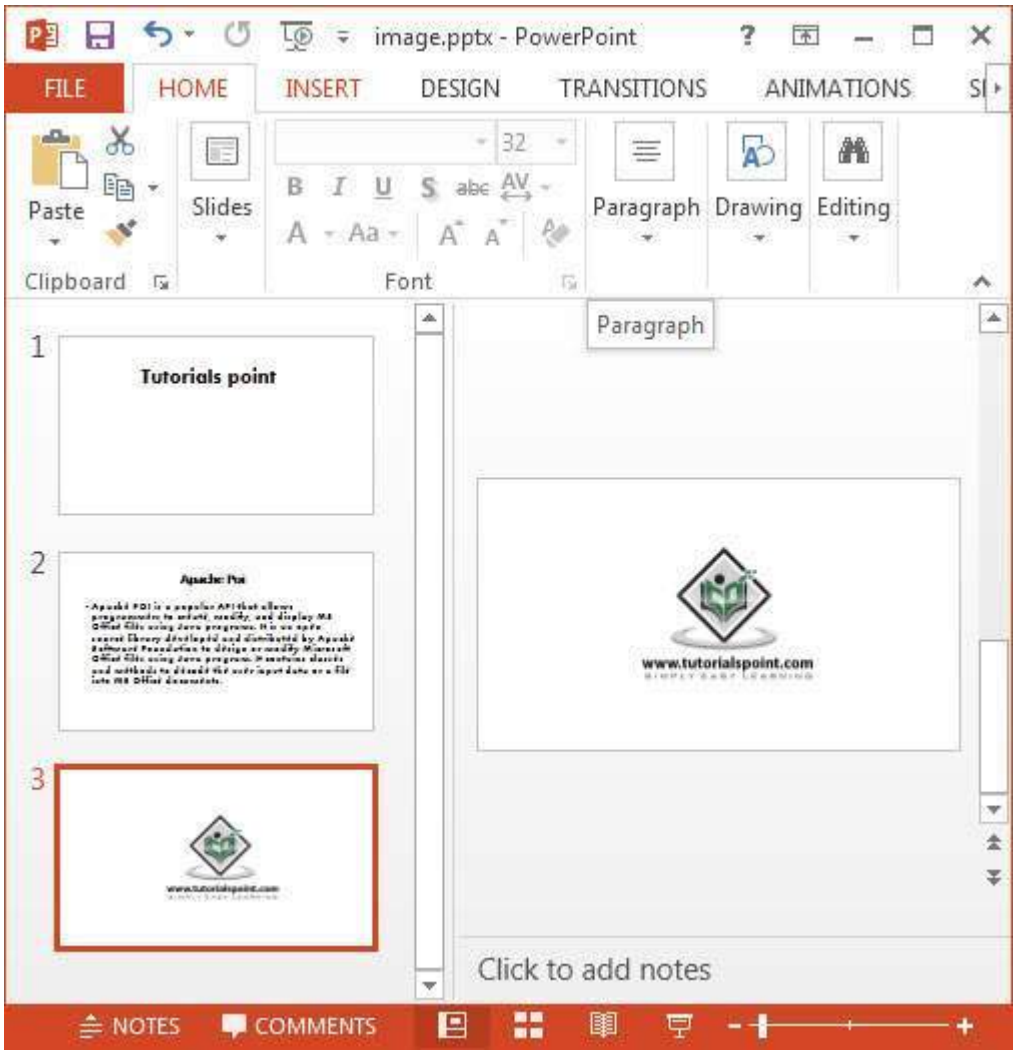
Save the above Java code as **ReorderSlide.java**, and then compile and execute it from the command prompt as follows:

```
$javac ReorderSlide.java
$java ReorderSlide
```

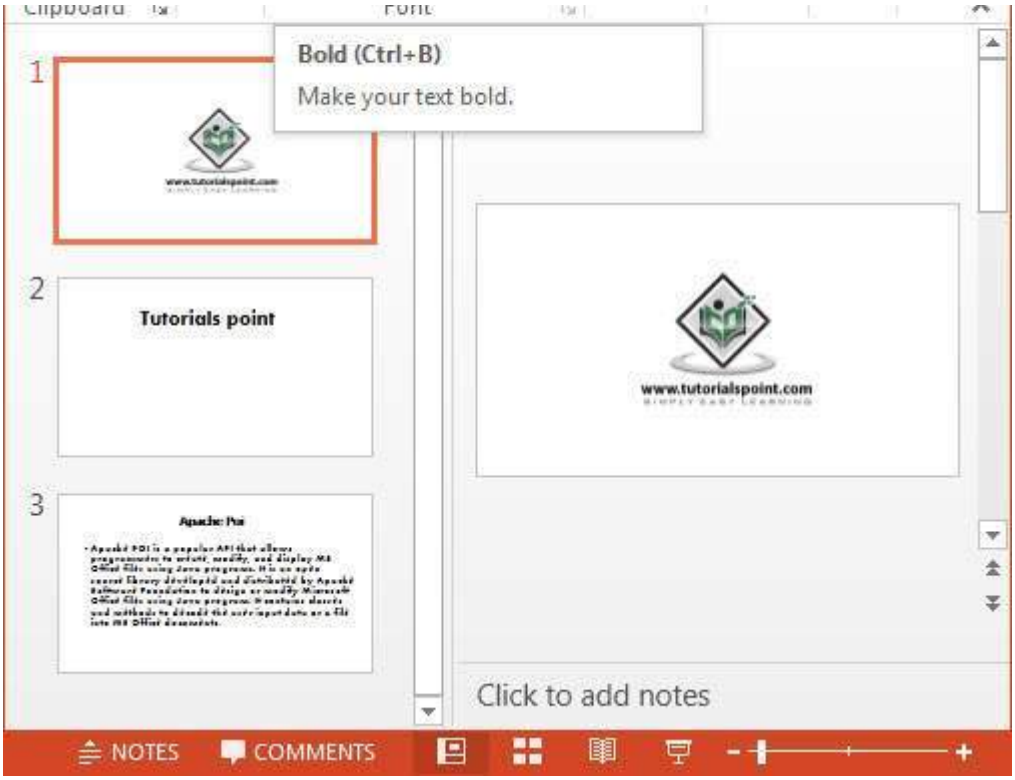
It will compile and execute to generate the following output.

```
Reordering of the slides is done
```

Given below is the snapshot of the presentation before reordering the slides:



After reordering the slides, the presentation appears as follows. Here we have selected the slide with image and moved it to the top.



Deleting Slides

You can delete the slides using the **removeSlide** method. Follow the steps given below to delete slides.

Open an existing presentation using the **XMLSlideShow** class as shown below:

```
File file=new File("C://POIPPT//Examples//image.pptx");
XMLSlideShow ppt = new XMLSlideShow(new FileInputStream(file));
```

Delete the required slide using the **removeSlide** method. This method accepts an integer parameter. Pass the index of the slide that is to be deleted to this method.

```
ppt.removeSlide(1);
```

Given below is the program to delete slides from a presentation:

```
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;

import org.apache.poi.xslf.usermodel.XMLSlideShow;

public class Deleteslide {

   public static void main(String args[]) throws IOException{

      //Opening an existing slide
      File file=new File("image.pptx");
      XMLSlideShow ppt = new XMLSlideShow(new FileInputStream(file));

      //deleting a slide
      ppt.removeSlide(1);

      //creating a file object
      FileOutputStream out = new FileOutputStream(file);

      //Saving the changes to the presentation
      ppt.write(out);
      out.close();
```

```
    }
}
```

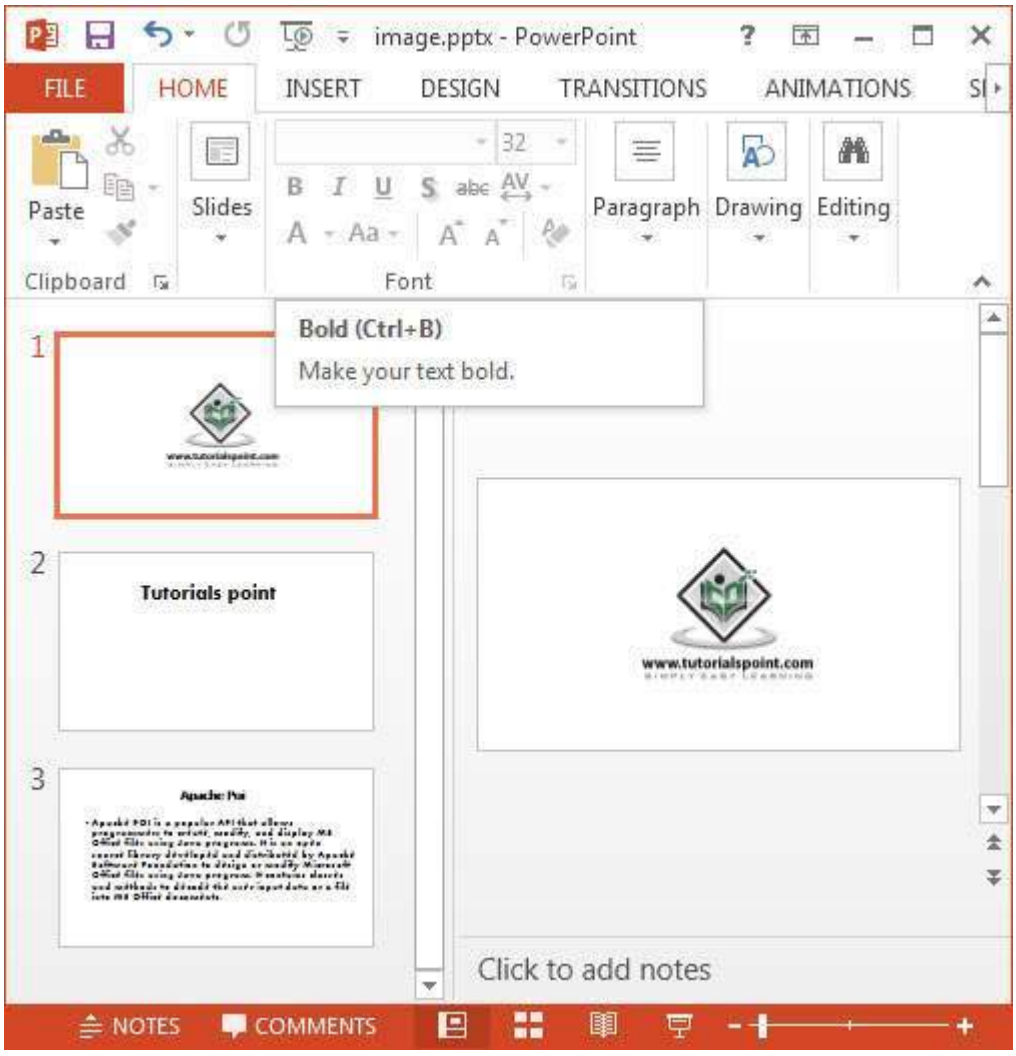
Save the above Java code as **Deleteslide.java**, and then compile and execute it from the command prompt as follows:

```
$javac Deleteslide.java
$java Deleteslide
```

It will compile and execute to generate the following output:

```
reordering of the slides is done
```

The snapshot below is of the presentation before deleting the slide:



After deleting the slide, the presentation appears as follows: