

## Exercises for Programming Practice 1

The submitted program will be evaluated as described in “Lesson Plan (2021)”.

Note:

- ✓ Do not create “module-info.java”, when you create a Java Project.
- ✓ Do not set Package name in the window “New Java Class”.
- ✓ The following information must be included as a comment at the first part of the program.
  - Contents of the program
    - ❖ Do not use the title of exercise for “contents of the program”  
Think about “contents of the program” yourself.
  - Submission date
  - Program creator
- ✓ Wrong file name (including case sensitivity) is not accepted.

The deadline for submitting the programs is 18:00 on December 9th, 2021.

Download “imageL1.jpg” and “image2.jpg” from “Week 11” page Resource of “Programming Practice 1”, manaba+R.

### Exercise 23 (file name “Exercise23.java”)

Create Java program “Exercise23.java” which displays the image. The program has to satisfy the following condition.

- (1) The window size is 500 pixels wide by 500 pixels vertical.
- (2) Reduce and display the entire image of the file “imageL1.jpg” in the window. The size of “imageL1.jpg” is 1080 x 720.
- (3) The upper left corner of the image is located at (10, 70) of the window as shown in Figure 1. The reason for drawing a little below is to prepare a space to place buttons in the future.

Tips1:

- (1) Obtain the size of “imageL1.jpg”.
- (2) In order to fit the image to the window size width 480, calculate the ratio to make the width 480 or less and assign it to the double variable “rate”.
- (3) Refer to “Quiz 8” or “image2.java” (Chapter 4 (2)) in “Scaling Display of Image” (The eleventh day) of “Programming Language”.

(10, 70)

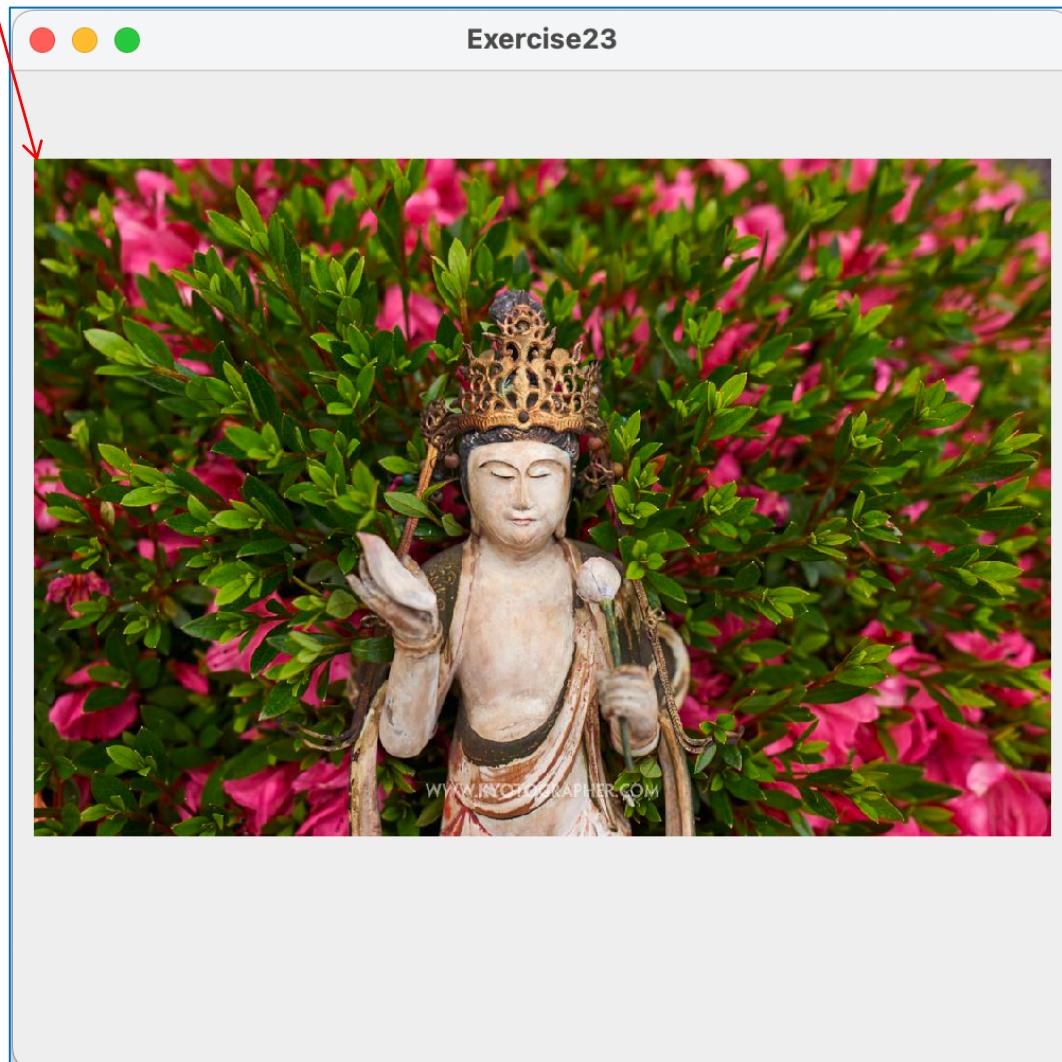


Figure 1. The result of Exercise 23.

#### Exercise 24 (file name “Exercise24.java”)

Create Java program “Exercise24.java” which cuts out arbitrary area of the image. The program has to satisfy the following condition.

- (1) The window size is 500 pixels wide by 800 pixels vertical.
- (2) Use the file “image2.jpg”. The upper left corner of the image is located at (10, 70) of the window. Since the size of “image2.jpg” is 480 pixels in width and 360 pixels in height, please display the whole in image in the window
- (3) Use the mouse as follows to obtain a part of the image and assign it to the BufferedImage variable “getImage”.
  - (a) Record the coordinate  $\alpha$  in the mouse press.

- (b) During dragging, the position of the mouse cursor is  $\beta$ . Draw a red rectangle with the upper left position as  $\alpha$  and the lower right position as  $\beta$ .
- (c) When the mouse is released, the coordinate  $\gamma$  is obtained. Substituting a partial image whose the upper left position is  $\alpha$  and whose lower right position is  $\gamma$  is assigned to the `BufferedImage` variable “`getImage`”.
- (4) Display “`getImage`” whose upper left corner of the image is located at  $(10, 440)$  of the window. In this exercise, there is no problem even if a large image is cut out and cannot be displayed in the window.

Refer “Image5.java” introduced in “Image Segmentation by Mouse” (The eleventh day) of “Programming Language”. An execution example of “Exercise24.java” is shown in Figure 2.

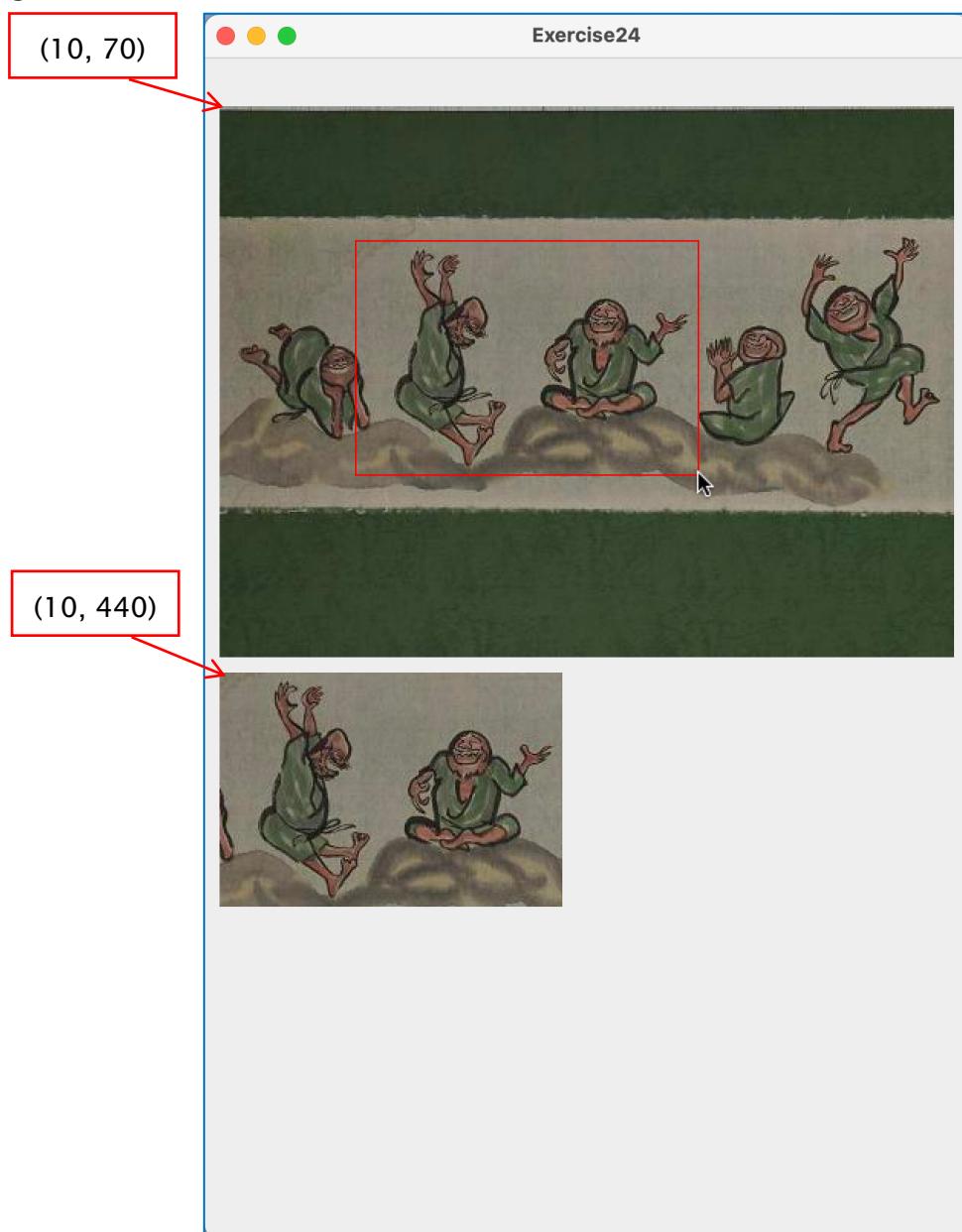


Figure 2. An example of Exercise 24.

### **Exercise 25 (file name “Exercise25.java”)**

Create Java program “Exercise25.java” which saves the cutout image. The program is added the following function to “Exercise24.java”.

- (1) Add a button named “save image” at the top of the screen to save the image when pressed.
- (2) After cutting out the image, the file “save.jpg” is saved by clicking the “save image” button.

An execution example of Exercise25.java is shown in Figure 3. The obtained image is shown in Figure 4.

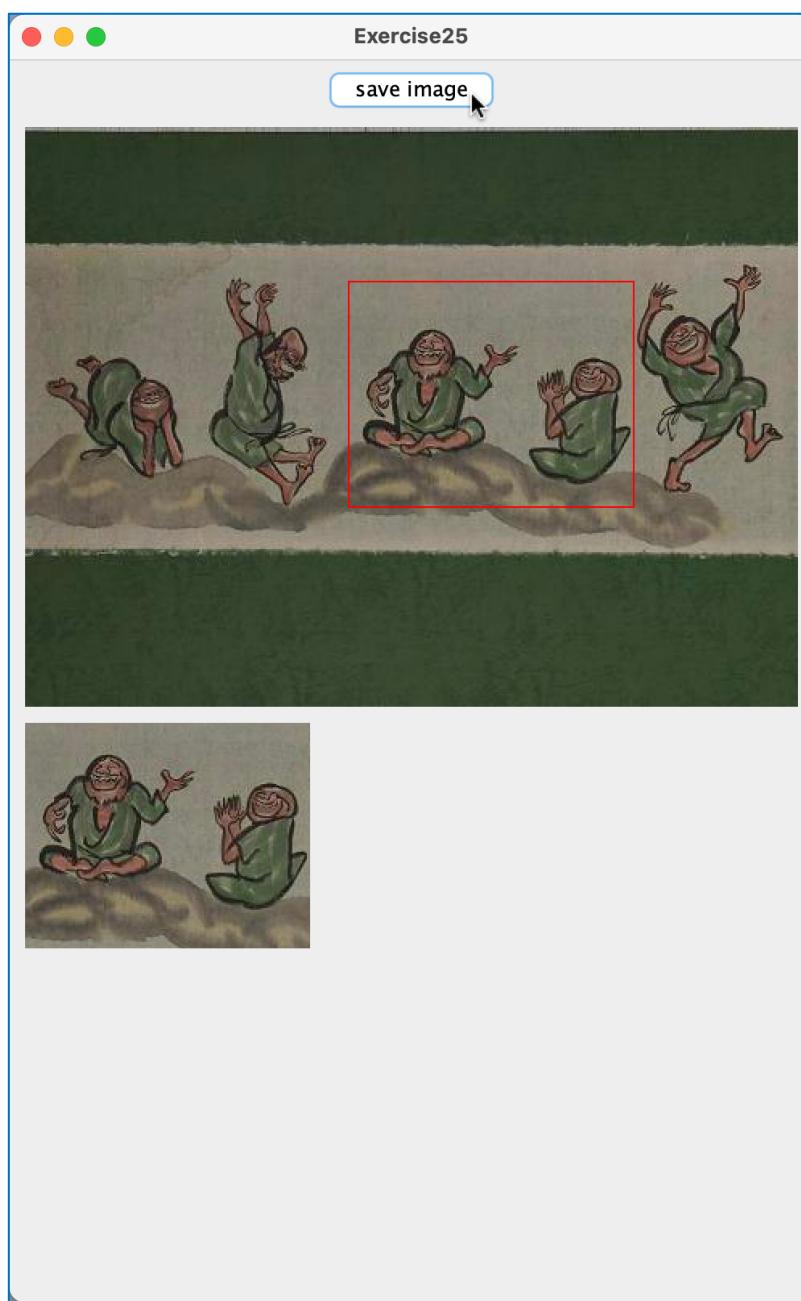


Figure 3. An execution example of Exercise25.java



Figure 4. The obtained image.