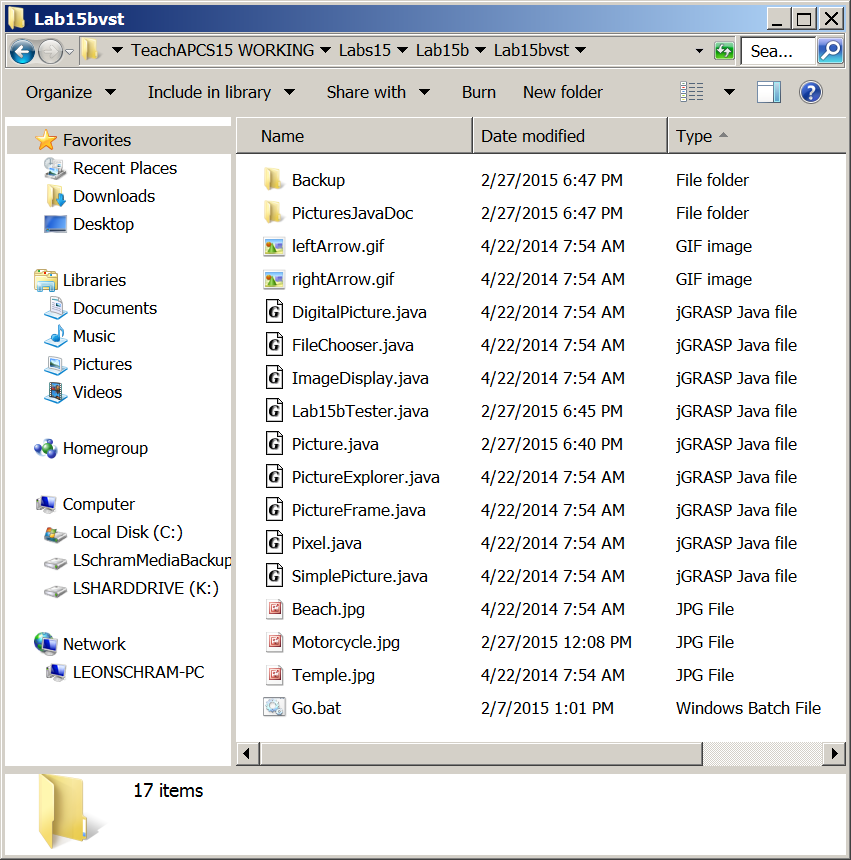
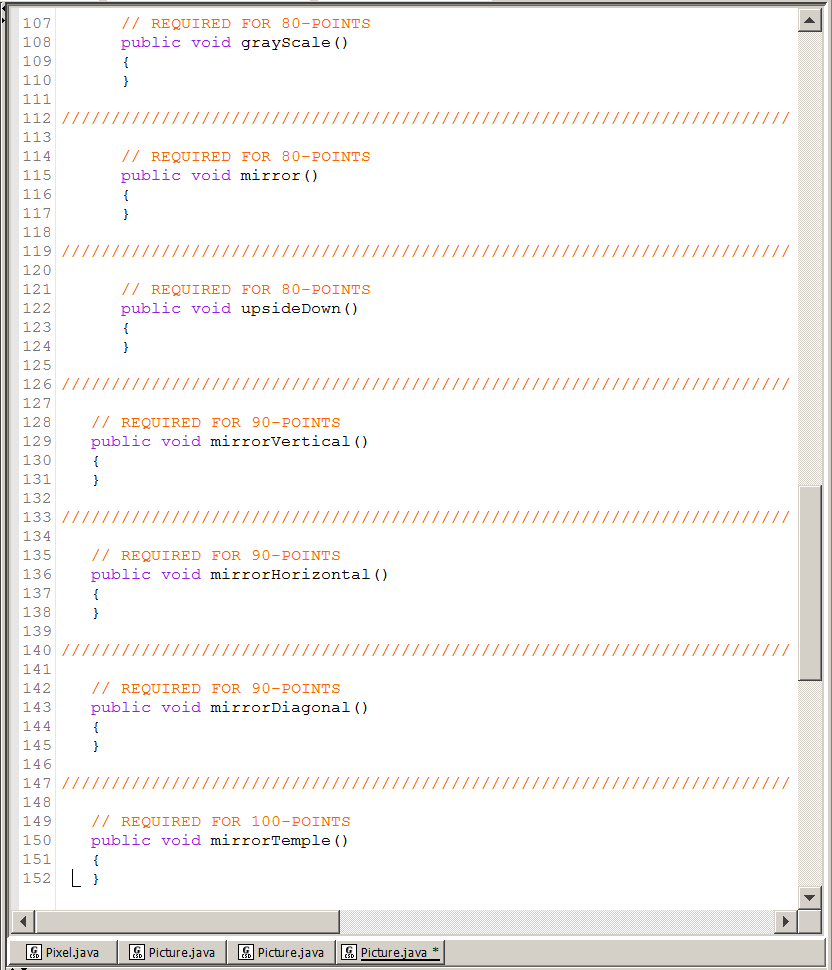
|  |  |
| --- | --- |
| **AP® Computer Science (A)** | **Minor Lab 15b** |
| **The "AP Picture Labs" Program** | **80, 90 & 100 Point Versions** |
| **Assignment Purpose:**  The purpose of this program is to manipulate two-dimensional static arrays to alter digital images. | |

During Chapter XV you have looked at a variety of Lab Experiments that involved the AP Picture Labs. This lab assignment requires knowledge you learned while doing the experiments. Start by looking at the **Lab15bvst** folder. The folder contains many files. They are mostly **\*.java files**, but there are also several picture **\*.jpg** and **\*.gif** files. You have seen most of these files when you did the lab experiments for Chapter XV. The **PicturesJavaDoc** folder provides the web page references with information about all the classes and methods in the AP picture Labs.



This is a team lab assignment. You and your partner will work with only two files. The first one is **Lab15bTester.java**. This file you use when you test your lab assignment and is not to be altered, except for the comments in the main method to select the 80, 90 or 100 point version of Lab15b.

The real work will be done in the **Picture.java** file. This class contains a set of methods that all have empty bodies. You need to complete each method to the specifications that follow. The starting file contains 7 methods that need to be written. The empty methods bodies are shown below along with the comments indicating which method is required for different lab assignment versions.



|  |  |
| --- | --- |
| **80 Point Version Specifics and Output**  In writing the 7 methods you need to use the JavaDoc web pages for various classes and their methods to know what capabilities are available. All these methods are in the **Picture** class, but you will use many methods from other classes in writing the required 7 methods. Every method execution starts with displaying the original picture that will be manipulated. The 80-Point methods do not stand alone. You will need to execute these three methods for the 90-Point Version and the 100-Point Version as well. | |
| Original Picture | Method **grayScale**  Method **grayScale** alters the **beach.jpg** image by changing every colored pixel to a gray shade that is equivalent to the average value of the colored pixels. |
| Method mirror  Method **mirror** alters the **beach.jpg** image- which is now a *black&white image* - to its mirror image. | Method upsideDown  Method **upsideDown** alters the **beach.jpg** image - which is now a *black&white mirror image* - upside down. |
| **90 Point Version Specifics and Output**  This version requires three **mirror** methods that each mirrors half the image around a pivot, mirror line. The testing program has been written such that each method works with an original image of the **motorcycle.jpg** file, unlike the previous 80-Point version, which used the altered images. | |
| Original Picture | Method **mirrorHorizontal**  Method **mirrorHorizontal** alters **motorcycle.jpg** by copying a mirror image of the top-half on the bottom-half. |
| Method **mirrorVertical**  Method **mirrorVertical** alters **motorcycle.jpg** by copying a mirror image of the left-half on the right-half. | Method **mirrorDiagonal**  Method **mirrorDiagonal** alters **motorcycle.jpg** by copying a mirror image across a diagonal line. |
| **100 Point Version Specifics and Output**  A single additional method, **mirrorTemple**, is required for the 100-Point Version. It may appear that this is simply a repeat of the **mirrorVertical** method. Method **mirrorTemple** certainly resemble the  mirror image around a vertical pivot, but it is a specialized method.  Look at the images below. The aim is to try and display the temple as it looked before the roof damage. After the roof is repaired, you will still see the person walking in front of the temple. Using method **mirrorVertical** would give an unrealistic look to the repair. Your objective is to only repair the roof and not alter the rest of the picture. | |
| Original Picture | Method **mirrorTemple** |