7. Place your code in the box below. Make sure you include the method header. Recall the name of the method is `appendPicture`, it takes one parameter of type `Picture`, and it returns nothing.

```java
public void appendPicture(Picture newPic) {
    Picture[] picArray = new Picture[this.length + 1];
    for (int i = 0; i < this.length; i++) {
        picArray[i] = this.pictureArray[i];
    }
    Picture picToCopy = new Picture(newPic);
    picArray[picArray.length - 1] = picToCopy;
    this.pictureArray = picArray;
}
```
7. Place your code in the box below. Make sure you include the method header. Recall the name of the method is `appendPicture`, it takes one parameter of type `Picture`, and it returns nothing.

```java
public void appendPicture()
{
    Pixel[] pixelArray = new Picture.getPixels();
    Picture[] pictureArray = new Picture[];
    for (int i = 0; i < pixelArray.length(); i++)
    {
        pictureArray[i].setColor(pixelArray.getColor());
    }
}
```
Place your code in the box below. Make sure you include the method header. Recall the name of the method is `appendPicture`, it takes one parameter of type `Picture`, and it returns nothing.

```java
public void appendPicture(Picture picToAdd) {
    Picture[] longArray = new Picture[this.length+1];
    Picture tempPic = new Picture();
    for (int i = 0; i < this.length; i++) {
        tempPic = this[i];
        longArray[i] = tempPic;
    }
    longArray[this.length] = picToAdd;
    this = longArray;
}
```
public void appendPicture(Picture otherPicture)
{
    int length = this.pictureArray.length + 1;
    Picture[] newPicArray = new Picture[length];
    for (int i = 0; i < this.pictureArray.length; i++)
    {
        newPicArray[i] = this.pictureArray[i];
    }
    newPicArray[length-1] = otherPicture;
    for (int y = 0; y < newPicArray.length; y++)
    {
        this.picArray[y] = newPicArray[y];
    }
}
Place your code in the box below. Make sure you include the method header. Recall the name of the method is `appendPicture`, it takes one parameter of type `Picture`, and it returns nothing.

```java
public void appendPicture(Picture append) {
    Picture pic = new Picture(append);
    pictureArray = new Picture[1];
    pictureArray1 = new Picture[2];
    pictureArray2 = append;
    i = pictureArray2[0];
    for (i = pictureArray2[0]; i < pictureArray2.length; i++) {
        pic.set(pic.setValue(
```
7. Place your code in the box below. Make sure you include the method header. Recall the name of the method is `appendPicture`, it takes one parameter of type `Picture`, and it returns nothing.

```java
public void appendPicture (Picture pic)
{
    Picture[] copy = new Picture[pictureArray.length+1];
    for (int i = 0; i < pictureArray.length+1; i++)
    {
        if (i < pictureArray.length)
        {
            copy[i] = pictureArray[i];
        }
        else
        {
            copy[i] = getPixels(pictureArray);
        }
    }
}
```