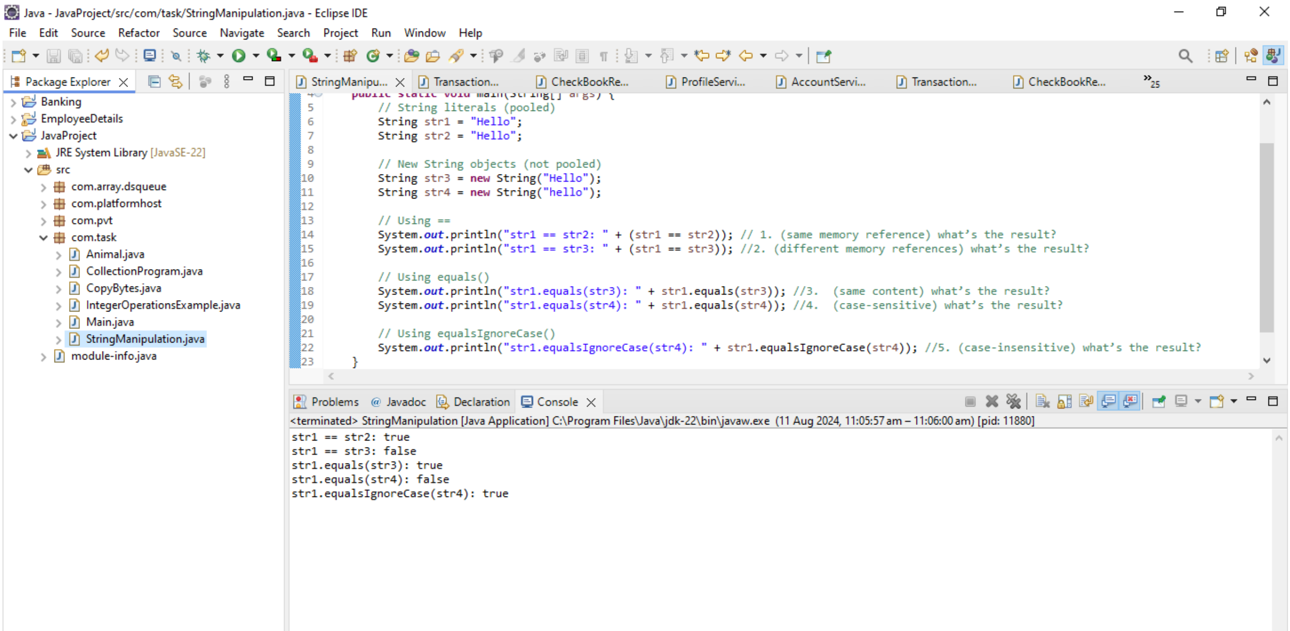
Weekend Assignment in Java

1. Please find case 1 and mention the result for the mentioned statements using strings.

CODE:

```
public class StringComparisonExample {
   public static void main(String[] args) {
      // String literals (pooled)
      String str1 = "Hello";
      String str2 = "Hello";

      // New String objects (not pooled)
      String str3 = new String("Hello");
      String str4 = new String("hello");

      // Using ==
      System.out.println("str1 == str2: " + (str1 == str2)); // 1. (same memory
reference) what’s the result?
      System.out.println("str1 == str3: " + (str1 == str3)); //2. (different memory
references) what’s the result?

      // Using equals()
      System.out.println("str1.equals(str3): " + str1.equals(str3)); //3.  (same content)
what’s the result?
      System.out.println("str1.equals(str4): " + str1.equals(str4)); //4.  (case-sensitive)
what’s the result?

      // Using equalsIgnoreCase()
      System.out.println("str1.equalsIgnoreCase(str4): " +
str1.equalsIgnoreCase(str4)); //5. (case-insensitive) what’s the result?
   }
}
```

OUTPUT:



```java
    public static void main(String[] args) {
        // String literals (pooled)
        String str1 = "Hello";
        String str2 = "Hello";

        // New String objects (not pooled)
        String str3 = new String("Hello");
        String str4 = new String("hello");

        // Using ==
        System.out.println("str1 == str2: " + (str1 == str2)); // 1. (same memory reference) what's the result?
        System.out.println("str1 == str3: " + (str1 == str3)); //2. (different memory references) what's the result?

        // Using equals()
        System.out.println("str1.equals(str3): " + str1.equals(str3)); //3.  (same content) what's the result?
        System.out.println("str1.equals(str4): " + str1.equals(str4)); //4.  (case-sensitive) what's the result?

        // Using equalsIgnoreCase()
        System.out.println("str1.equalsIgnoreCase(str4): " + str1.equalsIgnoreCase(str4)); //5. (case-insensitive) what's the result?
    }
```

```
<terminated> StringManipulation [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe  (11 Aug 2024, 11:05:57 am – 11:06:00 am) [pid: 11880]
str1 == str2: true
str1 == str3: false
str1.equals(str3): true
str1.equals(str4): false
str1.equalsIgnoreCase(str4): true
```

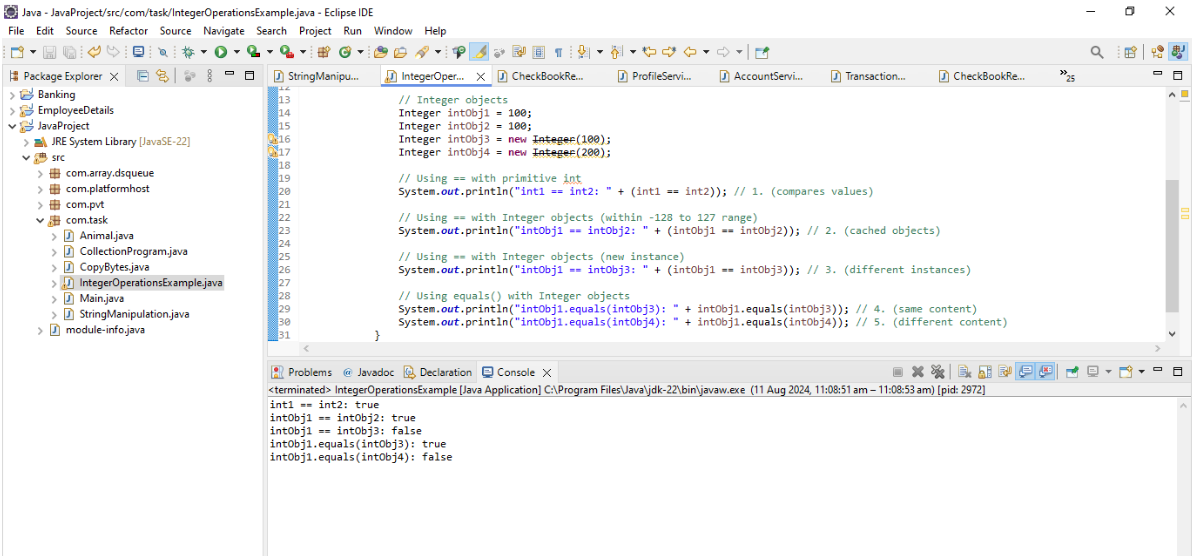
2. Find case 2 and mention the result for the statements using integers.

CODE:

```
public class IntegerComparisonExample {
  public static void main(String[] args) {
//Mention what's the result in 1, 2, 3,4 and 5
      // Primitive int
      int int1 = 100;
      int int2 = 100;
      // Integer objects
      Integer intObj1 = 100;
      Integer intObj2 = 100;
      Integer intObj3 = new Integer(100);
      Integer intObj4 = new Integer(200);

      // Using == with primitive int
      System.out.println("int1 == int2: " + (int1 == int2)); // 1. (compares values)

      // Using == with Integer objects (within -128 to 127 range)
      System.out.println("intObj1 == intObj2: " + (intObj1 == intObj2)); // 2. (cached
objects)

      // Using == with Integer objects (new instance)
```

```
        System.out.println("intObj1 == intObj3: " + (intObj1 == intObj3)); // 3. (different
instances)

        // Using equals() with Integer objects
        System.out.println("intObj1.equals(intObj3): " + intObj1.equals(intObj3)); // 4.
(same content)
        System.out.println("intObj1.equals(intObj4): " + intObj1.equals(intObj4)); // 5.
(different content)
    }
}
```

OUTPUT:



```
        // Integer objects
        Integer intObj1 = 100;
        Integer intObj2 = 100;
        Integer intObj3 = new Integer(100);
        Integer intObj4 = new Integer(200);

        // Using == with primitive int
        System.out.println("int1 == int2: " + (int1 == int2)); // 1. (compares values)

        // Using == with Integer objects (within -128 to 127 range)
        System.out.println("intObj1 == intObj2: " + (intObj1 == intObj2)); // 2. (cached objects)

        // Using == with Integer objects (new instance)
        System.out.println("intObj1 == intObj3: " + (intObj1 == intObj3)); // 3. (different instances)

        // Using equals() with Integer objects
        System.out.println("intObj1.equals(intObj3): " + intObj1.equals(intObj3)); // 4. (same content)
        System.out.println("intObj1.equals(intObj4): " + intObj1.equals(intObj4)); // 5. (different content)
    }
```

```
<terminated> IntegerOperationsExample [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe  (11 Aug 2024, 11:08:51 am – 11:08:53 am) [pid: 2972]
int1 == int2: true
intObj1 == intObj2: true
intObj1 == intObj3: false
intObj1.equals(intObj3): true
intObj1.equals(intObj4): false
```

3. Find case 3 and mention how Basic I/O resources are getting closed and the difference that you implemented earlier in the code – copyBytes.java

CODE:

```
import java.io.BufferedReader;
import java.io.FileReader;
import java.io.IOException;

public class TryWithResourcesExample {
//Eliminating finally block to close resources.

    public static void main(String[] args) {
        // File path (adjust the path as needed)
        String filePath = "example.txt";

        // Traditional try-with-resources block
        try (BufferedReader reader = new BufferedReader(new FileReader(filePath))) {
            String line;
            while ((line = reader.readLine()) != null) {
                System.out.println(line);
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```

ANSWER:

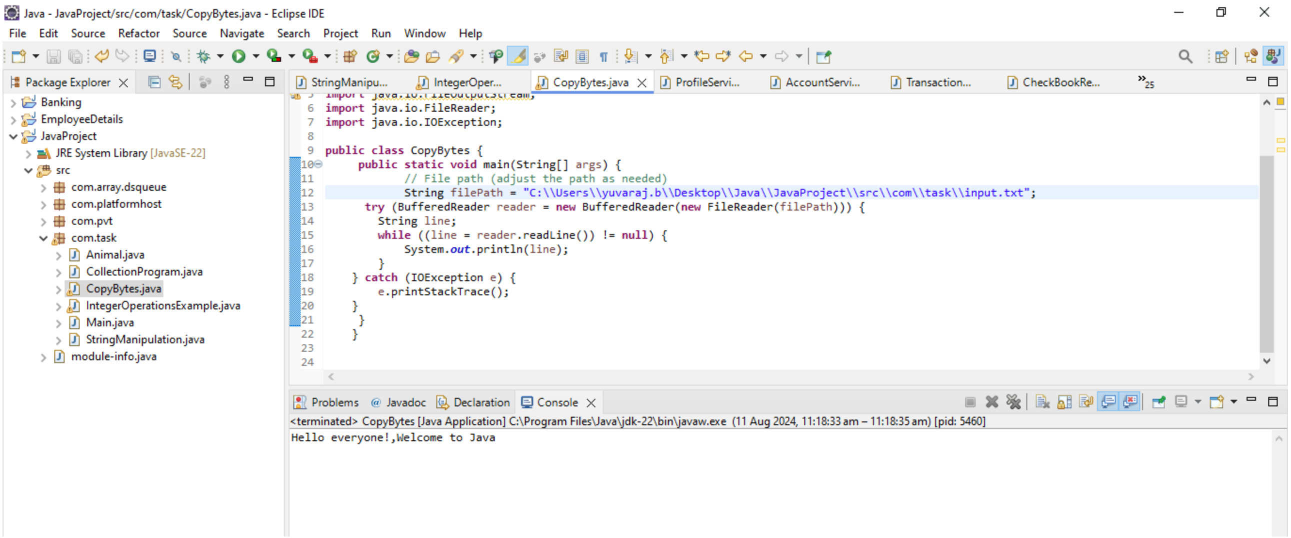
Automatic Resource Management:

The BufferedReader and FileReader resources are automatically closed when the try-with-resources block is exited. You do not need a finally block to close these resources manually, which is the traditional approach.

Enhanced Code (Using try-with-resources):

```
import java.io.BufferedReader;
import java.io.FileReader;
import java.io.IOException;

public class TryWithResourcesExample {
//Eliminating finally block to close resources.

    public static void main(String[] args) {
        // File path (adjust the path as needed)
        String filePath = "example.txt";
  try (BufferedReader reader = new BufferedReader(new FileReader(filePath))) {
    String line;
    while ((line = reader.readLine()) != null) {
        System.out.println(line);
    }
} catch (IOException e) {
    e.printStackTrace();
}
 }
}
```

OUTPUT:


Java - JavaProject/src/com/task/CopyBytes.java - Eclipse IDE
File Edit Source Refactor Source Navigate Search Project Run Window Help
Package Explorer
Banking
EmployeeDetails
JavaProject
JRE System Library [JavaSE-22]
src
com.array.dsqueue
com.platformhost
com.pvt
com.task
Animal.java
CollectionProgram.java
CopyBytes.java
IntegerOperationsExample.java
Main.java
StringManipulation.java
module-info.java
StringManipu... IntegerOper... CopyBytes.java ProfileServi... AccountServi... Transaction... CheckBookRe...
Problems @ Javadoc Declaration Console

```
import java.io.FileOutputStream;
import java.io.FileReader;
import java.io.IOException;

public class CopyBytes {
    public static void main(String[] args) {
            // File path (adjust the path as needed)
            String filePath = "C:\\Users\\yuvaraj.b\\Desktop\\Java\\JavaProject\\src\\com\\task\\input.txt";
     try (BufferedReader reader = new BufferedReader(new FileReader(filePath))) {
       String line;
       while ((line = reader.readLine()) != null) {
           System.out.println(line);
       }
   } catch (IOException e) {
       e.printStackTrace();
   }
    }
   }
```

```
<terminated> CopyBytes [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe  (11 Aug 2024, 11:18:33 am – 11:18:35 am) [pid: 5460]
Hello everyone!,Welcome to Java
```

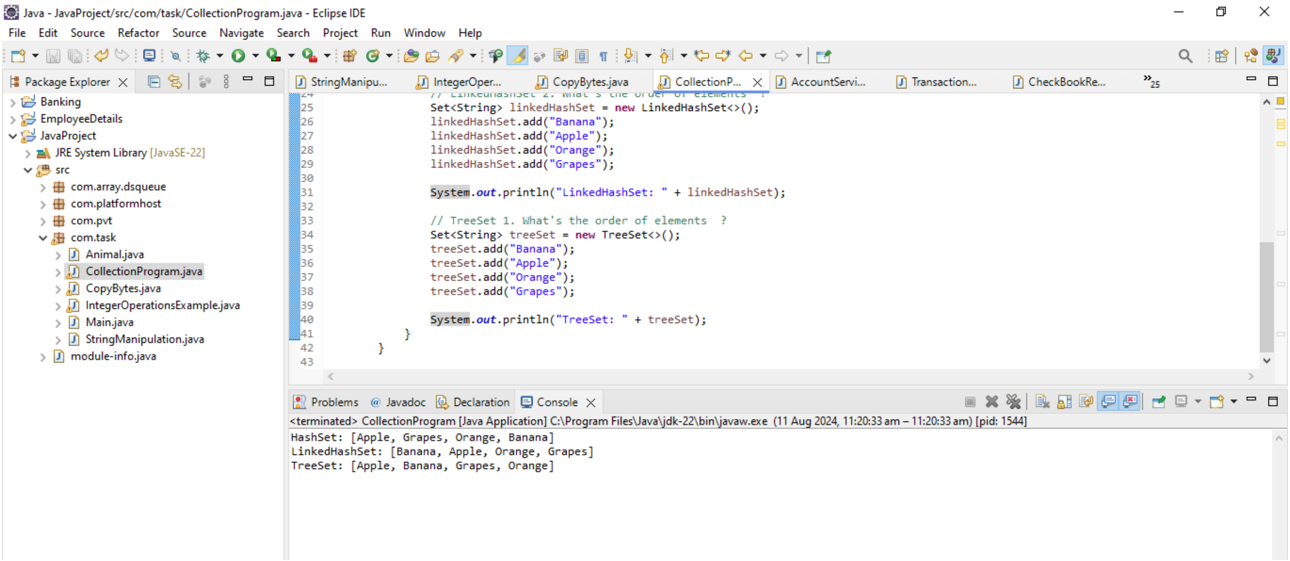
4.Find case 4 and mention the order for 1,2 and 3 using collections

CODE:

```
import java.util.HashSet;
import java.util.LinkedHashSet;
import java.util.Set;
import java.util.TreeSet;
public class SetExample {
    public static void main(String[] args) {
        // Set 1. What's the order of elements?
        Set<String> hashSet = new HashSet<>();
        hashSet.add("Banana");
        hashSet.add("Apple");
        hashSet.add("Orange");
        hashSet.add("Grapes");

        System.out.println("HashSet: " + hashSet);

        // LinkedHashSet 2. What's the order of elements  ?
        Set<String> linkedHashSet = new LinkedHashSet<>();
        linkedHashSet.add("Banana");
        linkedHashSet.add("Apple");
        linkedHashSet.add("Orange");
        linkedHashSet.add("Grapes");

        System.out.println("LinkedHashSet: " + linkedHashSet);
```

```
            // TreeSet 1. What's the order of elements  ?
            Set<String> treeSet = new TreeSet<>();
            treeSet.add("Banana");
            treeSet.add("Apple");
            treeSet.add("Orange");
            treeSet.add("Grapes");

            System.out.println("TreeSet: " + treeSet);
        }
    }
```

OUTPUT:



```
        Set<String> linkedHashSet = new LinkedHashSet<>();
        linkedHashSet.add("Banana");
        linkedHashSet.add("Apple");
        linkedHashSet.add("Orange");
        linkedHashSet.add("Grapes");

        System.out.println("LinkedHashSet: " + linkedHashSet);

        // TreeSet 1. What's the order of elements  ?
        Set<String> treeSet = new TreeSet<>();
        treeSet.add("Banana");
        treeSet.add("Apple");
        treeSet.add("Orange");
        treeSet.add("Grapes");

        System.out.println("TreeSet: " + treeSet);
    }
}
```

```
<terminated> CollectionProgram [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe  (11 Aug 2024, 11:20:33 am – 11:20:33 am) [pid: 1544]
HashSet: [Apple, Grapes, Orange, Banana]
LinkedHashSet: [Banana, Apple, Orange, Grapes]
TreeSet: [Apple, Banana, Grapes, Orange]
```