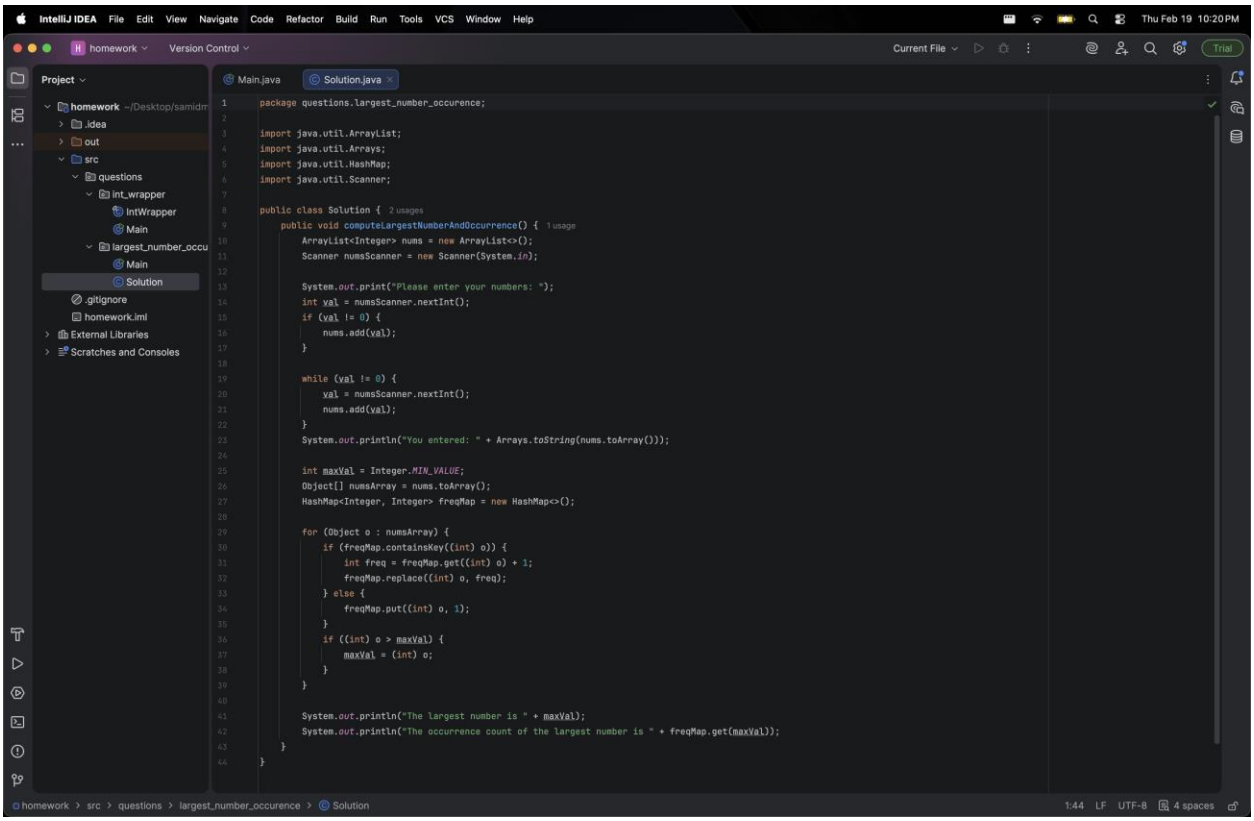
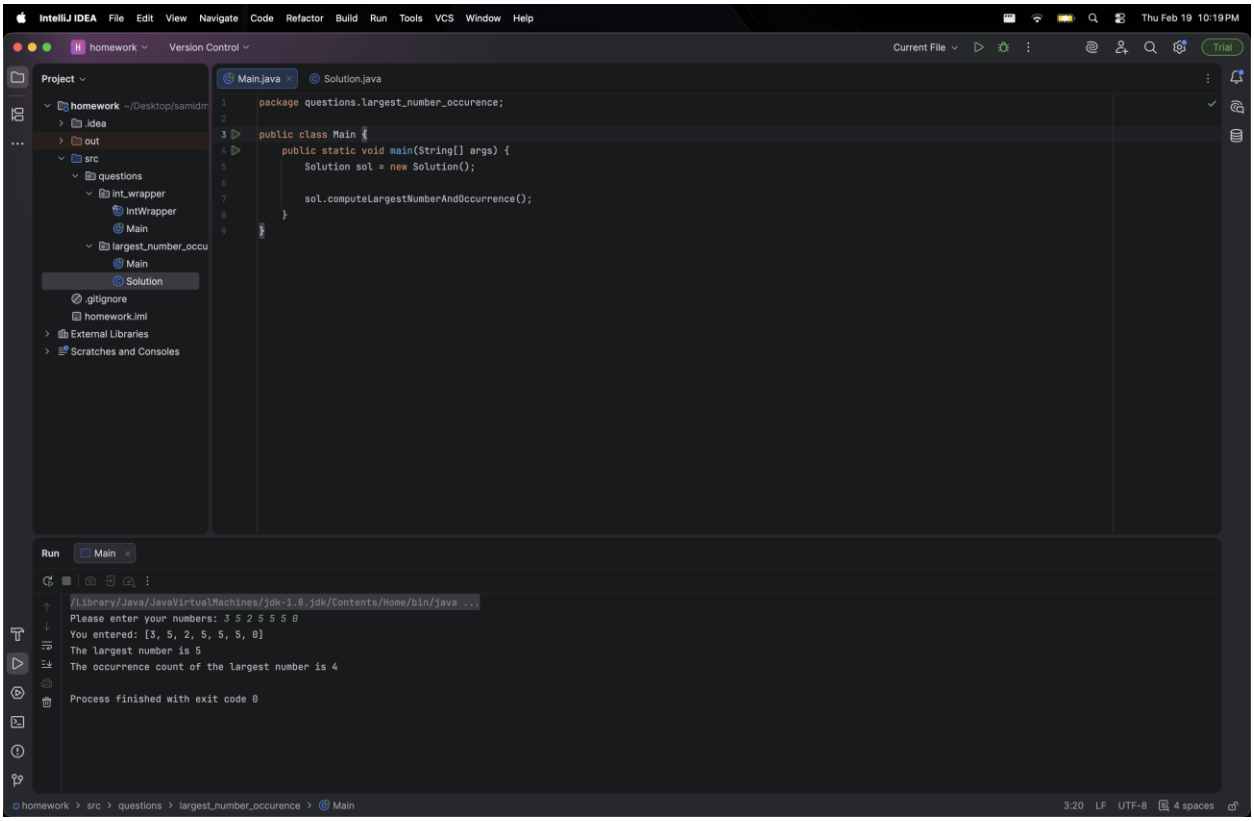
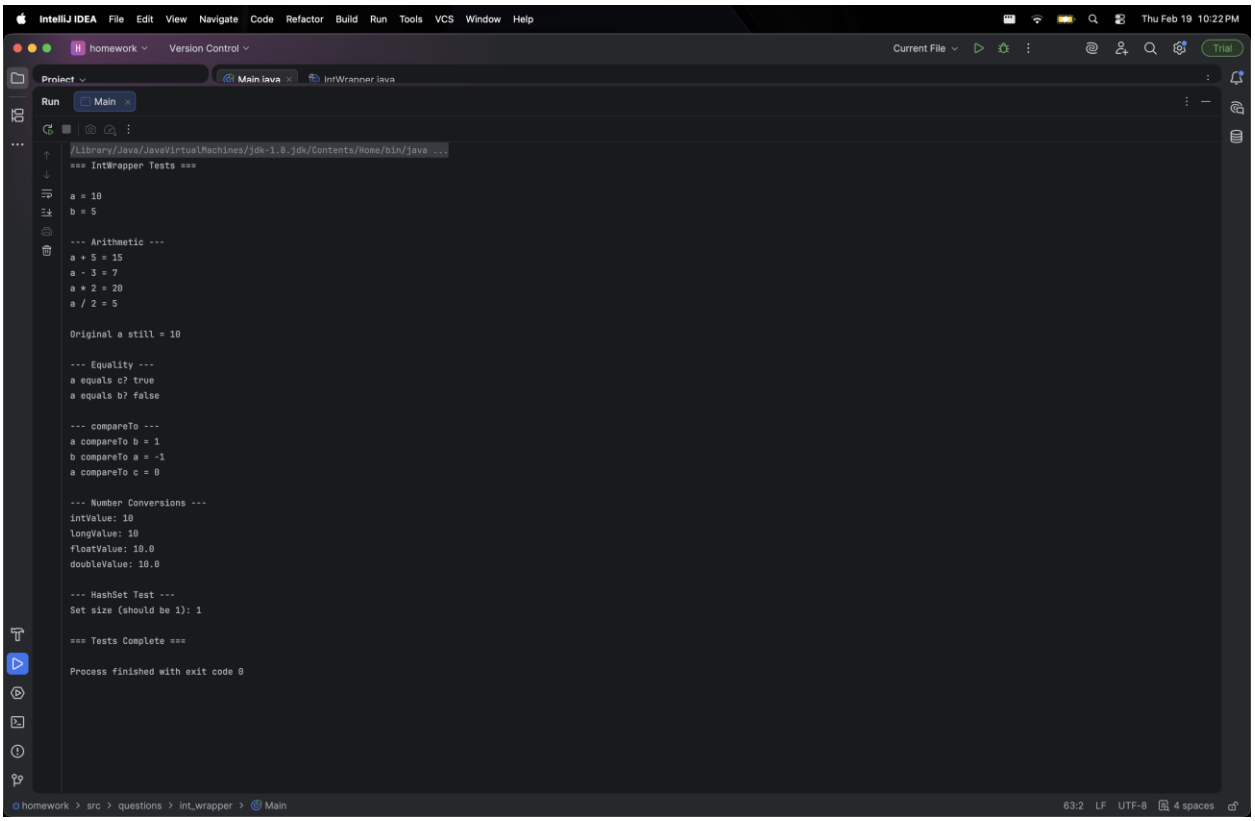
Week 1 – Day 2 Assignment (February 19th, 2026)

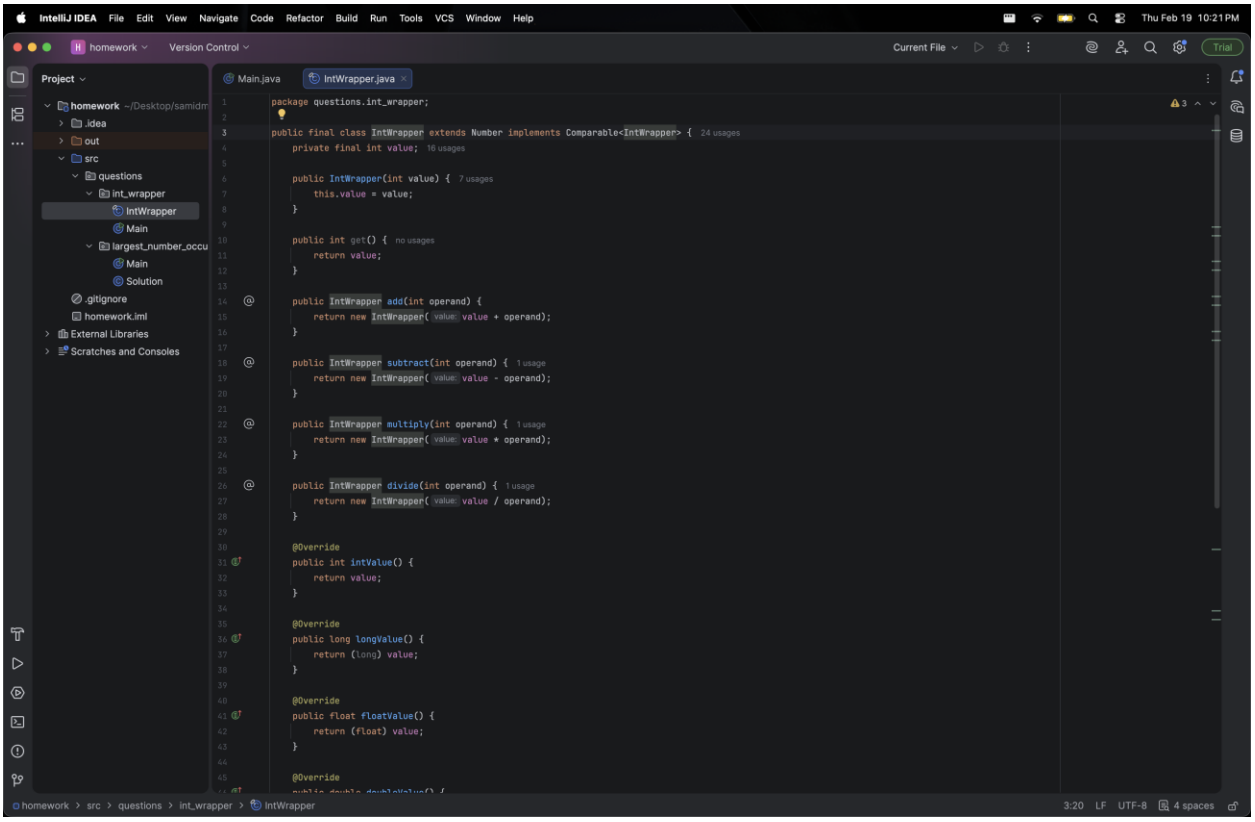
Question 1 – Occurrence of Largest Number

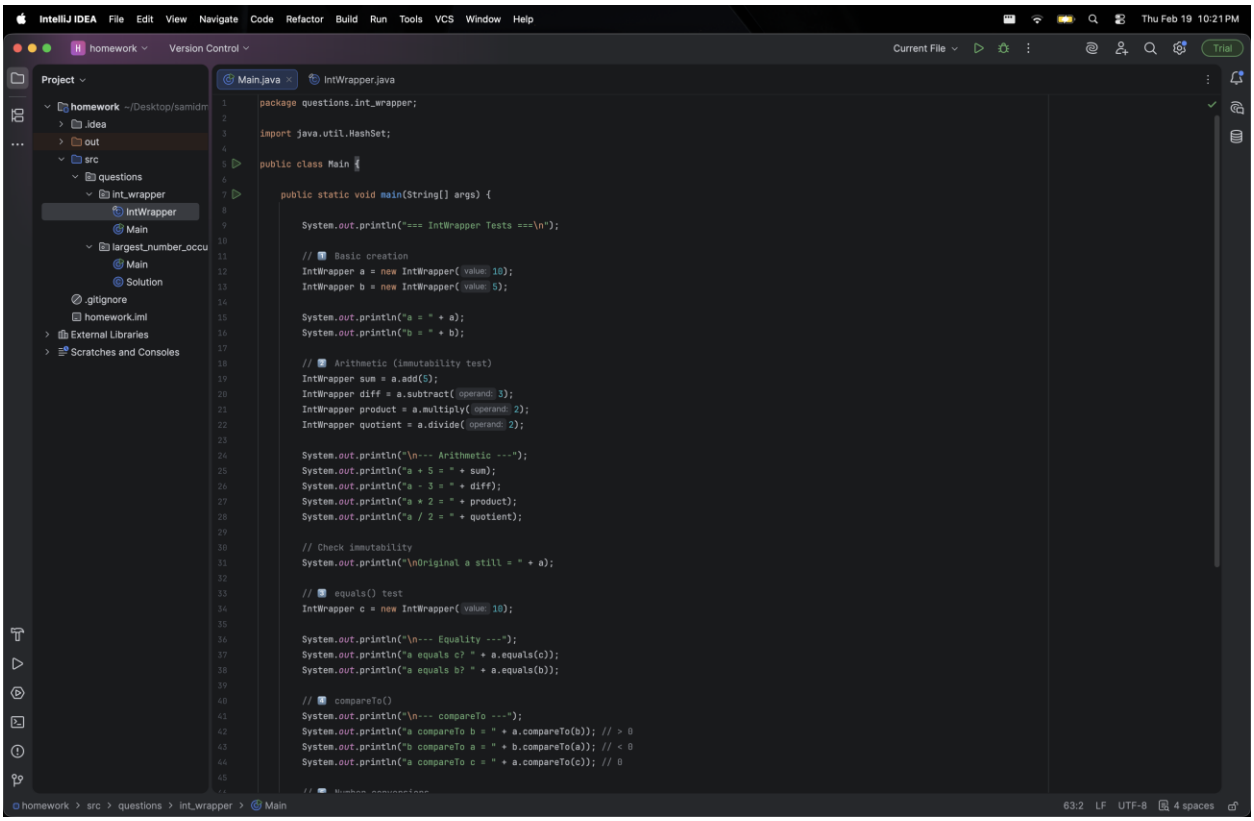

IntelliJ IDEA File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
Thu Feb 19 10:20 PM
homework Version Control
Current File
Trial
Project
homework ~/Desktop/samidm
.idea
out
src
questions
int_wrapper
IntWrapper
Main
largest_number_occu
Main
Solution
.gitignore
homework.iml
External Libraries
Scratches and Consoles
Main.java
Solution.java
package questions.largest_number_occurence;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.HashMap;
import java.util.Scanner;
public class Solution { 2 usages
public void computeLargestNumberAndOccurrence() { 1 usage
ArrayList<Integer> nums = new ArrayList<>();
Scanner numsScanner = new Scanner(System.in);
System.out.print("Please enter your numbers: ");
int val = numsScanner.nextInt();
if (val != 0) {
nums.add(val);
}
while (val != 0) {
val = numsScanner.nextInt();
nums.add(val);
}
System.out.println("You entered: " + Arrays.toString(nums.toArray()));
int maxVal = Integer.MIN_VALUE;
Object[] numsArray = nums.toArray();
HashMap<Integer, Integer> freqMap = new HashMap<>();
for (Object o : numsArray) {
if (freqMap.containsKey((int) o)) {
int freq = freqMap.get((int) o) + 1;
freqMap.replace((int) o, freq);
} else {
freqMap.put((int) o, 1);
}
if ((int) o > maxVal) {
maxVal = (int) o;
}
}
System.out.println("The largest number is " + maxVal);
System.out.println("The occurrence count of the largest number is " + freqMap.get(maxVal));
}
}
homework > src > questions > largest_number_occurence > Solution
1:44 LF UTF-8 4 spaces


IntelliJ IDEA File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
Thu Feb 19 10:19 PM
homework Version Control
Current File
Trial
Project
homework ~/Desktop/samidm
.idea
out
src
questions
int_wrapper
IntWrapper
Main
largest_number_occu
Main
Solution
.gitignore
homework.iml
External Libraries
Scratches and Consoles
Main.java
Solution.java
package questions.largest_number_occurence;
public class Main {
public static void main(String[] args) {
Solution sol = new Solution();
sol.computeLargestNumberAndOccurrence();
}
}
Run
Main
/Library/Java/JavaVirtualMachines/jdk-1.8.jdk/Contents/Home/bin/java ...
Please enter your numbers: 3 5 2 5 5 5 0
You entered: [3, 5, 2, 5, 5, 5, 0]
The largest number is 5
The occurrence count of the largest number is 4
Process finished with exit code 0
homework > src > questions > largest_number_occurence > Main
3:20 LF UTF-8 4 spaces

Question 2 – Custom int Wrapper Class


IntelliJ IDEA File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
Thu Feb 19 10:22 PM
homework Version Control
Current File
Trial
Project
Main.java
IntWrapper.java
Run
Main
/Library/Java/JavaVirtualMachines/jdk-1.8.jdk/Contents/Home/bin/java ...
=== IntWrapper Tests ===
a = 10
b = 5
--- Arithmetic ---
a + 5 = 15
a - 3 = 7
a * 2 = 20
a / 2 = 5
Original a still = 10
--- Equality ---
a equals c? true
a equals b? false
--- compareTo ---
a compareTo b = 1
b compareTo a = -1
a compareTo c = 0
--- Number Conversions ---
intValue: 10
longValue: 10
floatValue: 10.0
doubleValue: 10.0
--- HashSet Test ---
Set size (should be 1): 1
=== Tests Complete ===
Process finished with exit code 0
homework > src > questions > int_wrapper > Main
63:2 LF UTF-8 4 spaces


IntelliJ IDEA File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
Thu Feb 19 10:21 PM
homework Version Control
Current File
Trial
Project
homework ~/Desktop/samidm
.idea
out
src
questions
int_wrapper
IntWrapper
Main
largest_number_occu
Main
Solution
.gitignore
homework.iml
External Libraries
Scratches and Consoles
Main.java
IntWrapper.java
package questions.int_wrapper;
public final class IntWrapper extends Number implements Comparable<IntWrapper> { 24 usages
private final int value; 16 usages
public IntWrapper(int value) { 7 usages
this.value = value;
}
public int get() { no usages
return value;
}
public IntWrapper add(int operand) {
return new IntWrapper(value: value + operand);
}
public IntWrapper subtract(int operand) { 1 usage
return new IntWrapper(value: value - operand);
}
public IntWrapper multiply(int operand) { 1 usage
return new IntWrapper(value: value * operand);
}
public IntWrapper divide(int operand) { 1 usage
return new IntWrapper(value: value / operand);
}
@Override
public int intValue() {
return value;
}
@Override
public long longValue() {
return (long) value;
}
@Override
public float floatValue() {
return (float) value;
}
@Override
homework > src > questions > int_wrapper > IntWrapper
3:20 LF UTF-8 4 spaces


IntelliJ IDEA File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
Thu Feb 19 10:21 PM
homework Version Control
Current File
Trial
Project
homework ~/Desktop/samidm
.idea
out
src
questions
int_wrapper
IntWrapper
Main
largest_number_occu
Main
Solution
.gitignore
homework.iml
External Libraries
Scratches and Consoles
Main.java
IntWrapper.java
package questions.int_wrapper;
import java.util.HashSet;
public class Main {
public static void main(String[] args) {
System.out.println("=== IntWrapper Tests ===\n");
// Basic creation
IntWrapper a = new IntWrapper(value: 10);
IntWrapper b = new IntWrapper(value: 5);
System.out.println("a = " + a);
System.out.println("b = " + b);
// Arithmetic (immutability test)
IntWrapper sum = a.add(5);
IntWrapper diff = a.subtract(operand: 3);
IntWrapper product = a.multiply(operand: 2);
IntWrapper quotient = a.divide(operand: 2);
System.out.println("\n--- Arithmetic ---");
System.out.println("a + 5 = " + sum);
System.out.println("a - 3 = " + diff);
System.out.println("a * 2 = " + product);
System.out.println("a / 2 = " + quotient);
// Check immutability
System.out.println("\nOriginal a still = " + a);
// equals() test
IntWrapper c = new IntWrapper(value: 10);
System.out.println("\n--- Equality ---");
System.out.println("a equals c? " + a.equals(c));
System.out.println("a equals b? " + a.equals(b));
// compareTo()
System.out.println("\n--- compareTo ---");
System.out.println("a compareTo b = " + a.compareTo(b)); // > 0
System.out.println("b compareTo a = " + b.compareTo(a)); // < 0
System.out.println("a compareTo c = " + a.compareTo(c)); // 0
homework > src > questions > int_wrapper > Main
63:2 LF UTF-8 4 spaces