Task 1: Convert Primitive to Wrapper and Vice Versa

```
public class Task1 {
    public static void main(String[] args) {
        int a = 10;
        double d = 25.67;
        char c = 'Z';

        // Primitive to Wrapper
        Integer intObj = Integer.valueOf(a);
        Double doubleObj = Double.valueOf(d);
        Character charObj = Character.valueOf(c);

        // Wrapper to Primitive
        int a2 = intObj.intValue();
        double d2 = doubleObj.doubleValue();
        char c2 = charObj.charValue();

        System.out.println("Primitive to Wrapper: " + intObj + ", " +
doubleObj + ", " + charObj);
        System.out.println("Wrapper to Primitive: " + a2 + ", " + d2 + ", " +
c2);
    }
}
```

Task 2: Autoboxing and Unboxing

```
import java.util.*;

public class Task2 {
    public static void main(String[] args) {
        List<Double> marks = new ArrayList<>();
        marks.add(85.5);  // autoboxing
        marks.add(90.0);
        marks.add(78.25);

        double sum = 0;
        for (double m : marks) { // unboxing
            sum += m;
        }
        double avg = sum / marks.size();

        System.out.println("Marks: " + marks);
        System.out.println("Average: " + avg);
    }
}
```

Task 3: Parsing from Strings

```
public class Task3 {
    public static void main(String[] args) {
        String intStr = "123";
        String dblStr = "45.67";
        String boolStr = "true";

        int i = Integer.parseInt(intStr);
        double d = Double.parseDouble(dblStr);
        boolean b = Boolean.parseBoolean(boolStr);

        System.out.println("int: " + i);
        System.out.println("double: " + d);
        System.out.println("boolean: " + b);
    }
}
```

Task 4: Comparing Wrapper Objects

```
public class Task4 {
    public static void main(String[] args) {
        Integer a = 100;           // autoboxing
        Integer b = new Integer(100); // explicit wrapper

        System.out.println("== result: " + (a == b)); // false (reference
comparison)
        System.out.println(".equals() result: " + a.equals(b)); // true
(value comparison)
    }
}
```

Task 5: Wrapper Classes in Generics

```
class Box<T> {
    T value;

    void set(T val) {
        this.value = val;
    }

    T get() {
        return value;
    }
}
```

```
public class Task5 {
    public static void main(String[] args) {
        Box<Integer> intBox = new Box<>();
        intBox.set(50); // autoboxing
        int x = intBox.get(); // unboxing

        Box<Double> dblBox = new Box<>();
        dblBox.set(99.99);

        System.out.println("Integer Value: " + x);
        System.out.println("Double Value: " + dblBox.get());
    }
}
```

Task 6: Wrapper with ArrayList (Grades Tracker)

```
import java.util.*;

public class Task6 {
    public static void main(String[] args) {
        ArrayList<Integer> grades = new ArrayList<>(Arrays.asList(85, 75, 90,
60, 88));

        Collections.sort(grades);
        grades.remove(0); // remove lowest

        int max = Collections.max(grades);
        int min = Collections.min(grades);

        int sum = 0;
        for (int g : grades) sum += g;

        System.out.println("Grades: " + grades);
        System.out.println("Max: " + max + ", Min: " + min + ", Avg: " + (sum
/ grades.size()));
    }
}
```

Task 7: Temperature Converter App

```
import java.util.*;

public class Task7 {
    public static void main(String[] args) {
        String[] temps = {"36", "38", "40"};
        ArrayList<Double> fahrenheitTemps = new ArrayList<>();

        for (String t : temps) {
            int celsius = Integer.parseInt(t);
            double f = (celsius * 9.0 / 5) + 32;
            fahrenheitTemps.add(f);
        }
        System.out.println("Fahrenheit Temps: " + fahrenheitTemps);
    }
}
```

Task 8: Bank Account Using Wrapper Class

```
class BankAccount {
    private String accNo;
    private Double balance;

    public BankAccount(String accNo) {
        this.accNo = accNo;
        this.balance = 0.0; // Handle null by default
    }

    public void deposit(double amount) {
        balance += amount;
    }

    public void withdraw(double amount) {
        if (balance >= amount)
            balance -= amount;
        else
            System.out.println("Insufficient funds");
    }

    public void showBalance() {
        System.out.println("Balance: ₹" + balance);
    }
}

public class Task8 {
    public static void main(String[] args) {
        BankAccount acc = new BankAccount("AC12345");
        acc.deposit(5000);
        acc.withdraw(1200);
        acc.showBalance();
    }
}
```

Task 9: Null Handling in Wrapper Classes

```
public class Task9 {
    public static void main(String[] args) {
        Double salary = null;

        try {
            double net = salary + 1000; // NullPointerException
            System.out.println("Net salary: " + net);
        } catch (NullPointerException e) {
            System.out.println("Salary is null. Using default value.");
            salary = 0.0;
            double net = salary + 1000;
            System.out.println("Net salary: " + net);
        }
    }
}
```

Task 10: Wrapper Classes with Sorting

```
import java.util.*;

public class Task10 {
    public static void main(String[] args) {
        Integer[] nums = {20, 10, 40, 30, 50};
        Arrays.sort(nums); // sort ascending

        System.out.println("Sorted: " + Arrays.toString(nums));
        System.out.println("2nd highest: " + nums[nums.length - 2]);
    }
}
```

Student Scoreboard App

```
import java.util.*;

public class Scoreboard {
    public static void main(String[] args) {
        Map<String, List<Integer>> students = new HashMap<>();

        students.put("Alice", Arrays.asList(85, 90, 95));
        students.put("Bob", Arrays.asList(70, 75, 80));
        students.put("Charlie", Arrays.asList(88, 92, 86));

        String topStudent = "";
        double highestAvg = 0;

        for (String name : students.keySet()) {
```

```
            List<Integer> marks = students.get(name);
            int sum = 0;
            for (int mark : marks) sum += mark;
            double avg = sum / (double) marks.size();

            System.out.println(name + " Average: " + avg);

            if (avg > highestAvg) {
                highestAvg = avg;
                topStudent = name;
            }
        }

        System.out.println("Top Scorer: " + topStudent + " with Avg: " +
highestAvg);
    }
}
```