Java Final Mandatory Assignment Solutions

1. Polymorphism with Shape Interface

```
interface Shape {
    double calculateArea();
    double calculatePerimeter();
}

class Circle implements Shape {
    private double radius;
    public Circle(double radius) { this.radius = radius; }
    public double calculateArea() { return Math.PI * radius * radius; }
    public double calculatePerimeter() { return 2 * Math.PI * radius; }
}

class Triangle implements Shape {
    private double base, height, side1, side2;
    public Triangle(double base, double height, double side1, double side2) {
        this.base = base; this.height = height; this.side1 = side1; this.side2 = side2;
    }
    public double calculateArea() { return 0.5 * base * height; }
    public double calculatePerimeter() { return base + side1 + side2; }
}
```

2. Invoking Parent Constructor from Child Class

```
class Parent {
    Parent() { System.out.println("Parent Constructor Invoked"); }
}

class Child extends Parent {
    Child() { super(); System.out.println("Child Constructor Invoked"); }
}

public class Main {
    public static void main(String[] args) {
        Child child = new Child();
    }
}
```

3. Exception Handling for Negative Integer

```
import java.util.Scanner;

public class NegativeNumberException {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int num = sc.nextInt();
        try {
            if (num < 0) throw new IllegalArgumentException("Negative number not allowed!");
            System.out.println("Valid Number: " + num);
        } catch (Exception e) {
            System.out.println("Exception: " + e.getMessage());
        }
    }
}
```

4. Bank Account Simulation

```
class BankAccount {
    private double balance;
    public BankAccount(double initialBalance) { this.balance = initialBalance; }
    public void deposit(double amount) { balance += amount; }
    public void withdraw(double amount) { if (amount <= balance) balance -= amount; else System.out.println("Insufficient Funds!"); }
    public double checkBalance() { return balance; }
}
```

(More solutions will be added for the remaining questions.)