**Answers for Java Jigsaw : Code Reconstruction Challenge**

1. **NumberChecker**

**if** (number > 0)

**else** **if** (number < 0)

1. OOPs – Class & Object

Car c = new Car();

c.model = "Honda";

System.out.println(c.model);

1. Loop And DataType

**for** (**int** i = 1; i <= 5; i++) {

System.***out***.println(i);

1. Constructor

**this**.brand = brand;

C4\_Car myCar = **new** C4\_Car("Toyota");

myCar.drive();

1. Even Or Odd Checker -Scanner

The necessary import statement import java.util.Scanner; is missing.

The scanner.next() method returns a String, not an int. It should be scanner.nextInt()

1. C6\_Circle - Static & Final

**public** **class** C6\_Circle {

// This is the correct way to declare a static final constant.

**public** **static** **final** **double** ***PI*** = 3.14159;

// This is a static method that doesn't require an object to be called.

**public** **static** **double** calculateArea(**double** radius) {

**return** ***PI*** \* radius \* radius;

}

**public** **static** **void** main(String[] args) {

// Now the static method can be called directly using the class name.

**double** area = C6\_Circle.*calculateArea*(10);

System.***out***.println("The area of the circle is: " + area);

}

}

1. C7\_TestPolymorphism Abstract & poly

**abstract** **class** Shape {

**abstract** **void** draw();

}

**class** Circle **extends** Shape {

@Override

**void** draw() {

System.***out***.println("Drawing a Circle");

}

}

**public** **class** C7\_TestPolymorphism {

**public** **static** **void** main(String[] args) {

Shape myShape = **new** Circle();

myShape.draw();

}

}

1. C8\_Company - instanceOf

**class** Employee {}

**class** Manager **extends** Employee {}

**public** **class** C8\_Company {

**public** **static** **void** main(String[] args) {

Employee myPerson = **new** Manager();

**if** (myPerson **instanceof** Manager) {

System.***out***.println("myPerson is a Manager.");

} **else** **if** (myPerson **instanceof** Employee) {

System.***out***.println("myPerson is a regular Employee.");

}

}

}

1. C9\_GradeChecker - Scanner

**public** **class** C9\_GradeChecker {

**public** **static** **void** main(String[] args) {

Scanner scanner = **new** Scanner(System.***in***);

System.***out***.print("Enter your score: ");

**int** score = scanner.nextInt();

**if** (score >= 90) {

System.***out***.println("Your grade is A.");

}

**else** **if** (score >= 80) {

System.***out***.println("Your grade is B.");

}

**else** {

System.***out***.println("Your grade is C.");

}

}

}

1. Inheritance & super - C\_10

**class Shape {**

**int width;**

**// ERROR 4: Logical inconsistency in parameter name.**

**public Shape(int width) {**

**this.width = width;**

**}**

**public void printArea() {**

**System.out.println("Area of shape is not defined.");**

**}**

**}**

**// ERROR 1: Missing 'extends' keyword.**

**class Square extends Shape {**

**// ERROR 3: Variable 'length' is unnecessary and confusing.**

**// The 'width' field from the parent class is sufficient.**

**public Square(int side) {**

**// ERROR 2: The 'super' call is correct, but only after fixing ERROR 1.**

**super(side);**

**}**

**@Override // Good practice to indicate method overriding**

**public void printArea() {**

**// Correctly use the parent's variable to calculate the area.**

**int area = width \* width;**

**System.out.println("Area of square is " + area);**

**}**

**}**

**public class Geometry {**

**public static void main(String[] args) {**

**// Correct polymorphic instantiation**

**Shape myShape = new Square(5);**

**myShape.printArea(); // This will correctly call the overridden method in Square.**

**}**

**}**

11.

public class EvenFinder {

public static void main(String[] args) {

int[] numbers = {2, 4, 6, 8, 10};

for (int i = 0; i < numbers.length; i++) {

if (numbers[i] % 2 == 0) {

System.out.println(numbers[i]);

}

}

}

}