Q76–Q85: Debugging & Predict Output (Functions Focused)

Q76. What will be the output?

```
public class Main {
    public static int calculate(int a, int b) {
        if (b == 0) return 1;
        return a * calculate(a, b - 1);
    }

    public static void main(String[] args) {
        System.out.println(calculate(3, 3));
    }
}
```

Q77. Output prediction with return in loop

```
public class Main {
    public static int mystery(int n) {
        for (int i = 0; i < n; i++) {
            if (i % 2 == 0)
                return i;
        }
        return -1;
    }

    public static void main(String[] args) {
        System.out.println(mystery(5));
    }
}
```

.Q78. What will be the output? Recursive trace

```
public class Main {
    public static int fun(int n) {
        if (n <= 1) return n;
        return fun(n - 1) + fun(n - 2);
    }

    public static void main(String[] args) {
        System.out.println(fun(4));
    }
}
```

Q79. Predict the output with shadowed variable

```
public class Main {
    static int x = 5;

    public static void changeX(int x) {
        x = 10;
    }

    public static void main(String[] args) {
        changeX(x);
        System.out.println(x);
    }
}
```

Q80. Trace return values

```
public class Main {
    public static int compute(int x) {
        if (x == 1)
            return 1;
        return x + compute(x - 1);
    }

    public static void main(String[] args) {
        System.out.println(compute(4));
    }
```

Q81. Output with multiple returns (careful tracing)

```
public class Main {
    public static int sample(int x) {
        if (x < 0) return -1;
        if (x == 0) return 0;
        return sample(x - 1) + x;
    }

    public static void main(String[] args) {
        System.out.println(sample(3));
    }
}
```

Q82. Logical condition with recursion

```
public class Main {
    public static boolean isEven(int n) {
        if (n == 0) return true;
        if (n == 1) return false;
        return isEven(n - 2);
    }

    public static void main(String[] args) {
        System.out.println(isEven(9));
    }
}
```

Q83. Output prediction – variable scope

```
public class Main {
    static int count = 0;

    public static void increment() {
        int count = 10;
        count++;
    }

    public static void main(String[] args) {
        increment();
        System.out.println(count);
    }
}
```

Q84. Recursive multiplication using addition

```
public class Main {
    public static int multiply(int a, int b) {
        if (b == 0) return 0;
        return a + multiply(a, b - 1);
    }

    public static void main(String[] args) {
        System.out.println(multiply(4, 3));
    }
}
```

Q85. Return value overwrite confusion

```
public class Main {

    public static int demo() {
        int x = 5;
        return x;
        // x = 10; // this line won't execute
    }

    public static void main(String[] args) {
        System.out.println(demo());
    }
}
```