CDAC MUMBAI

Lab Assignment

SECTION 1: Error-Driven Learning Assignment: Loop Errors

Instructions:

Analyze each code snippet for errors or unexpected behavior. For each snippet, determine:

1. Why does the error or unexpected behavior occur?
2. How can the code be corrected to achieve the intended behavior?

Snippet 1:

public class InfiniteForLoop { public static void main(String[] args) {

for (int i = 0; i < 10; i--) {

System.out.println(i);

}

} }

// Error to investigate: Why does this loop run infinitely? How should the loop control variable be adjusted?

**Because of decrement condition this loop run infinitely.**

**Instead of i-- write i++.**

**public class InfiniteForLoop {**

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

**for (int i = 0; i < 10; i++) {**

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

**}**

**}**

**}**

Snippet 2:

public class IncorrectWhileCondition { public static void main(String[] args) {

int count = 5; while (count = 0) {

System.out.println(count);

count--;

}

}

}

// Error to investigate: Why does the loop not execute as expected? What is the issue with the condition in the `while` loop?

**This loop is assigning the value as 0 and not comparing the values therefore the loop not execute as expected.**

**In condition write count>0 then it will work.**

**public class Main {**

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

**int count = 5;**

**while (count>0) {**

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

**count--;**

**}**

**}**

**}**

Snippet 3:

public class DoWhileIncorrectCondition { public static void main(String[] args) {

int num = 0;

do {

System.out.println(num);

num++; } while (num > 0); }

}

// Error to investigate: Why does the loop only execute once? What is wrong with the loop condition in the `do while` loop?

**the loop only execute once because in while(0>0) which gets false so it will come out of the loop .**

**Write while(num==0) then it will work.**

**public class Main {**

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

**int num = 0;**

**do {**

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

**num++;**

**} while (num==0);**

**}**

**}**

Snippet 4:

public class OffByOneErrorForLoop { public static void main(String[] args) { for (int i = 1; i <= 10; i++) {

System.out.println(i);

}

// Expected: 10 iterations with numbers 1 to 10

// Actual: Prints numbers 1 to 10, but the task expected only 1 to 9

}

}

// Error to investigate: What is the issue with the loop boundaries? How should the loop be adjusted to meet the expected output?

**Write in condition i<10 then 9 iterations will occur.**

Snippet 5:

public class WrongInitializationForLoop { public static void main(String[] args) { for (int i = 10; i >= 0; i++) {

System.out.println(i);

}

}

}

// Error to investigate: Why does this loop not print numbers in the expected order? What is the problem with the initialization and update statements in the `for` loop?

**Instead of i++ write i—**

**public class Main {**

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

**for (int i = 10; i >= 0; i--) {**

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

**}**

**}**

**}**

Snippet 6:

public class MisplacedForLoopBody { public static void main(String[] args) {

for (int i = 0; i < 5; i++) System.out.println(i);

System.out.println("Done");

}

}

// Error to investigate: Why does "Done" print only once, outside the loop? How should the loop body be enclosed to include all statements within the loop?

**Because there is not curly braces so it will take only one statement.**

**public class Main {**

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

**for (int i = 0; i < 5; i++)**

**{**

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

**System.out.println("Done");**

**}**

**}**

**}**

Snippet 7:

public class UninitializedWhileLoop { public static void main(String[] args) { int count;

while (count < 10) {

System.out.println(count);

count++;

}

}

}

// Error to investigate: Why does this code produce a compilation error? What needs to be done to initialize the loop variable properly?

**Because count variable is not initialize .**

**public class Main {**

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

**int count=0;**

**while (count < 10) {**

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

**count++;**

**}**

**}**

**}**

Snippet 8:

public class OffByOneDoWhileLoop {

public static void main(String[] args) {

int num = 1;

do {

System.out.println(num);

num--;

} while (num > 0);

}

}

// Error to investigate: Why does this loop print unexpected numbers? What adjustments are needed to print the numbers from 1 to 5?

**Because in while condition is not satisfying.**

**Write num++ and change the condition in while**

**public class Main{**

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

**int num = 1;**

**do {**

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

**num++;**

**} while (num <= 5);**

**}**

**}**

Snippet 9:

public class InfiniteForLoopUpdate { public static void main(String[] args) { for (int i = 0; i < 5; i += 2) {

System.out.println(i);

}

}

}

// Error to investigate: Why does the loop print unexpected results or run infinitely? How should the loop update expression be corrected?

**public class Main {**

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

**for (int i = 0; i < 5; i++) {**

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

**}**

**}**

**}**

Snippet 10:

public class IncorrectWhileLoopControl { public static void main(String[] args) {

int num = 10;

while (num = 10) { System.out.println(num);

num--;

}

} }

// Error to investigate: Why does the loop execute indefinitely? What is wrong with the loop condition?

**public class Main {**

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

**int num = 10;**

**while (num>0) {**

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

**num--;**

**}**

**}**

**}**

Snippet 11:

public class IncorrectLoopUpdate {

public static void main(String[] args) { int i = 0;

while (i < 5) {

System.out.println(i);

i += 2; // Error: This may cause unexpected results in output

}

}

}

// Error to investigate: What will be the output of this loop? How should the loop variable be updated to achieve the desired result?

**Output gives even nos. till 5.**

**public class Main {**

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

**int i = 0;**

**while (i < 5) {**

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

**i++; // Error: This may cause unexpected results in output**

**}**

**}**

**}**

Snippet 12:

public class LoopVariableScope { public static void main(String[] args) {

for (int i = 0; i < 5; i++) {

int x = i \* 2;

}

System.out.println(x); // Error: 'x' is not accessible here

}

}

// Error to investigate: Why does the variable 'x' cause a compilation error? How does scope

**Because x is defined inside the for loop which means it has local scope.**

SECTION 2: Guess the Output

Instructions:

1. Perform a Dry Run: Carefully trace the execution of each code snippet manually to determine the output.
2. Write Down Your Observations: Document each step of your dry run, including the values of variables at each stage of execution.
3. Guess the Output: Based on your dry run, provide the expected output of the code.
4. Submit Your Assignment: Provide your dry run steps along with the guessed output for each code snippet.

Snippet 1:

public class NestedLoopOutput { public static void main(String[] args) {

for (int i = 1; i <= 3; i++) { for (int j = 1; j <= 2; j++) {

System.out.print(i + " " + j + " ");

}

System.out.println();

}

}

}

// Guess the output of this nested loop.

**i j**

**1 1**

**2 print--> 1 1 1 2**

**2 1**

**2 print--> 2 1 2 2**

**3 1**

**2 print--> 3 1 3 2**

Snippet 2:

public class DecrementingLoop { public static void main(String[] args) { int total = 0; for (int i = 5; i > 0; i--) { total += i; if (i == 3) continue;

total -= 1;

}

System.out.println(total);

}

}

// Guess the output of this loop.

**total i**

**0**

**(0+5)=5 5**

**(5-1)=4 5**

**(4+4)=8 4**

**(8-1)=7 4**

**(7+3)=10 3**

**(10+2)=12 2**

**(12-1)=11 2**

**(11+1)=12 1**

**(12-1)=11 1**

**0**

**Total=11**

Snippet 3:

public class WhileLoopBreak { public static void main(String[] args) {

int count = 0; while (count < 5) {

System.out.print(count + " ");

count++; if (count == 3) break;

}

System.out.println(count);

}

}

// Guess the output of this while loop.

**0 1 2 3**

Snippet 4:

public class DoWhileLoop { public static void main(String[] args) { int i = 1; do {

System.out.print(i + " ");

i++;

} while (i < 5);

System.out.println(i);

}

}

// Guess the output of this do-while loop.

**1 2 3 4 5**

Snippet 5:

public class ConditionalLoopOutput { public static void main(String[] args) {

int num = 1; for (int i = 1; i <= 4; i++) { if (i % 2 == 0) { num += i; } else { num -= i;

}

}

System.out.println(num);

}

}

// Guess the output of this loop.

**num i**

**1**

1. **1**
2. **2**

**-1 3**

**3 4**

**Print 🡪 3**

Snippet 6:

public class IncrementDecrement {

public static void main(String[] args) { int x = 5; int y = ++x - x-- + --x + x++;

System.out.println(y);

}

}

// Guess the output of this code snippet.

**6 – 6 + 4 + 4 =y**

Snippet 7:

public class NestedIncrement { public static void main(String[] args) { int a = 10; int b = 5; int result = ++a \* b-- - --a + b++;

System.out.println(result);

}

}

// Guess the output of this code snippet.

**11 \* 5 – 10 + 4**

**Ans = 49**

Snippet 8:

public class LoopIncrement { public static void main(String[] args) {

int count = 0;

for (int i = 0; i < 4; i++) {

count += i++ - ++i;

}

System.out.println(count);

}

}

// Guess the output of this code snippet.

**cnt i**

**0 0**

**-2 2**

**-4 4 count=-4**

SECTION 3: Lamborghini Exercise:

Instructions:

1. Complete Each Program: Write a Java program for each of the tasks listed below.
2. Test Your Code: Make sure your code runs correctly and produces the expected output.
3. Submit Your Solutions: Provide the complete code for each task along with sample output.

Tasks:

* 1. Write a program to calculate the sum of the first 50 natural numbers.

**public class Sum50Nos{**

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

**{**

**int num=1;**

**while(num<=50)**

**{**

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

**num++;**

**}**

**}**

**}**

* 1. Write a program to compute the factorial of the number 10.

**public class Fact10{**

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

**{**

**int fact=1;**

**for(int i=1;i<=10;i++)**

**{**

**fact=fact\*i;**

**}**

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

**}**

**}**

* 1. Write a program to print all multiples of 7 between 1 and 100.

**public class Multiple7{**

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

**{**

**for(int i=1;i<=100;i++)**

**{**

**if(i%7==0)**

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

**}**

**}**

**}**

* 1. Write a program to reverse the digits of the number 1234. The output should be 4321.

**public class ReverseNo{**

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

**{**

**int num=1234,rev=0;**

**while(num!=0)**

**{**

**rev=rev\*10+num%10;**

**num=num/10;**

**}**

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

**}**

**}**

* 1. Write a program to print the Fibonacci sequence up to the number 21.

**public class Fibo21{**

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

**{**

**int f1=0,f2=1,res=0;**

**System.out.print(f1+" ");**

**for(int i=1;i<=21;i++)**

**{**

**f1=f2;**

**f2=res;**

**res=f1+f2;**

**System.out.print(res+" ");**

**}**

**}**

**}**

* 1. Write a program to find and print the first 5 prime numbers.

**public class primeNo5**

**{**

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

**{**

**for(int i=2;i<=11;i++)**

**{**

**for(int j=2;j<=i/2;j++)**

**{**

**if(i%j==0)**

**{**

**cnt++;**

**break;**

**}**

**}**

**if(cnt==0)**

**{**

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

**}**

**}**

**}**

**}**

* 1. Write a program to calculate the sum of the digits of the number 9876. The output should be 30 (9 + 8 + 7 + 6).

**public class SumofDigits{**

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

**{**

**int num=9876,sum=0;**

**while(num!=0)**

**{**

**sum+=num%10;**

**num=num/10;**

**}**

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

**}**

**}**

* 1. Write a program to count down from 10 to 0, printing each number.

**public class Print10to0{**

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

**{**

**for(int i=10;i>=0;i--)**

**{**

**System.out.print(i+" ");**

**}**

**}**

**}**

* 1. Write a program to find and print the largest digit in the number 4825.

**public class LargeDigit{**

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

**{**

**int num=4825,rem1=0,rem2=0,rem3=0,rem4=0;**

**while(num!=0)**

**{**

**rem1=num%10;**

**num/=10;**

**rem2=num%10;**

**num/=10;**

**rem3=num%10;**

**rem4=num/10;**

**}**

**if(rem1>rem2 && rem1>rem3)**

**{**

**if(rem1>rem4)**

**System.out.print(rem1);**

**}**

**else if(rem2>rem3 && rem2>rem4)**

**{**

**System.out.print(rem2);**

**}**

**else if (rem3>rem4)**

**{**

**System.out.println(rem3);**

**}**

**else**

**{**

**System.out.println(rem4);**

**}**

**}**

**}**

* 1. Write a program to print all even numbers between 1 and 50.

**public class EvenTill50{**

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

**{**

**for(int i=1;i<=50;i++)**

**{**

**if(i%2==0)**

**System.out.print(i+" ");**

**}**

**}**

**}**

* 1. Write a Java program to demonstrate the use of both pre-increment and post-decrement operators in a single expression

**public class UnaryOperator{**

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

**{**

**int a=14;**

**int res=a++ \* a-- + ++a \* --a;**

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

**}**

**}**

* 1. Write a program to draw the following pattern:

\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

**public class pattern12{**

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

**{**

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

**{**

**for(int j=1;j<=5;j++)**

**{**

**System.out.print("\*");**

**}**

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

**}**

**}**

**}**

* 1. Write a program to print the following pattern:

1

2\*2

3\*3\*3

4\*4\*4\*4

5\*5\*5\*5\*5

5\*5\*5\*5\*5

4\*4\*4\*4

3\*3\*3

2\*2

1

**public class pattern13{**

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

**{**

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

**{**

**for(int j=1;j<=2\*i-1;j++)**

**{**

**if(j%2==0)**

**System.out.print("\*");**

**else**

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

**}**

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

**}**

**for(int i=5;i>=1;i--)**

**{**

**for(int j=1;j<=2\*i-1;j++)**

**{**

**if(j%2==0)**

**System.out.print("\*");**

**else**

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

**}**

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

**}**

**}**

**}**

* 1. Write a program to print the following pattern:

\*

\*\*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*

**public class pattern14{**

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

**{**

**for(int i=1;i<=9;i++)**

**{**

**if(i%2==0 &&i>2)**

**{**

**continue;**

**}**

**for(int j=1;j<=i;j++)**

**{**

**System.out.print("\*");**

**}**

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

**}**

**}**

**}**

* 1. Write a program to print the following pattern:

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

**public class pattern15**

**{**

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

**{ int n=5;**

**for(int i=1;i<=n;i++)**

**{**

**for(int j=1;j<=n-i;j++)**

**{**

**System.out.print(" ");**

**}**

**for(int j=1;j<=i;j++)**

**{**

**System.out.print(" \*");**

**}**

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

**}**

**}**

**}**

* 1. Write a program to print the following pattern:

\*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*

**public class pattern16{**

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

**{ int n=5;**

**for(int i=1;i<=n;i++)**

**{**

**for(int j=1;j<=n-i;j++)**

**{**

**System.out.print(" ");**

**}**

**for(int j=1;j<=2\*i-1;j++)**

**{**

**System.out.print("\*");s**

**}**

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

**}**

**}**

**}**

* 1. Write a program to print the following pattern:

\*\*\*\*\*

\*\*\*\*

\*\*\*

\*\*

\*

* 1. Write a program to print the following pattern:

**public class pattern18{**

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

**{**

**int n=4;**

**for(int i=1;i<=n;i++)**

**{**

**for(int j=1;j<=n-i;j++)**

**{**

**System.out.print(" ");**

**}**

**for(int j=1;j<=2\*i-1;j++)**

**{**

**System.out.print("\*");**

**}**

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

**}**

**for(int i=n-1;i>=1;i--)**

**{**

**for(int j=1;j<=n-i;j++)**

**{**

**System.out.print(" ");**

**}**

**for(int j=1;j<=2\*i-1;j++)**

**{**

**System.out.print("\*");**

**}**

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

**}**

**}**

**}**

* 1. Write a program to print the following pattern:

1

1\*2

1\*2\*3

1\*2\*3\*4

1\*2\*3\*4\*5

**public class pattern19**

**{**

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

**{**

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

**{**

**for(int j=1;j<=i;j++)**

**{**

**System.out.print(j);**

**if(j<i)**

**System.out.print("\*");**

**}**

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

**}**

**}**

**}**

* 1. Write a program to print the following pattern:

5

5\*4

5\*4\*3

5\*4\*3\*2

5\*4\*3\*2\*1

**public class pattern20**

**{**

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

**{**

**for(int i=5;i>=1;i--)**

**{**

**for(int j=5;j>=i;j--)**

**{**

**System.out.print(j);**

**if(j>i)**

**System.out.print("\*");**

**}**

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

**}**

**}**

**}**

* 1. Write a program to print the following pattern:

1

1\*3

1\*3\*5

1\*3\*5\*7

1\*3\*5\*7\*9

**public class pattern21**

**{**

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

**{**

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

**{**

**for(int j=1;j<=2\*i-1;j++)**

**{**

**if(j%2!=0)**

**{**

**System.out.print(j);**

**}**

**else**

**{**

**System.out.print("\*");**

**}**

**}**

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

**}**

**}**

**}**

* 1. Write a program to print the following pattern:

\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*

\*\*\*

\*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*

**public class pattern22**

**{**

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

**{**

**int n=5;**

**for(int i=n;i>=1;i--)**

**{**

**for(int j=1;j<=n-i;j++)**

**{**

**System.out.print(" ");**

**}**

**for(int j=2\*i-1;j>=1;j--)**

**{**

**System.out.print("\*");**

**}**

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

**}**

**for(int i=2;i<=n;i++)**

**{**

**for(int j=1;j<=n-i;j++)**

**{**

**System.out.print(" ");**

**}**

**for(int j=1;j<=2\*i-1;j++)**

**{**

**System.out.print("\*");**

**}**

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

**}**

**}**

**}**

* 1. Write a program to print the following pattern:

11111

22222

33333

44444

55555

**public class pattern23**

**{**

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

**{**

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

**{**

**for(int j=1;j<=5;j++)**

**{**

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

**}**

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

**}**

**}**

**}**

* 1. Write a program to print the following pattern:

1

22

333

4444

55555

**public class pattern24**

**{**

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

**{**

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

**{**

**for(int j=1;j<=i;j++)**

**{**

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

**}**

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

**}**

**}**

**}**

* 1. Write a program to print the following pattern:

1

12

123

1234

12345

**public class pattern25**

**{**

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

**{**

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

**{**

**for(int j=1;j<=i;j++)**

**{**

**System.out.print(j);**

**}**

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

**}**

**}**

**}**

* 1. Write a program to print the following pattern:

1

2 3

4 5 6

7 8 9 10

11 12 13 14 15

**public class pattern26**

**{**

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

**{ int num=1;**

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

**{**

**for(int j=1;j<=i;j++)**

**{**

**System.out.print(num+" ");**

**num++;**

**}**

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

**}**

**}**

**}**