Section A:

-----------------------------------------------------------------------------------------------------------------------------------------------------------

**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?**

Error: As i is starting from zero and decrementing ,the condition i<=10 never becomes true and loop will execute countinously

correct snippet:

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?**

Error:int cannot be converted to boolean

public class IncorrectWhileCondition {

public static void main(String[] args) {

int count=0;

while (count>=5)

{

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 `dowhile`loop**

The loop starts from 0 until condition num>0 until the condition satisfies

so in this case the condition satisfies as num start as 0 print 0 and then increment to 1 which satisfies the condition num>0

public class DoWhileIncorrectCondition {

public static void main(String[] args) {

int num = 0;

do {

System.out.println(num);

num++;

} while (num > 5);

}

}

------------------------------------------------------------------------------------------------------------------------------------

**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?**

in for loop the condition is i<=10 which means loop until i values less or equal to 10 so it will print until 10

if we remove the equal to '=' sign the loop will excecute until i value less than 10

**Correct snippet:**

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

}

}

-----------------------------------------------------------------------------------------------------------------------------------------------------------

**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?**

Error: her i initialvalue is 10 and condition is i greater then equal to zero

due to increment of i thecondition never satisfies and the loop will excecute infinite time, we need to decrement the value of i to satiesfy the condition.

correct code:

public class WrongInitializationForLoop {

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?**

Error:

For loop scope not define, we need to close the for loop body in curly braces.

correct snippet;

public class MisplacedForLoopBody {

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?

error: variable count might not have been initialized

while (count < 10) {

^

1 error

correct snippet:

public class UninitializedWhileLoop {

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?

Error: here the condition is satiesfied in loop only as num >0 we need to change the condition and increment the loop.

correct snippet:

public class OffByOneDoWhileLoop {

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 InfiniteForLoopUpdate {

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?

error: incompatible types: int cannot be converted to boolean

while (num = 10) {

^

= is assignment operator we need to use a conditional operator here

and the loop start with value 10 and decrement which do not satiesfy the condition

correct snippet:

public class IncorrectWhileLoopControl {

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 is:

0

2

4

in this case the i value increment by 2 in every loop due to i +=2;

correct snippet:

public class IncorrectLoopUpdate {

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

Error: output of x is declared out of the scope of loop

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

}

}

}

---------------------------------------------------------------------------------------------------------

SECTION 2: Guess the Output

------------------------------------------------------------------------------------------------------

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();

Dry run:

i J Output

1 1 1 1

2 1 2

2 1 2 1

2 2 2

3 1 3 1

2 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.

Dry run:

i Total Output

5 5

4 9

3 12 11

2 14

1 15

---------------------------------------------------------------------------

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.

Dry run:

count print i++ if( Count==3) print

0 0 1 No

1 1 2 No

2 2 3 yes 3

Output is 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.

Dry run:

i print i++ print

1 1 2

2 2 3

3 3 4

4 4 5 5

Output: 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.

Dry run:

i if Num print

1 False 0

2 True 3 3

3 False 2

4 False 4

Output: 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.

Dry run:

y = ++x - x-- + --x + x++;

6 - 6 + 4 + 4

4 + 4

8

Output: 8

---------------------------------------------------------------------

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.

++a \* b-- - --a + b++;

11 \* 5 - 10 + 4

45 + 4

49

Output: 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.

Dry run:

count i++ - ++i

0 0 - 2

-2 1 - 3

-4 2 - 4

-2 3 - 5

-4

Output:-4

--------------------------------------------------------------------

Section: 3

--------------------------------------------------------------------

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

Code:

public class NaturalNum

{

public static void main(String[] args)

{

int i, num = 50, sum=0;

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

{

sum=sum+i;

}

System.out.println("Sum of first "+num+" Natural number is :"+sum);

}

}

Code:

A computer screen shot of a code

Description automatically generated

Output:

A screen shot of a computer

Description automatically generated

-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

2. Write a program to compute the factorial of the number 10.

Code:

public class FactNum

{

public static void main(String[] args)

{

int i, num =10, fact=1;

for(i=1;i<=num;i++)

{

fact=fact\*i;

}

System.out.println("Factorial of "+num+" is: "+fact);

}

}

Output:

A black background with white text

Description automatically generated

-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

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

Code:

public class Factseven

{

public static void main(String[] args)

{

int i=1 ;

System.out.println("Multiple of 7 from 1 to 100 are:");

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

{

if(i%7==0)

{

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

}

}

}

}

output:A black screen with white text

Description automatically generated

-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

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

Code:

public class Rev

{

public static void main(String[] args)

{

int i=1234,rev=0 ;

while( i!= 0)

{

int rem = i % 10;

rev = rev \* 10 + rem;

i = i/10;

}

System.out.println("The reverse of the given number is: " + rev);

}

}

Output:

A black background with white text

Description automatically generated

-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

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

Code:

public class Fibo

{

public static void main(String[] args) {

int n =15, num1 = 0, num2 = 1;

System.out.println("Fibonacci Series till " + n + " terms:");

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

while(num1<=21)

{

System.out.print(num1 + ", ");

int nex = num1 + num2;

num1 = num2;

num2 = nex;

}

}

}

}

Output:

A black screen with white text

Description automatically generated

-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

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

Code:

int i=start;

while(i<=end)

{

int chk = 0;

int j=2;

while(j<i)

{

if(i%j==0)

{

chk++;

break;

}

j++;

}

if(chk==0)

System.out.println(i);

i++;

}

Output:

A computer screen with white text

Description automatically generated

7. Write a program to calculate the sum of the digits of the number 9876. The output should be

30 (9 + 8 + 7 + 6).

Code:

public class SumOfNum

{

public static void main (String[]args)

{

int num = 9876, sum = 0;

while(num!=0)

{

sum += num % 10;

num = num / 10;

}

System.out.println ("Sum of digits : " + sum);

}

}

Output:

A black background with white text

Description automatically generated

-----------------------------------------------------------------------------------------------------------------------------------------------------------

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

Code:

public class Countnum

{

public static void main(String[] args)

{

for(int i=10;i>=1;i--)

{

System.out.println(i);

}

}

}

Output:

A computer screen with white text

Description automatically generated

-----------------------------------------------------------------------------------------------------------------------------------------------------------

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

public class Largest {

public static void main(String[] args)

{

int n=4825,r=0,max=0;

while(n>0)

{

r=n%10;

if(r>max)

{

max=r;

}

n=n/10;

}

System.out.println("Largest digit: "+max);

}

}

Output:

A black background with white text

Description automatically generated

-----------------------------------------------------------------------------------------------------------------------------------------------------------

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

Code:

public class Even

{

public static void main(String[] args)

{

int n=50;

System.out.print("Even numbers from 1 to 50: ");

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

{

if (i%2==0)

{

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

}

}

}

}

Output:

A black background with white text

Description automatically generated

-----------------------------------------------------------------------------------------------------------------------------------------------------------

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

public class Prepost

{

public static void main(String[] args)

{

int n=5;

int m=2;

int a=++n - m--;

System.out.print("Result of preincrement and postdecrement addition is :"+a);

}

}

Output :

A black screen with white text

Description automatically generated

-----------------------------------------------------------------------------------------------------------------------------------------------------------

12. Write a program to draw the following pattern:

\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

Code:

public class Starpattern1

{

public static void main(String args[])

{

int i, j;

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

{

for(j=0; j<=5; j++)

{

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

}

System.out.println();

}

}

}

Output:

A computer screen with white text

Description automatically generated

-----------------------------------------------------------------------------------------------------------------------------------------------------------

13. 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

Code:

public class StarnumPattern

{

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(i);

}

for(int i=4;i>=1;i--)

{

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

{

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

}

System.out.println(i);

}

}

}

Output:

A computer screen with white text

Description automatically generated

-----------------------------------------------------------------------------------------------------------------------------------------------------------

14. Write a program to print the following pattern:

\*

\*\*

\*\*\*

\*\*\*\*\*

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

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

Code:

import java.util.Scanner;

public class Starpattern3{

public static void main(String args[]){

int n;

System.out.println("Enter a number");

Scanner scanner = new Scanner(System.in);

n = scanner.nextInt();

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

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

continue;

}

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

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

}

System.out.println();

}

}

}

Output:

A computer screen with white text

Description automatically generated

-----------------------------------------------------------------------------------------------------------------------------------------------------------

15. Write a program to print the following pattern:

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

Code: public class Pattern{

public static void main(String []args)

{

int i,j,rows=5;

for(i=0;i<=rows;i++)

{

for (j=rows-i; j>1; j--)

{

System.out.print(" ");

}

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

{

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

}

System.out.println();

}

}

}

Output:

A computer screen with white text

Description automatically generated

-----------------------------------------------------------------------------------------------------------------------------------------------------------

16. Write a program to print the following pattern:

\*

\*\*\*

\*\*\*\*\*

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

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

public class Pattern4{

public static void main(String []args)

{

int i,j,rows=9;

for(i=1;i<=rows;i++)

{

for (j=rows-i; j>1; j--)

{

System.out.print(" ");

}

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

{

if(i%2==0)

{

continue;

}

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

}

System.out.println();

}

}

}

Output:

A computer screen with white text

Description automatically generated

-----------------------------------------------------------------------------------------------------------------------------------------------------------

17. Write a program to print the following pattern:

\*\*\*\*\*

\*\*\*\*

\*\*\*

\*\*

\*

Code:

public class Pattern5

{

public static void main(String args[])

{

int rows=5;

for (int i= 0; i<= rows-1; i++)

{

for (int j=0; j<=i; j++)

{

System.out.print(" ");

}

for (int k=0; k<=rows-1-i; k++)

{

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

}

System.out.println();

}

}

}

Output:

A black screen with white text

Description automatically generated

18. Write a program to print the following pattern:

\*

\*\*\*

\*\*\*\*\*

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

\*\*\*\*\*

\*\*\*

\*

Code:

public class Pattern6

{

public static void main(String args[])

{

int row=4, i, j, s = 1;

s = row - 1;

for (j = 1; j<=row; j++)

{

for (i = 1; i<=s; i++)

{

System.out.print(" ");

}

s--;

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

{

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

}

System.out.println("");

}

s = 1;

for (j = 1; j<= row - 1; j++)

{

for (i = 1; i<= s; i++)

{

System.out.print(" ");

}

s++;

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

{

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

}

System.out.println("");

}

}

}

Output:

A computer screen with white text

Description automatically generated

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

**1**

**1\*2**

**1\*2\*3**

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

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

Code:

public class Numpattern1

{

public static void main(String[] args)

{

int rows=5, i,j;

for(i=1;i<rows;i++)

{

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

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

{

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

}

}

System.out.print(i);

}

}

Output:

A computer screen with white text

Description automatically generated

-----------------------------------------------------------------------------------------------------------------------------------------------------------

20. Write a program to print the following pattern:

5

5\*4

5\*4\*3

5\*4\*3\*2

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

Code:

public class Numpattern2

{

public static void main(String[] args)

{

int rows=5, i,j;

for(i=rows;i>=1;i--)

{

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

for(j=rows;j>=i;j--)

{

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

}

}

System.out.println();

}

}Output:

A computer screen with white text

Description automatically generated

21. Write a program to print the following pattern:

1

1\*3

1\*3\*5

1\*3\*5\*7

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

Code:

public class Numpattern3

{

public static void main(String[] args)

{

int rows=9, i,j;

for(i=1;i<rows;i+=2)

{

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

for(j=1;j<=i;j+=2)

{

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

}

}

System.out.print(i);

}

}

Output:

A screen shot of a computer

Description automatically generated

22. Write a program to print the following pattern:

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

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

\*\*\*\*\*

\*\*\*

\*

\*\*\*

\*\*\*\*\*

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

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

Code:

public class Pattern7

{

public static void main(String []args)

{

int i,j,rows=9;

{

for(i=0;i<=rows-1;i++)

{

for (j=0; j<i; j++)

{

System.out.print(" ");

}

for( j=i;j<=rows-1;j++)

{

if(i%2==0)

{

continue;

}

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

}

System.out.println();

}

}

for(i=1;i<=rows;i++)

{

for (j=rows-i; j>1; j--)

{

System.out.print(" ");

}

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

{

if(i%2==0)

{

continue;

}

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

}

System.out.println();

}

}

}

Output:

A computer screen shot of a black screen

Description automatically generated

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

23. Write a program to print the following pattern:

11111

22222

33333

44444

55555

Code:

public class Numpattern4

{

public static void main(String args[])

{

int i, j;

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

{

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

{

System.out.print(i);

}

System.out.println("");

}

}

}

Output:

A screen shot of a computer

Description automatically generated

24. Write a program to print the following pattern:

1

22

333

4444

55555

Code:

public class Numpattern5

{

public static void main(String args[])

{

int i, j;

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

{

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

{

System.out.print(i);

}

System.out.println("");

}

}

}

Output:

A computer screen with white text

Description automatically generated

-----------------------------------------------------------------------------------------------------------------------------------------------------------

25. Write a program to print the following pattern:

1

12

123

1234

12345

Code:

public class Numpattern6

{

public static void main(String args[])

{

int i, j;

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

{

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

{

System.out.print(j);

}

System.out.println("");

}

}

}

Output: A computer screen with white text

Description automatically generated

26. Write a program to print the following pattern:

1

2 3

4 5 6

7 8 9 10

11 12 13 14 15

Code:

public class Numpattern7

{

public static void main(String args[])

{

int i, j,k=1;

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

{

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

{

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

}

System.out.println("");

}

}

}

Output:

A computer screen with white text

Description automatically generated