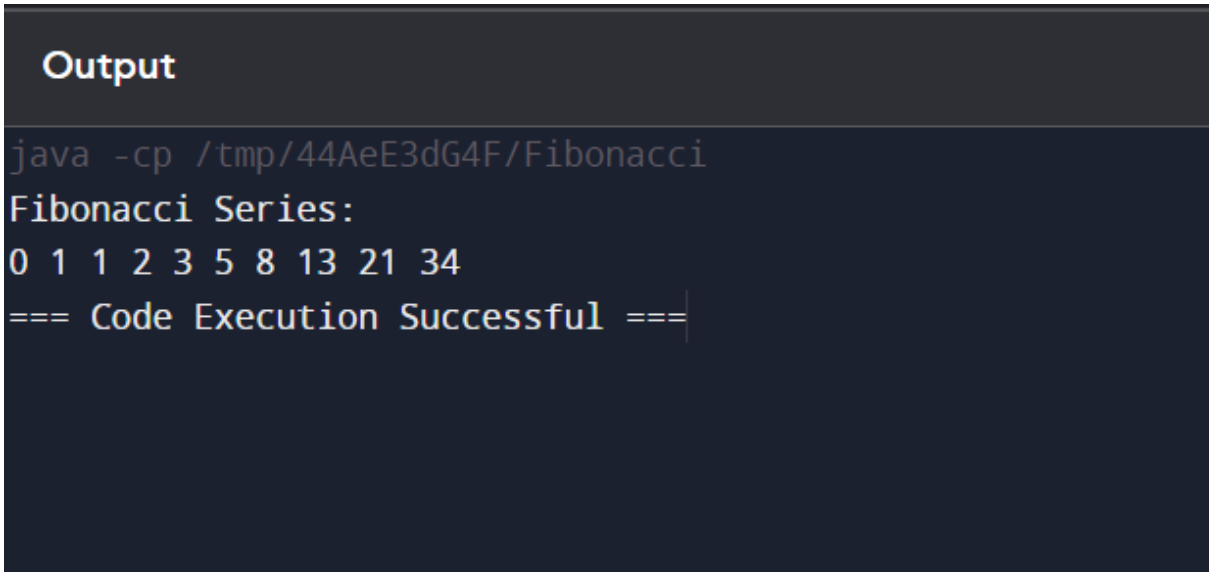
1.Recursive fibonacci series

CODE:

```
public class Fibonacci {
static int fibonacci(int n) {
if (n <= 1)
return n;
return fibonacci(n - 1) + fibonacci(n - 2);
}
public static void main(String[] args) {
int n = 10;
System.out.println("Fibonacci Series:");
for (int i = 0; i < n; i++) {
System.out.print(fibonacci(i) + " ");
}
}
}
```

OUTPUT:



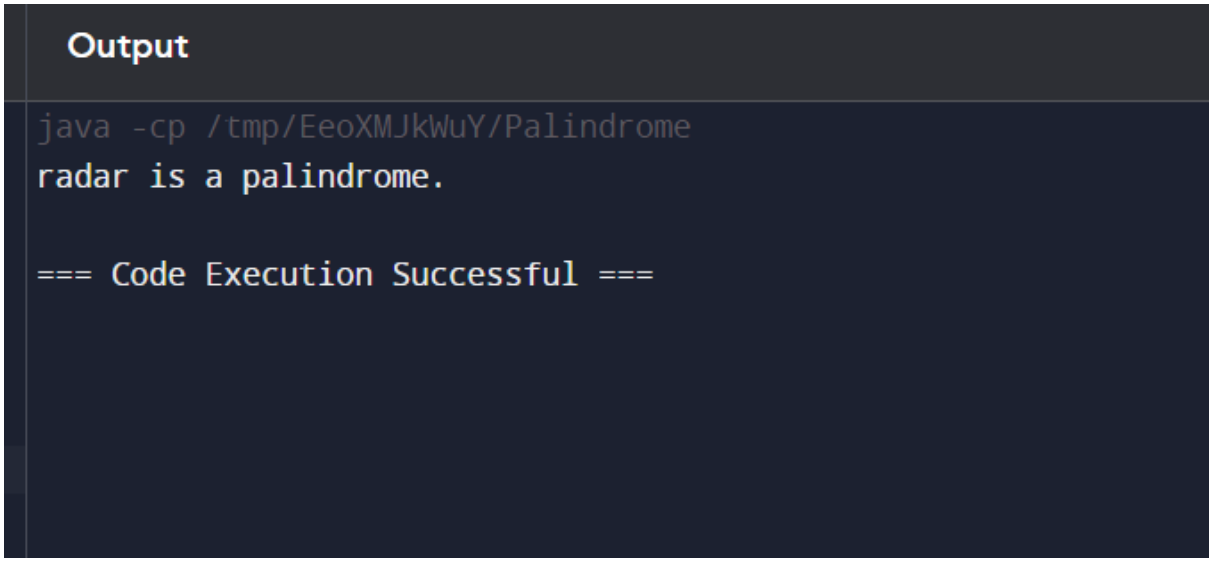
2. check whether the string is palindrome or not

CODE:

```
public class Palindrome {
static boolean isPalindrome(String str, int start, int end) {
```

```
if (start >= end)
return true;
if (str.charAt(start) != str.charAt(end))
return false;
return isPalindrome(str, start + 1, end - 1);
}
public static void main(String[] args) {
String str = "radar";
if (isPalindrome(str, 0, str.length() - 1))
System.out.println(str + " is a palindrome.");
else
System.out.println(str + " is not a palindrome.");
}
}
```

OUTPUT:



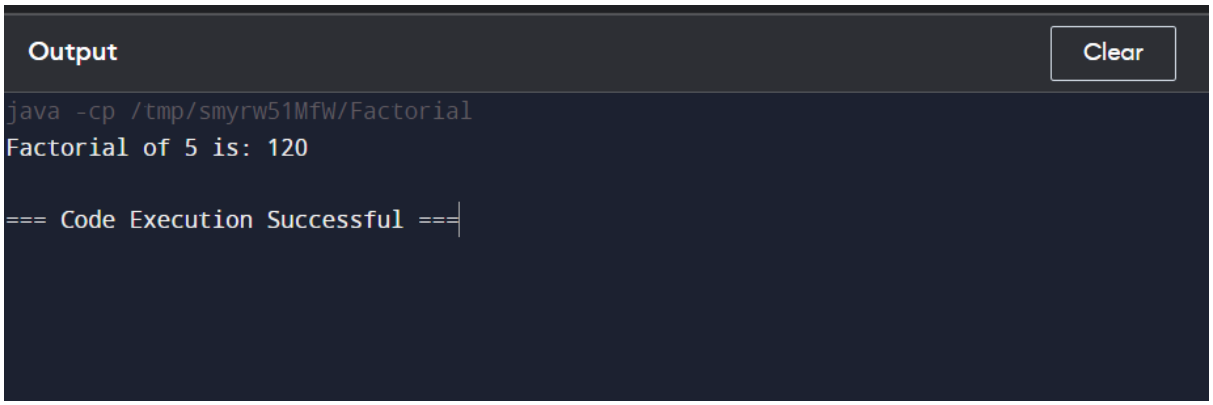
3. find the factorial

CODE:

```
public class Factorial {
static int factorial(int n) {
if (n == 0)
```

```
return 1;
return n * factorial(n - 1);
}
public static void main(String[] args) {
int n = 5;
System.out.println("Factorial of " + n + " is: " + factorial(n));
}
}
```

OUTPUT:



4. to find the series of number with recusive methods

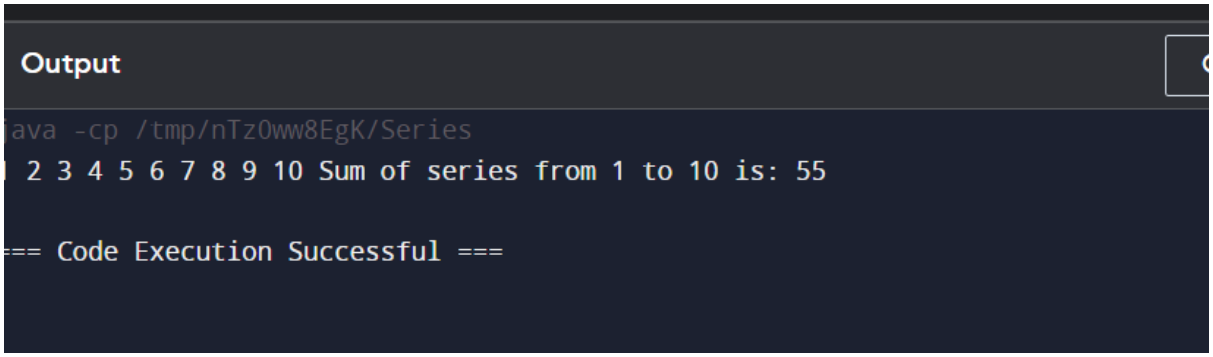
CODE:

```
public class Series {
static int sumSeries(int n) {
if (n == 1) {
System.out.print(n + " ");
return 1;
}
int sum = n + sumSeries(n - 1);
System.out.print(n + " ");
return sum;
}
public static void main(String[] args) {
int n = 10;
System.out.println("Sum of series from 1 to " + n + " is: " + sumSeries(n));
```

}

}

OUTPUT:


Output
java -cp /tmp/nTzOww8EgK/Series
2 3 4 5 6 7 8 9 10 Sum of series from 1 to 10 is: 55
=== Code Execution Successful ===