10.Implement a program to print the first 15 prime numbers using a **while** loop. Break the loop if a number is not prime.

public class PrimeNumbers {

public static void main(String[] args) {

int count = 0;

int num = 2;

while (count < 15) {

boolean isPrime = true;

// Optimize by checking divisibility only up to the square root of num

for (int i = 2; i <= Math.sqrt(num); i++) {

if (num % i == 0) {

isPrime = false;

break; // Exit the inner loop if a divisor is found

}

}

if (isPrime) {

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

count++;

}

num++; // Move to the next number

}

}

}

Here's a line-by-line explanation of the Java code:

1. public class PrimeNumbers {

* This line declares a public class named PrimeNumbers, which acts as a container for the code.

2. public static void main(String[] args) {

* This line defines the main method, the entry point of the program.

3. int count = 0;

* Declares and initializes an integer variable count to 0, which will keep track of the number of prime numbers found.

4. int num = 2;

* Declares and initializes an integer variable num to 2, the first number to be checked for primality.

5. while (count < 15) {

* Starts a while loop that continues as long as count is less than 15 (meaning we haven't found 15 prime numbers yet).

6. boolean isPrime = true;

* Declares and initializes a boolean variable isPrime to true, assuming the current num is prime.

7. for (int i = 2; i <= Math.sqrt(num); i++) {

* Starts a for loop to check for divisibility. It iterates from 2 up to the square root of num (optimization).

8. if (num % i == 0) {

* Checks if num is divisible by the current i. If it is, it means num is not prime.

9. isPrime = false;

* Sets isPrime to false to indicate that num is not prime.

10. break; - Exits the inner for loop, as there's no need to check further divisors.

11. } - Closes the for loop.

12. if (isPrime) { - Checks if isPrime is still true (meaning no divisors were found).

13. System.out.print(num + " "); - Prints the prime number num followed by a space.

14. count++; - Increments the count of prime numbers found.

15. } - Closes the if statement.

16. num++; - Increments num to check the next number for primality.

17. } - Closes the while loop.

18. } - Closes the main method.

19. } - Closes the PrimeNumbers class.