1. public class LargestPrime {
2. public static int getLargestPrime(int number) {
3. if (number < 2) {
4. return -1;
5. }
6. for (int i = 2; i < number; i++) {
7. if ((number % i) == 0) {
8. number /= i;
9. i--;
10. }
11. }
12. return number;
13. }
14. }

Explanation-

1. in for loop i = 2; i < number; means now 2 < number;
2. for example lets keep number = 45
3. so, 2 < 45 which is true so it will enter the for loop
5. first iteration 45 % 2 ==0 which is false
6. so the condition will become true in following cases
7. 3 , 5 , 9 ,15 so
9. 1) if(45 % 3 == 0) true and
10. number = 45 / 3
11. number = 15
12. where i value is still 3
13. then i-- will make i = 2 as it decrements
14. then the i++ in the for loop again makes it 3
15. and the number = number / i
16. makes number value to be updated to 15
17. now condition is checked with i = 3 and number =15
19. 2)if(3 < 15) true and
20. if(15 % 3 == 0) true
21. number = 15 / 3
22. number = 5
23. where i value is still 3
24. then i-- will make i = 2 as it decrements
25. then the i++ in the for loop again makes it 3
26. and the number = number / i
27. makes number value to be updated to 5
28. now condition is checked with i = 3 and number =5
30. 3)if(3 < 5) true and
31. if(5 % 3 ==0) false
32. the if condition dosent run
33. so when we return the number value which is 5 will be returned



38. so generally i-- is used to keep the i value as minimum factor which can divide the number and we are reducing the number.