Operators and More:

1.A vampire number v is a number with an even number of digits n, that can be factored into two numbers x and y each with n/2 digits and not both with trailing zeroes, where v contains precisely all the digits from x and from y, in any order. Write a java program to print first 100 vampire numbers.

Description:

Finding a vampire number involves identifying a number that has an even number of factors, where the number can be expressed as a product of two factors each having half the number of digits of the original number. Additionally, neither of these factors should have trailing zeros, and all digits in these factors should be present in the vampire number.

Approach:

Check for Even Number of Digits: Verify if the vampire number (v) has an even number of digits. If it does not, return false.

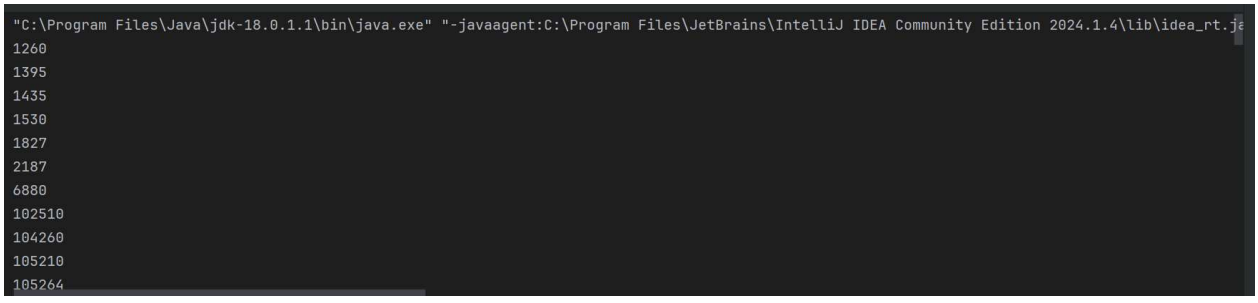
Find Factors: Iterate through potential factors from the smallest number with n/2 digits to the largest number with n/2 digits. For example, if v has 2 digits, iterate from 1 to 9.

Validate Factors: - Ensure both factors have exactly n/2 digits.
- Check that neither factor has trailing zeros.

Digit Validation: Verify that all digits in the two factors (x and y) are present in the vampire number (v). Concatenate and sort the digits of x and y, and compare this sorted concatenation with the sorted digits of v.

Return Result: If all conditions are satisfied, return true and print the vampire number. Continue this process to find the first 100 vampire numbers.

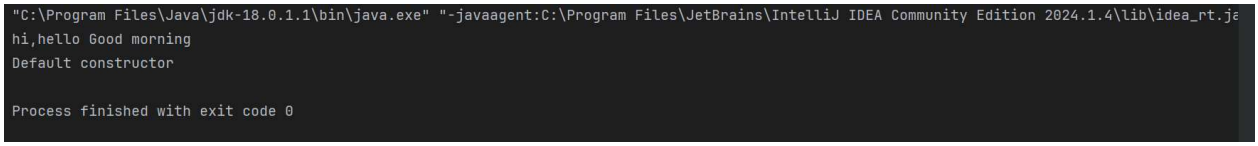
output



2.Create a class with two (overloaded) constructors. Using this, call the second constructor inside the first one

approach-
-Can call one constructor from other with the help of this (which refers to current object)
-Called parametrized constructor from default with the help this and passed the string in it

Output



3.Create a class with a constructor that takes a String argument. During construction, print the argument. Create an array of object references to this class, but don't actually create objects to assign into the array. When you run the program, notice whether the initialization messages from the constructor calls are printed.

-when the object references array of size 4 created and not assigned with object references then the constructor is not been called

4.Complete the previous exercise by creating objects to attach to the array of references.

-when objects are created and and attached to the reference of the array of object references then the constructors are been called