```csharp
using System;

namespace NumberToWords
{
    public class NumberToWordsConverter
    {
        private string [] numbers = { "Zero", "One", "Two", "Three", "Four", "Five", "Six", "Seven", "Eight", "Nine", "Ten", "Eleven", "Twelve", "Thirteen", "Fourteen", "Fifteen", "Sixteen", "Seventeen", "Eighteen", "Nineteen" };

        private string [] tens = { "", "Ten", "Twenty", "Thirty", "Forty", "Fifty", "Sixty", "Seventy", "Eighty", "Ninety" };

        public NumberToWordsConverter()
        { }

        public string ConvertToStringWithIf(int number)
        {
            char [] digits = number.ToString().ToCharArray();
            string words = null;

            if (number >= 0 && number <= 19)
            {
                words = words + numbers[number];
            }
            else if (number >= 20 && number <= 99)
            {
                int firstDigit = (int)char.GetNumericValue(digits[0]);
                int secondPart = number % 10;
```

```csharp
        words = words + tens[firstDigit];


        if (secondPart > 0)

        {

            words = words + " " + ConvertToStringWithIf(secondPart);

        }

    }

    else if (number >= 100 && number <= 999)

    {

        int firstDigit = (int)char.GetNumericValue(digits[0]);

        int secondPart = number % 100;


        words = words + numbers[firstDigit] + " hundred";


        if (secondPart > 0)

        {

            words = words + " and " + ConvertToStringWithIf(secondPart);

        }

    }

    else if (number >= 1000 && number <= 3000)

    {

        int firstPart = (int)char.GetNumericValue(digits[0]);

        if (number >= 10000)

        {

            string twoDigits = digits[0].ToString() + digits[1].ToString();

            firstPart = Convert.ToInt16(twoDigits);
```

```csharp
        }

        int secondPart = number % 1000;


        words = words + numbers[firstPart] + " thousand";


        if (secondPart > 0 && secondPart <= 99)

        {

            words = words + " and " + ConvertToStringWithIf(secondPart);

        }

        else if (secondPart >= 100)

        {

            words = words + " " + ConvertToStringWithIf(secondPart);

        }

    }


    return words;

}


public string ConvertToStringWithSwich(int number)

{

    char [] digits = number.ToString().ToCharArray();

    string words = null;

    int digitCount = (int)Math.Log10(number) + 1;


    switch (digitCount)

    {

        case 1:
```

```csharp
            words = words + numbers[number];
            break;
        case 2:
            int firstDigit = (int)char.GetNumericValue(digits[0]);
            int secondPart = number % 10;

            words = words + tens[firstDigit];

            if (secondPart > 0)
            {
                words = words + " " + ConvertToStringWithSwich(secondPart);
            }
            break;
        case 3:
            firstDigit = (int)char.GetNumericValue(digits[0]);
            secondPart = number % 100;

            words = words + numbers[firstDigit] + " hundred";

            if (secondPart > 0)
            {
                words = words + " and " + ConvertToStringWithSwich(secondPart);
            }
            break;
        case 4:
            int firstPart = (int)char.GetNumericValue(digits[0]);
```

```csharp
            if (number >= 10000)

            {

                string twoDigits = digits[0].ToString() + digits[1].ToString();

                firstPart = Convert.ToInt16(twoDigits);

            }


            secondPart = number % 1000;


            words = words + numbers[firstPart] + " thousand";


            if (secondPart > 0 && secondPart <= 99)

            {

                words = words + " and " + ConvertToStringWithSwich(secondPart);

            }

            else if (secondPart >= 100)

            {

                words = words + " " + ConvertToStringWithSwich(secondPart);

            }

            break;

        }


        return words;

    }

  }

}
```