```
using System;

namespace SubSet_of_Length_2
{
    class Program
    {
        static int EnterIntVal(string name)
        {
            string val;
            int v;

            do
            {
                Console.Write(name + " = ");
                val = Console.ReadLine();
            } while (!(int.TryParse(val, out v) && v > 1));

            return v;
        }

        static void Print(string[] set, bool last)
        {
            Console.Write("{ ");
            for (int i = 0; i < set.Length; i++)
            {
                Console.Write("{0}", set[i].ToString());

                if (i < set.Length - 1) Console.Write(", ");
                else Console.Write(" }");
            }

            if (!last) Console.Write(", ");
            else Console.WriteLine();
        }

        static string[] subSet1 = new string[1];

        static void SubSetLengthOfOne(string[] set, int crntElement, int nextElement)
        {
            if (crntElement + nextElement == set.Length)
            {
                //if (crntElement <= set.Length - 1) SubSetLengthOfOne(set, crntElement + 1, 0);
                return;
            }

            subSet1[0] = set[crntElement + nextElement];

            bool lastElement = (crntElement + nextElement < set.Length - 1) ? false : true;
            Print(subSet1, lastElement);

            SubSetLengthOfOne(set, crntElement, nextElement + 1);
        }

        static string[] subSet2 = new string[2];

        static void SubSetLengthOfTwo(string[] set, int crntElement, int nextElement)
        {

            if (crntElement + nextElement == set.Length)
            {
                if(crntElement <= set.Length - 2) SubSetLengthOfTwo(set, crntElement + 1, 1);
                return;
            }
```

```
            subSet2[0] = set[crntElement];
            subSet2[1] = set[crntElement + nextElement];

            bool lastElement = (crntElement < set.Length - 2) ? false : true;

            Print(subSet2, lastElement);

            SubSetLengthOfTwo(set, crntElement, nextElement + 1);
        }

        static string[] subSet3 = new string[3];

        static void SubSetLengthOfThree(string[] set, int crntElement, int nextElement)
        {
            if (crntElement + nextElement + 1 == set.Length)
            {
                if (crntElement <= set.Length - 3) SubSetLengthOfThree(set, crntElement + 1, 1);
                return;
            }

            subSet3[0] = set[crntElement];
            subSet3[1] = set[crntElement + nextElement];

            for (int i = nextElement + 1; i + crntElement < set.Length; i++)
            {
                subSet3[2] = set[crntElement + i];
                bool lastElement = (crntElement < set.Length - 3) ? false : true;
                Print(subSet3, lastElement);
            }

            SubSetLengthOfThree(set, crntElement, nextElement + 1);
        }

        static void Main()
        {
            int s = EnterIntVal("s");
            string[] set = new string[s];

            for (int i = 0; i < s; i++) set[i] = i.ToString();

            Print(set, true);

            SubSetLengthOfOne(set, 0, 0);

            SubSetLengthOfTwo(set, 0, 1);

            SubSetLengthOfThree(set, 0, 1);
        }
    }
}
```

```
static string[] subSet1 = new string[1];

static void SubSetLengthOfOne(string[] set, int crntElement, int nextElement)
{
	if (crntElement + nextElement == set.Length)
	{
		//if (crntElement <= set.Length - 1) SubSetLengthOfOne(set, crntElement + 1, 0);
		return;
	}

	subSet1[0] = set[crntElement + nextElement];

	bool lastElement = (crntElement + nextElement < set.Length - 1) ? false : true;
	Print(subSet1, lastElement);

	SubSetLengthOfOne(set, crntElement, nextElement + 1);
}
```

```
static string[] subSet2 = new string[2];

static void SubSetLengthOfTwo(string[] set, int crntElement, int nextElement)
{

	if (crntElement + nextElement == set.Length)
	 {
		if(crntElement <= set.Length - 2) SubSetLengthOfTwo(set, crntElement + 1, 1);
		return;
	}


	 subSet2[0] = set[crntElement];
	 subSet2[1] = set[crntElement + nextElement];

	 bool lastElement = (crntElement < set.Length - 2) ? false : true;

	 Print(subSet2, lastElement);

	SubSetLengthOfTwo(set, crntElement, nextElement + 1);
}
```

```
static string[] subSet3 = new string[3];

static void SubSetLengthOfThree(string[] set, int crntElement, int nextElement)
{
	if (crntElement + nextElement + 1 == set.Length)
	 {
		if (crntElement <= set.Length - 3) SubSetLengthOfThree(set, crntElement + 1, 1);
		return;
	}

		subSet3[0] = set[crntElement];
		subSet3[1] = set[crntElement + nextElement];

	for (int i = nextElement + 1; i + crntElement < set.Length; i++)
	 {
		subSet3[2] = set[crntElement + i];
		  bool lastElement = (crntElement < set.Length - 3) ? false : true;
		  Print(subSet3, lastElement);
	}

	 SubSetLengthOfThree(set, crntElement, nextElement + 1);
}
```