Name : Welly
E-mail : wellytan09@gmail.com

```
public class Rule
{
	public int Number { get; }
	public string Message { get; }

	public Rule(int number, string message)
	{
		Number = number;
		Message = message;
	}
}

public class A
{
	private readonly List<Rule> rules;

	public A()
	{
		rules = new List<Rule>
		{
			new Rule(3, "foo"),
			new Rule(4, "bazz"),
			new Rule(5, "bar"),
			new Rule(7, "jazz"),
			new Rule(9, "huzz")
		};
	}

	public void AddRule(int number, string message)
	{
		// Check if the rule already exists and update it
		for (int i = 0; i < rules.Count; i++)
		{
```

```
                if (rules[i].Number == number)
                {
                        rules[i] = new Rule(number, message);
                        return;
                }
        }

        // If rule does not exist, add it
        rules.Add(new Rule(number, message));
    }

    public void GenerateOutput(int input)
    {
        for (int i = 1; i <= input; i++)
        {
            string print = "";

            // Check for custom rules
            foreach (var rule in rules)
            {
                if (i % rule.Number == 0)
                {
                    print += rule.Message;
                }
            }

            // Default behavior if no custom rule matches
            if (print == "")
            {
                print = i.ToString();
            }

            Console.WriteLine(print);
        }
    }
}

class Program
{
    static void Main()
```

```
{
        A instance = new A();

        while (true)
        {
                Console.WriteLine("Enter a number to generate output:");
                string userInput = Console.ReadLine();

                if (int.TryParse(userInput, out int result))
                {
                        instance.GenerateOutput(result);
                }
                else
                {
                        Console.WriteLine("Please enter a valid integer.\n");
                        continue; // Restart the loop to ask for a number
                }

                while (true)
                {
                        Console.WriteLine("\nDo you want to add a rule (Y/N):");
                        string userRule = Console.ReadLine();

                        if (userRule.ToUpper() == "Y")
                        {
                                Console.WriteLine("\n==================New Rule====================");
                                while (true)
                                {
                                        Console.WriteLine("\nPlease enter a number for the new rule: ");
                                        string userRuleNumber = Console.ReadLine();

                                        if (int.TryParse(userRuleNumber, out int newRuleNumber))
                                        {
                                                Console.WriteLine("Please enter a message for the rule: ");
                                                string userRuleMessage = Console.ReadLine();

                                                instance.AddRule(newRuleNumber, userRuleMessage);
                                                Console.WriteLine("\n================New Rule Added================");
                                                break; // Break out of the inner loop to ask about another rule
                                        }
```

```
                                        else
                                        {
                                                Console.WriteLine("Please enter a valid integer.");
                                        }
                                }
                        }
                        else if (userRule.ToUpper() == "N")
                        {
                                break;
                        }

                        else
                        {
                                Console.WriteLine("Please enter Y or N.");
                        }
                }

        }
    }
}
```

https://github.com/DoomedMean/Company-Test/tree/main/Formatrix