

/*1. Faça um programa para controlar as despesas e dados de um condomínio com 12 apartamentos. Os dados de cada apartamento são armazenados em um vetor de registros. Para cada apartamento tem-se os seguintes dados:

- a) nome (do responsável)
- b) número (do apartamento)
- c) área (em m²)
- d) número de moradores
- e) valor (a ser pago no mês)

O programa deve:

ler os dados dos apartamentos (exceto o campo valor, pois este deve ser calculado pelo sistema);

calcular e mostrar a área total do condomínio;

ler o total de despesas do condomínio e calcular para cada apartamento o valor a ser pago no mês (esse valor é proporcional à área do apartamento);

mostrar os dados do apartamento que tem o maior número de moradores. Em caso de haver mais de um, imprimir todos.

mostrar os dados de todos os apartamentos.*/*

```
struct Apartamento
```

```
{
```

```
    public string nome;
```

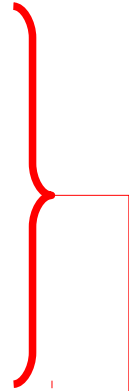
```
    public int numero;
```

```
    public double area;
```

```
    public int moradores;
```

```
    public decimal valor;
```

```
};
```



campos do struct Apartamento

```
static void Main(string[] args)
{
    const int TAM = 3;
    double area;
    int maior;
    Apartamento[] condominio = new Apartamento[TAM];
    lerDados(condominio);
    Console.Clear();
    area = calculaArea(condominio);
    calculaDespesa(condominio, area);
    maior = encontraMaior(condominio);
    mostraMaior(condominio, maior);
    mostraCondominio(condominio);
    Console.ReadKey();
}
```

```
static void Main(string[] args)
```

```
{
```

```
    const int TAM = 3;
```

```
    double area;
```

```
    int maior;
```

```
    Apartamento[] condominio = new Apartamento[TAM];
```

```
    lerDados(condominio);
```

```
    Console.Clear();
```

```
are static void lerDados(Apartamento[] c)
```

```
{
```

```
    int x;
```

```
    for (x = 0; x < c.Length; x++)
```

```
{
```

```
        Console.Write("NOME: ");
```

```
        c[x].nome = Console.ReadLine();
```

```
        Console.Write("NUMERO: ");
```

```
        c[x].numero = leNumero(c);
```

```
        Console.Write("AREA: ");
```

```
        c[x].area = Convert.ToDouble(Console.ReadLine());
```

```
        Console.Write("NUMERO DE MORADORES: ");
```

```
        c[x].moradores = Convert.ToInt16(Console.ReadLine());
```

```
        Console.WriteLine();
```

```
    }
```

```
}
```

```
static void lerDados(Apartamento[] c)
```

```
{
```

```
    int x;
```

```
    for (x = 0; x < c.Length; x++)
```

```
    {
```

```
        Console.Write("NOME: ");
```

```
        c[x].nome = Console.ReadLine();
```

```
        Console.Write("NUMERO: ");
```

```
        c[x].numero = leNumero(c);
```

```
        Console.Write("AREA: ");
```

```
        c[x].area = Convert.ToDouble(Console.ReadLine());
```

```
static int leNumero(Apartamento[] c)
```

```
{
```

```
    int numero = 0, flag = 1;
```

```
    while (flag == 1)
```

```
    {
```

```
        numero = Convert.ToInt16(Console.ReadLine());
```

```
        flag = verificaNumeroAp(c, numero);
```

```
        if (flag == 1)
```

```
            Console.Write("Número de apartamento já
```

```
cadastrado! Digite o numero novamente: ");
```

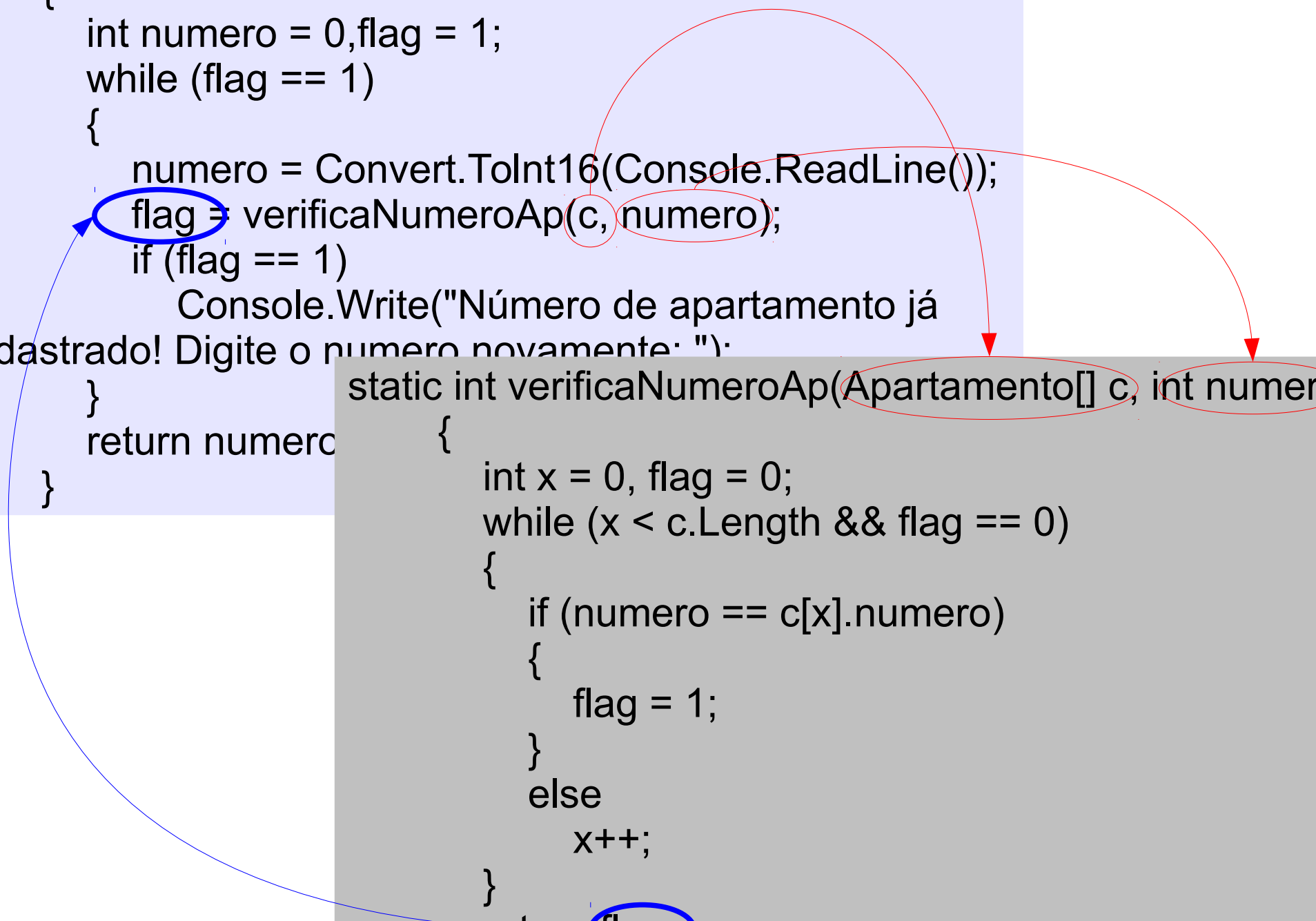
```
    }
```

```
    return numero;
```

```
}
```

```
static int leNumero(Apartamento[] c)
{
    int numero = 0, flag = 1;
    while (flag == 1)
    {
        numero = Convert.ToInt16(Console.ReadLine());
        flag = verificaNumeroAp(c, numero);
        if (flag == 1)
            Console.WriteLine("Número de apartamento já cadastrado! Digite o numero novamente.");
    }
    return numero;
}
```

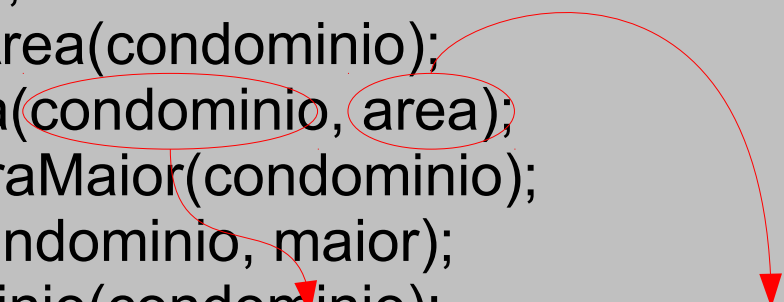
```
static int verificaNumeroAp(Apartamento[] c, int numero)
{
    int x = 0, flag = 0;
    while (x < c.Length && flag == 0)
    {
        if (numero == c[x].numero)
        {
            flag = 1;
        }
        else
            x++;
    }
    return flag;
}
```



```
static void Main(string[] args)
{
    const int TAM = 3;
    double area;
    int maior;
    Apartamento[] condominio = new Apartamento[TAM];
    lerDados(condominio);
    Console.Clear();
    area = calculaArea(condominio);
    calculaDespesa(condominio, area);
    maior = encontraMaior(condominio);
    mostraMaior(condominio, maior);
    mostraCondominio(condominio);
    Console.ReadKey();
}
```

```
static double calculaArea(Apartamento[] c)
{
    int x;
    double soma = 0;
    for (x = 0; x < c.Length; x++)
    {
        soma += c[x].area;
    }
    return soma;
}
```

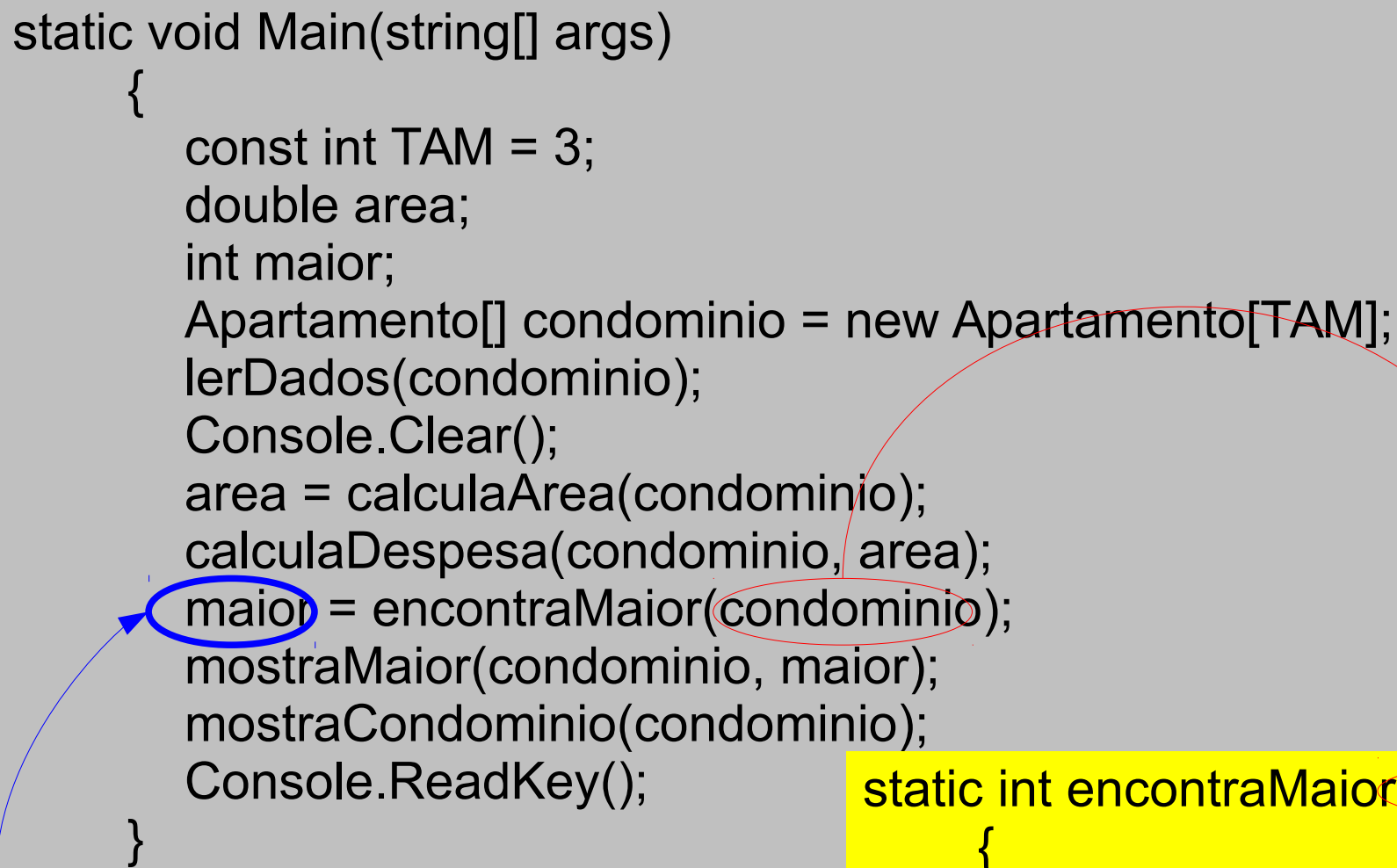
```
static void Main(string[] args)
{
    const int TAM = 3;
    double area;
    int maior;
    Apartamento[] condominio = new Apartamento[TAM];
    lerDados(condominio);
    Console.Clear();
    area = calculaArea(condominio);
    calculaDespesa(condominio, area);
    maior = encontraMaior(condominio);
    mostraMaior(condominio, maior);
    mostraCondominio(condominio);
}
```



```
static void calculaDespesa(Apartamento[] c, double areaTotal)
{
    decimal despesaGeral;
    int x;
    Console.WriteLine("DESPESA GERAL: ");
    despesaGeral = Convert.ToDecimal(Console.ReadLine());
    for (x = 0; x < c.Length; x++)
    {
        c[x].valor = (decimal)((((double)despesaGeral / areaTotal) * c[x].area);
    }
}
```



```
static void Main(string[] args)
{
    const int TAM = 3;
    double area;
    int maior;
    Apartamento[] condominio = new Apartamento[TAM];
    lerDados(condominio);
    Console.Clear();
    area = calculaArea(condominio);
    calculaDespesa(condominio, area);
    maior = encontraMaior(condominio);
    mostraMaior(condominio, maior);
    mostraCondominio(condominio);
    Console.ReadKey();
}
```



```
static int encontraMaior(Apartamento[] c)
{
    int maior = c[0].moradores, x;
    for (x = 1; x < c.Length; x++)
    {
        if (maior < c[x].moradores)
            maior = c[x].moradores;
    }
    return maior;
}
```

```
static void Main(string[] args)
```

```
{
```

```
    const int TAM = 3;
```

```
    double area;
```

```
    int maior;
```

```
    Apartamento[] condominio = new Apartamento[TAM];
```

```
    lerDados(condominio);
```

```
    Console.Clear();
```

```
    area = calculaArea(condominio);
```

```
    calculaDespesa(condominio, area);
```

```
    maior = encontraMaior(condominio);
```

```
    mostraMaior(condominio, maior);
```

```
    mostraCondominio(condominio);
```

```
    Console.ReadKey();
```

```
}
```

```
static void mostraMaior(Apartamento[] c, int maior)
```

```
{
```

```
    int x;
```

```
    Console.WriteLine("\nDados dos apartamentos  
com maior quantidade de moradores:\n");
```

```
    for (x = 0; x < c.Length; x++)
```

```
    {
```

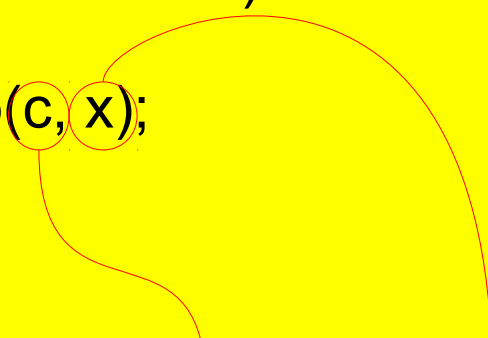
```
        if (maior == c[x].moradores)
```

```
        {
```

```
            mostraRegistro(c, x);
```

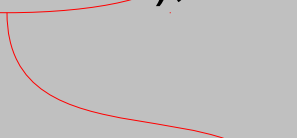
```
        }
```

```
static void mostraMaior(Apartamento[] c, int maior)
{
    int x;
    Console.WriteLine("\nDados dos apartamentos com maior quantidade de moradores:\n");
    for (x = 0; x < c.Length; x++)
    {
        if (maior == c[x].moradores)
        {
            mostraRegistro(c, x);
        }
    }
}
```



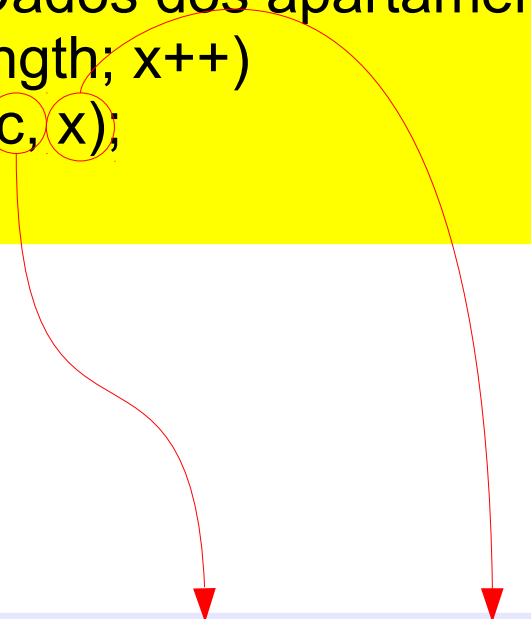
```
static void mostraRegistro(Apartamento[] c, int x)
{
    Console.WriteLine("\nNOME: "+ c[x].nome);
    Console.WriteLine("\nNUMERO: "+ c[x].numero);
    Console.WriteLine("\nAREA: "+ c[x].area);
    Console.WriteLine("\nNUMERO DE MORADORES: "+ c[x].moradores);
    Console.WriteLine("\nVALOR: {0:C}", c[x].valor);
    Console.WriteLine();
}
```

```
static void Main(string[] args)
{
    const int TAM = 3;
    double area;
    int maior;
    Apartamento[] condominio = new Apartamento[TAM];
    lerDados(condominio);
    Console.Clear();
    area = calculaArea(condominio);
    calculaDespesa(condominio, area);
    maior = encontraMaior(condominio);
    mostraMaior(condominio, maior);
    mostraCondominio(condominio);
    Console.ReadKey();
}
```



```
static void mostraCondominio(Apartamento[] c)
{
    int x;
    Console.WriteLine("\nDados dos apartamentos do condomínio:\n");
    for (x = 0; x < c.Length; x++)
        mostraRegistro(c, x);
}
```

```
static void mostraCondominio(Apartamento[] c)
{
    int x;
    Console.Write("\nDados dos apartamentos do condomínio:\n");
    for (x = 0; x < c.Length; x++)
        mostraRegistro(c, x);
}
```



```
static void mostraRegistro(Apartamento[] c, int x)
{
    Console.Write("\nNOME: "+ c[x].nome);
    Console.Write("\nNUMERO: "+ c[x].numero);
    Console.Write("\nAREA: "+ c[x].area);
    Console.Write("\nNUMERO DE MORADORES: "+ c[x].moradores);
    Console.Write("\nVALOR: {0:C}", c[x].valor);
    Console.WriteLine();
}
```