

## Microsoft Visual C# 2012 konsolinės programos kūrimas

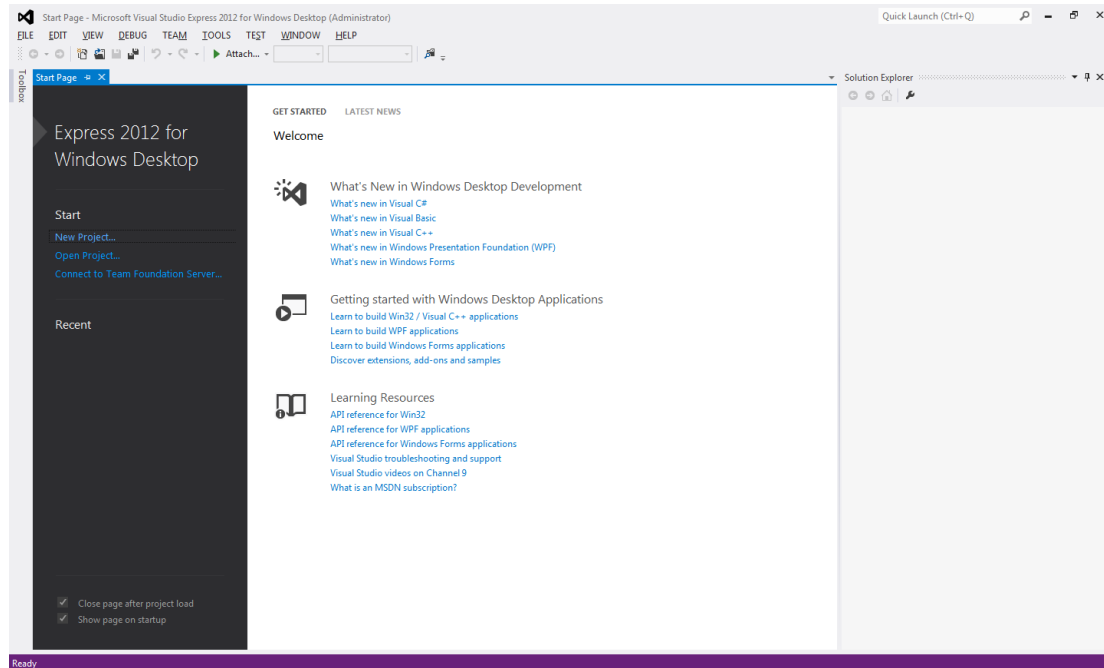
Susipažinsite su Microsoft Visual C# programavimo aplinka konsolinei programai sukurti:

- programavimo aplinkos konsoliniu darbo režimu;
- programos struktūra.

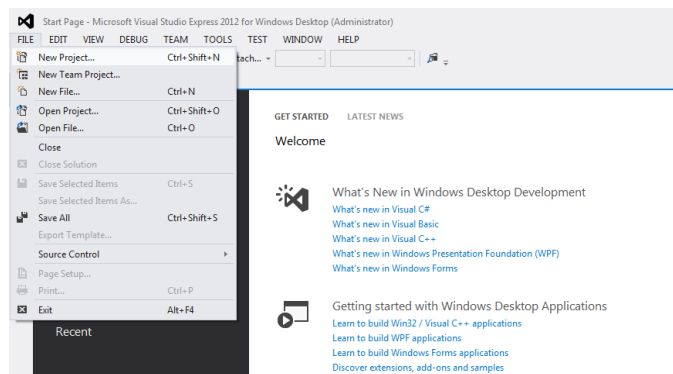
### Konsolinės programos sukūrimas

**Pirmas žingsnis.** Programos sukūrimo scenarijus

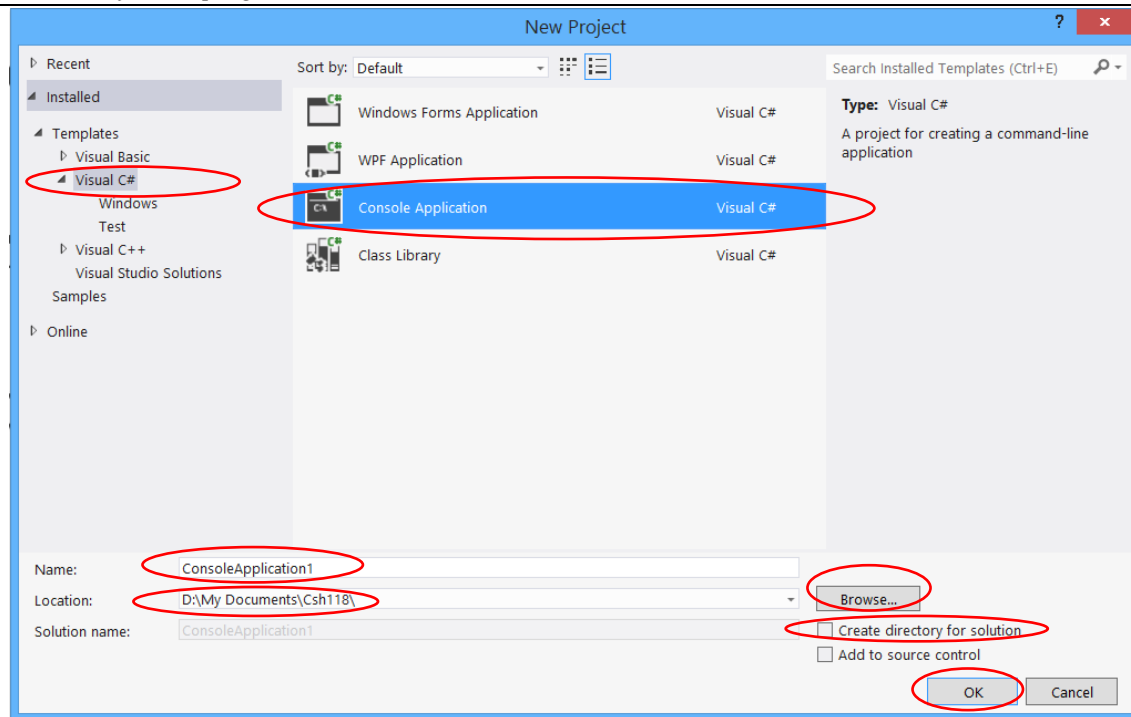
- Paleiskite programą **Microsoft Visual Studio Express 2012**.



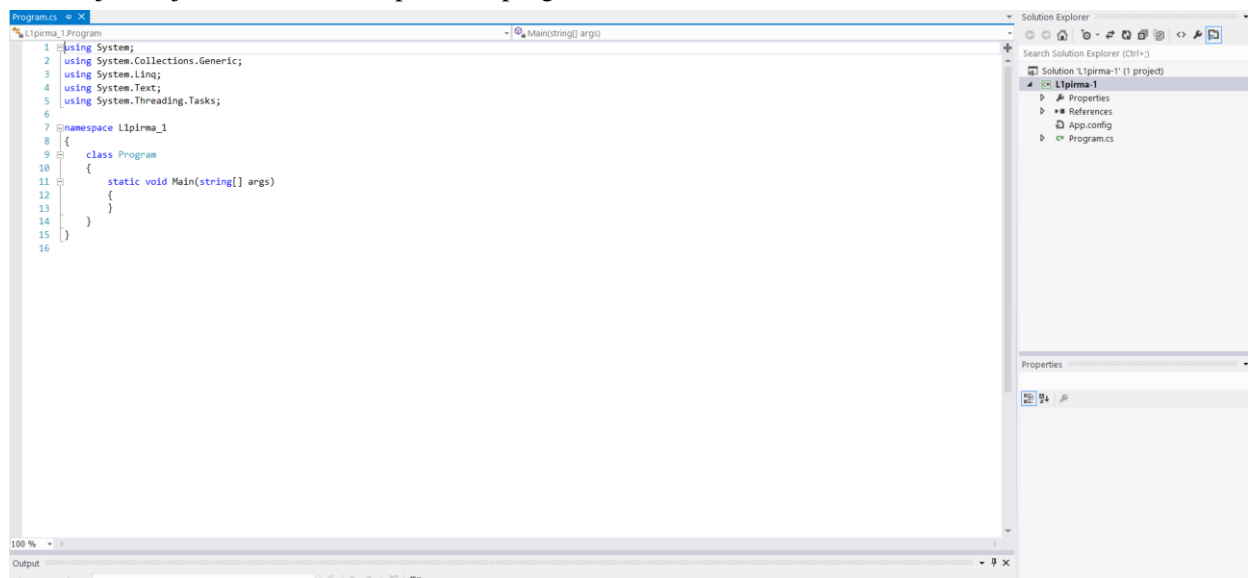
- Sukurkite naują projektą (naujos programos kūrimas **Microsoft Visual Studio Express** aplinkoje visada prasideda nuo projekto kūrimo):
- **FILE** meniu išsirinkite **New Project ...**



- Kairėje lango pusėje pasirinkite programavimo kalbą **Visual C#**. Dešinėje lango pusėje iš projektų tipų pasirinkite **Console Application Visual C#**.
- Įrašykite naujo projekto vardą `L1pirma-1`.
- Pasirinkite diską ir katalogą, kur saugosite programą: paspauskite mygtuką **Browse** ir išsirinkite **D** diską, susietą su konkrečiu kompiuteriu. Programas derinimo metu reikia saugoti **D** diske. Pastoviam saugojimui reikia naudoti **Z** diską. **Z** diskas – tai jūsų asmeninė sritis serveryje. Programos saugojimas derinimo metu **Z** diske labai apkrauna tinklą.

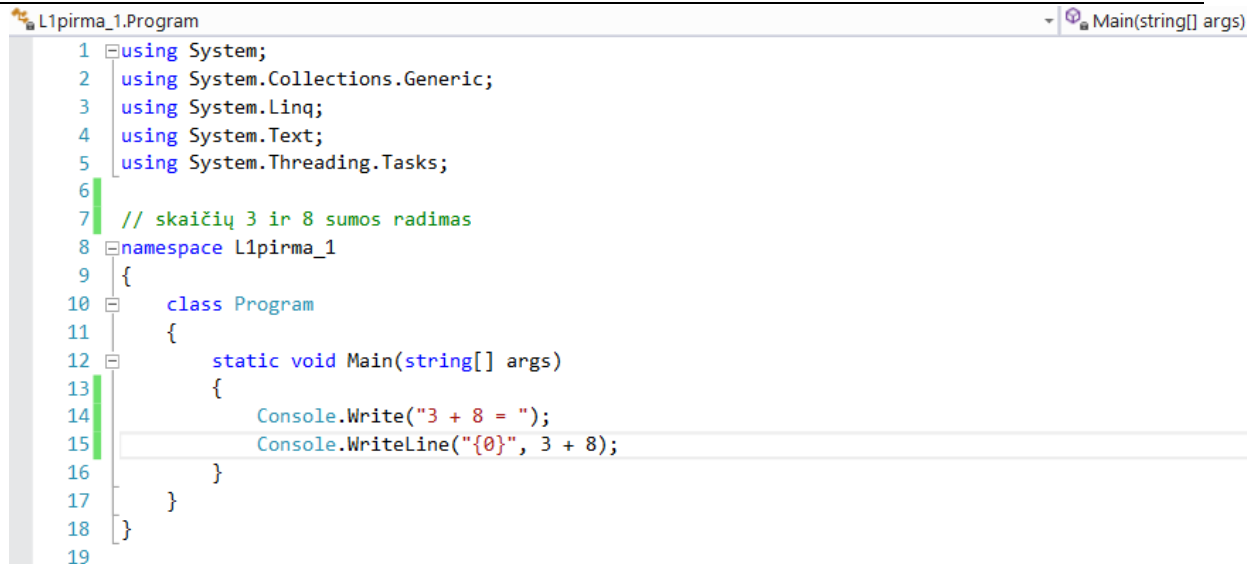


- Nuimkite varnelę **Create directory for solution** (papildomi katalogai nereikalingi).
- Patvirtinkite pasirinkimą mygtuku **OK**.
- Projektas jau sukurtas ir turi pradinės programos kodo eilutes:



## ➊ Antras žingsnis. Pirmoji programa.


- Klaviatūra įveskite programos tekstą, kuris yra paveiksle:

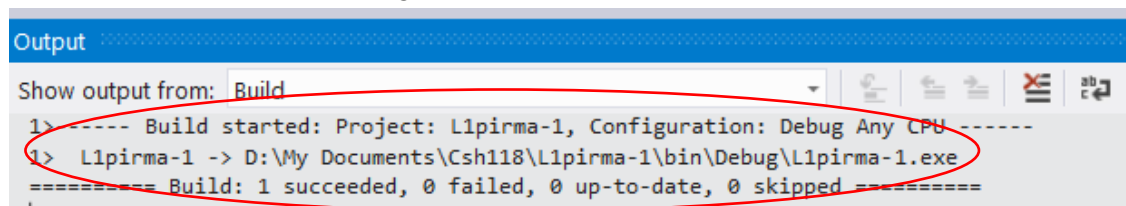


```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 // skaičių 3 ir 8 sumos radimas
8 namespace L1pirma_1
9 {
10     class Program
11     {
12         static void Main(string[] args)
13         {
14             Console.Write("3 + 8 = ");
15             Console.WriteLine("{0}", 3 + 8);
16         }
17     }
18 }
19

```

- Išsaugokite įvestą programą: paspauskite **Save All** piktogramą .
- Suskaičiuokite laukiamą rezultatą:  $3 + 8 = 11$ .
- Kompiliuokite programą: **Build** meniu paspauskite **Build Solution**.
- Visual C# lango **Output** dalyje matysite kompiliavimo eigos pranešimus. Jei įvedimo metu nepadarėte klaidų, pranešimų pabaigoje bus nurodyta, kad kompiliavimas ir vykdomojo programos failo sukūrimas buvo sėkmingas:



```

Output
Show output from: Build
1> ----- Build started: Project: L1pirma-1, Configuration: Debug Any CPU -----
1> L1pirma-1 -> D:\My Documents\Csh118\L1pirma-1\bin\Debug\L1pirma-1.exe
===== Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped =====

```

- Įvykdysite programą: **Debug** meniu paspauskite **Start Without Debugging**.
- Palyginkite gautą rezultatą su laukiamu.
- Pasižiūrėkite į katalogą L1pirma-1 ir nustatykite, kokius failus sukuria sistema.
- Baikite darbą su programavimo aplinka **File** meniu paspausdami **Exit**.
- Iš disko **D** pašalinkite katalogą L1pirma-1.

### ☛ Trečias žingsnis. Sintaksės klaidų taisymas, diagnostinių pranešimų analizė.

- Sukurkite projektą L1pirma-2, kurį išsaugosite **D** diske.
- Įveskite komentarus programai virš `namespace L1pirma-2`:

```

//-----
// 1.2 pvz. Dviejų kintamųjų reikšmių sumos radimas
// Dėmesio: programoje yra klaidų!
//-----

```

- Įveskite programos kodą:

```

static void Main(string[] args)
{
    int a = 3,          // pirmasis kintamasis
    int b = 8;          // antrasis kintamasis
    int suma = a + b;   // skaičių sumai saugoti
    Console.Write("{0} + {1} = ", a, b);
    Console.WriteLine("{0}", suma);
}

```

Pirmoji klaida –  
reikia kabliataškio

**Pastaba.** "{0} + {1} = " – nurodomas spausdinamas tekstas su specialiais pažymėjimais {0}, {1} įterpti kintamųjų reikšmėms. Po jo išvardinami kintamieji. Jie numeruojami nuo 0: {0} atitinka *a*, o {1} – *b*.

- Išsaugokite įvestą programą.
- Suskaičiuokite laukiamą rezultatą.

- Kompiliuokite programą.
- Išanalizuokite pranešimus apie klaidas. Taisykite klaidas po vieną. Programoje yra 3 klaidos (trūksta 2 kabliataškių, `int` tipas privalo prasidėti mažąja raide).
- Remdamiesi diagnostinių pranešimų informacija, suraskite ir ištaisykite programos tekstą. Kompiliuokite programą ištaisę kiekvieną klaidą.
- Išsaugokite ir įvykdysite pataisytą programą. Palyginkite gautą rezultatą su laukiamu.
- Baikite darbą su sistema.
- Katalogą `L1pirma-2` iš disko **D** perkeltkite į diską **Z**.

#### ❶ Ketvirtas žingsnis. Sveikųjų reikšmių įvedimas

- Sukurkite projektą `L1antra-1`, kurį išsaugosite **D** diske.
- Įveskite tokios programos tekstą:

```
//-----
// sveikųjų reikšmių įvedimas
//-----
static void Main(string[] args)
{
    int a;           // pirmasis kintamasis
    int b;           // antrasis kintamasis
    int suma;        // kintamasis skaičių sumai saugoti
    Console.WriteLine("Įveskite sveikąją a reikšmę");
    a = int.Parse(Console.ReadLine()); // tekstą konvertuoja į int tipo skaičių
    Console.WriteLine("Įveskite sveikąją b reikšmę:");
    b = int.Parse(Console.ReadLine());
    suma = a + b;
    Console.WriteLine("{0} + {1} = {2}", a, b, suma);
}
```

**Pastaba.** `(int)Parse` – nagrinėja tekstą ir atpažįsta jame skaičių.

- Išsaugokite įvestą programą.
- Kompiliuokite ir įvykdysite programą. Vykdydami įveskite tokias kintamųjų reikšmes: 10 8.
- Pakartokite sumos skaičiavimą su kitomis reikšmėmis.
- Baikite darbą su sistema.
- Katalogą `L1antra-1` iš disko **D** perkeltkite į diską **Z**.

#### ❷ Penktas žingsnis. Realųjų reikšmių įvedimas ir rezultatų tikrinimas

- Sukurkite projektą `L1antra-2`, kurį išsaugokite **D** diske.
- Įveskite nupjautinio kūgio tūrio skaičiavimo programos tekstą:

```
//-----
// Programa skaičiuoja kūgio tūrį V, kai kūgio pagrindo
// spinduliai R, r ir aukštinė H įvedami klaviatūra
//-----
static void Main(string[] args)
{
    double pi = 3.1415;           // matematinė konstanta
    double H;                     // kūgio aukštinė
    double R, r;                  // kūgio pagrindo spinduliai
    double V;                     // kūgio tūris
    Console.WriteLine("Įveskite kūgio aukštinės reikšmę");
    H = double.Parse(Console.ReadLine());
    Console.WriteLine("Įveskite kūgio viršutinio pagrindo spindulio reikšmę");
    r = double.Parse(Console.ReadLine());
    Console.WriteLine("Įveskite kūgio apatinio pagrindo spindulio reikšmę");
    R = double.Parse(Console.ReadLine());
    V = (1.0 / 3) * pi * H * (R * R + R * r + r * r);
    Console.WriteLine("Kūgio tūris = {0, 5:f}", V);
}
```

**Pastaba.** {0, 5:f} – nurodyto numerio elementas, nurodant formatavimą: 5 – visas lauko ilgis, f – realus skaičius. Realių skaičių formatavimo bendra išraiška **n:fm**, čia n – visas lauko ilgis, m – trupmeninės dalies lauko ilgis. Praleidus m, spausdinami 2 skaitmenys po kablelio.

- Išsaugokite įvestą programą.
  - Kompiliuokite ir įvykdysite programą. Vykdydami įveskite tokias duomenų reikšmes: aukštinė: 10.5, pagrindų spinduliai: 8.5 ir 2.3. Palyginkite programos gautą kūgio tūrio reikšmę su reikšme, suskaičiuota skaičiuotuvu.
  - Pakartokite kūgio tūrio skaičiavimą, kai aukštinė lygi 15, o pagrindų spinduliai – 20.5 ir 6.4. Gautą reikšmę palyginkite su reikšme, suskaičiuota skaičiuotuvu.
  - Baikite darbą su sistema.
  - Katalogą Llantra-2 iš disko **D** perkeltkite į diską **Z**.
-



**Pavyzdys:**

```
//=====
// PROGRAMOS PAVYZDYS: Studentai
// Skaitymi studentų duomenys iš failo į objektų masyvą, o po to
// spausdinami lentelė rezultatų faile
//=====
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
//=====
using System.IO;      // reikalinga skaitymui iš failo (StreamReader)
//=====
namespace Studentai
{
    //=====
    /** Klasė studento duomenims saugoti
    @class Studentas */
    class Studentas
    {
        private string pavVrd;    // studento pavardė ir vardas
        private int pazym;        // pažymys (įvertinimas)
        /** Studento duomenys:
        @param pavvrd studento pavardė ir vardas
        @param pazym pažymys */
        public Studentas(string pavv, int pazym)
        {
            pavVrd = pavv;
            this.pazym = pazym;
        }
        /** grąžina studento pavardę ir vardą */
        public string ImtiPavv()
        {
            return pavVrd;
        }
        /** grąžina pažymį */
        public int ImtiPazym()
        {
            return pazym;
        }
    }
    //=====
    class Program
    {
        //=====
        const int Cn = 500;                // maksimalus studentų skaičius
        const string CFd = "..\\..\\Studentai.txt"; // duomenų failo vardas
        const string CFr = "..\\..\\Rezultatai.txt"; // rezultatų failo vardas

        //=====
        static void Main(string[] args)
        {
            //=====
            Studentas[] Studentai = new Studentas[Cn]; // studentų objektų masyvas
            int kiek;                                // studentų skaičius

            //=====

            if (File.Exists(CFr))
                File.Delete(CFr); // Failo pašalinimas
            SkaitytiStudentus(CFd, Studentai, out kiek);
            SpausdintiStudentus(CFr, Studentai, kiek, "Studentų sąrašas (masyvas)");

            //=====
            Console.WriteLine("\n\nProgramos rezultatus žiūrėkite rezultatų faile.\n\n");
        }
    }
}
```

```

//=====
/** Skaityti VISI duomenys iš failo
    @param fv duomenų failo vardas
    @param Studentai objektų rinkinys studentų duomenims saugoti
    @param kiek - studentų skaičius */
static void SkaitytiStudentus(string fv, Studentas[] Studentai, out int kiek)
{
    using (StreamReader srutas = new StreamReader(fv))
    {
        string eilute; // visa duomenų failo eilutė
        int i = 0;
        while ((eilute = srutas.ReadLine()) != null && (i < Cn))
        {
            string[] eilDalis = eilute.Split(';'); // failo eilutės dalys
            string pavVrd = eilDalis[0];
            int pazym = int.Parse(eilDalis[1]);
            Studentai[i++] = new Studentas(pavVrd, pazym);
        }
        kiek = i;
    }
}

/** Spausdinami masyvo Studentai(kiek) duomenys lentele FAILE fv
    @param fv rezultatų failo vardas
    @param Studentai objektų rinkinys studentų duomenims saugoti
    @param kiek - studentų skaičius
    @param antraste - užrašas virš lentelės */
static void SpausdintiStudentus(string fv, Studentas[] Studentai,
                                int kiek, string antraste)
{
    const string virsus =
        "-----\n"
        + " Nr. Pavardė ir vardas Pažymys \n"
        + "-----";
    using (var fr = File.CreateText(fv))
    {
        fr.WriteLine("\n" + antraste);
        fr.WriteLine(virsus);
        for (int i = 0; i < kiek; i++)
            fr.WriteLine("{0, 3} {1, -20} {2, 2}",
                          i + 1, Studentai[i].ImtiPavv(),
                          Studentai[i].ImtiPazym());
        fr.WriteLine("-----\n");
    }
}
//=====
} // pabaiga class Program
} // pabaiga namespace Studentai

```

**Duomenų failo (Studentai.txt) pavyzdys:**

```

Jonaitis Jonas;      8;
Petraitis Petras;    7;
Antanaitis Antanas; 10;
Giedraitis Giedrius; 5;
Onaitytė Ona;        8;
Juozaitis Juozas;    4;
Ramunaitė Ramunė;    5;

```

**Rezultatų failo (Rezultatai.txt) pavyzdys:**

```

Studentų sąrašas (masyvas)
-----
Nr. Pavardė ir vardas Pažymys
-----
1 Jonaitis Jonas      8
2 Petraitis Petras    7
3 Antanaitis Antanas 10
4 Giedraitis Giedrius 5
5 Onaitytė Ona        8
6 Juozaitis Juozas    4
7 Ramunaitė Ramunė    5
-----

```