**Handson-1**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading;

using System.Threading.Tasks;

namespace AsyncHandson

{

class Program

{

public static async Task first()

{

await Task.Run(() =>

{

string value = second();

Console.WriteLine(value);

});

}

public static string second()

{

Console.WriteLine("Waiting to return");

Thread.Sleep(7000);

return "Second";

}

static void Main(string[] args)

{

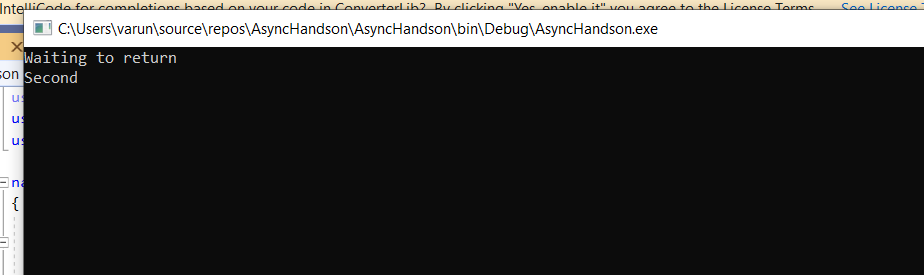
first();

Console.ReadKey();

}

}

}



**Handson 2**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace WindowsFormsApp1

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

public int CountChars()

{

int count = 0;

using (StreamReader streamReader = new StreamReader("C:\\Users\\varun\\Downloads\\varun.txt"))

{

string content = streamReader.ReadToEnd();

count = content.Length;

Thread.Sleep(2000);

}

return count;

}

private void Form1\_Load(object sender, EventArgs e)

{

}

private void label1\_Click(object sender, EventArgs e)

{

}

private async void button1\_ClickAsync(object sender, EventArgs e)

{

Task<int> task = new Task<int>(CountChars);

task.Start();

label1.Text = "File is processing";

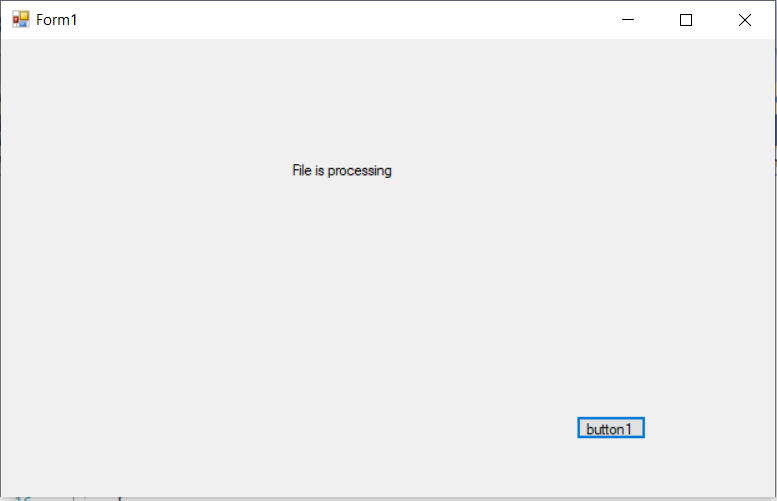
int count = await task;

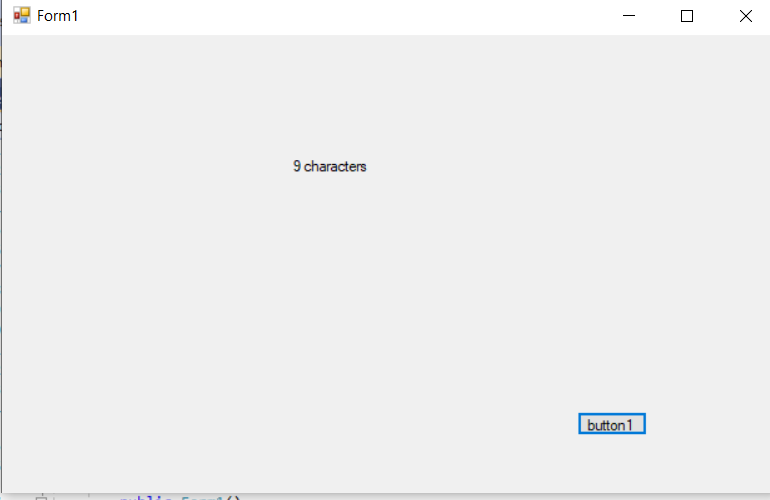
label1.Text = count.ToString() + " characters";

}

}

}





**Handson-3**

using System;

namespace ConsoleApp6

{

enum Mode

{

OBL,PARC

}

enum Track

{

JAVA,

DOTNET

}

class Program

{

public static void GetCohortDetails(string cohortName, int gencCount, Mode mode, Track track, string currentModule)

{

Console.WriteLine("It is " + cohortName + " with " + gencCount + " GenCs undergoing training for " + track + " thru " + mode + ". The current module of training is " + currentModule);

}

static void Main(string[] args)

{

GetCohortDetails(cohortName: "CDE", gencCount: 16, mode: Mode.OBL, track: Track.DOTNET, currentModule: "dotnet");

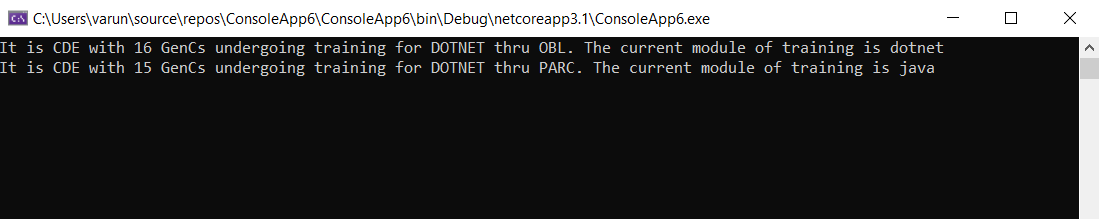
GetCohortDetails("CDE", 15, Mode.PARC, Track.DOTNET, "java");

Console.ReadKey();

}

}

}



**Handson-4**

using System;

namespace ConsoleApp7

{

class Program

{

public static void OrderDetails(string sellerName, string productName, int orderQuantity = 1, bool isReturnable = true)

{

Console.WriteLine($"Here is the order details –" + orderQuantity + " number of " + productName + " by " + sellerName + " is ordered. It’s returnable status is " + isReturnable);

}

static void Main(string[] args)

{

OrderDetails("Varun Raj", "Book");

Console.ReadKey();

}

}

}

