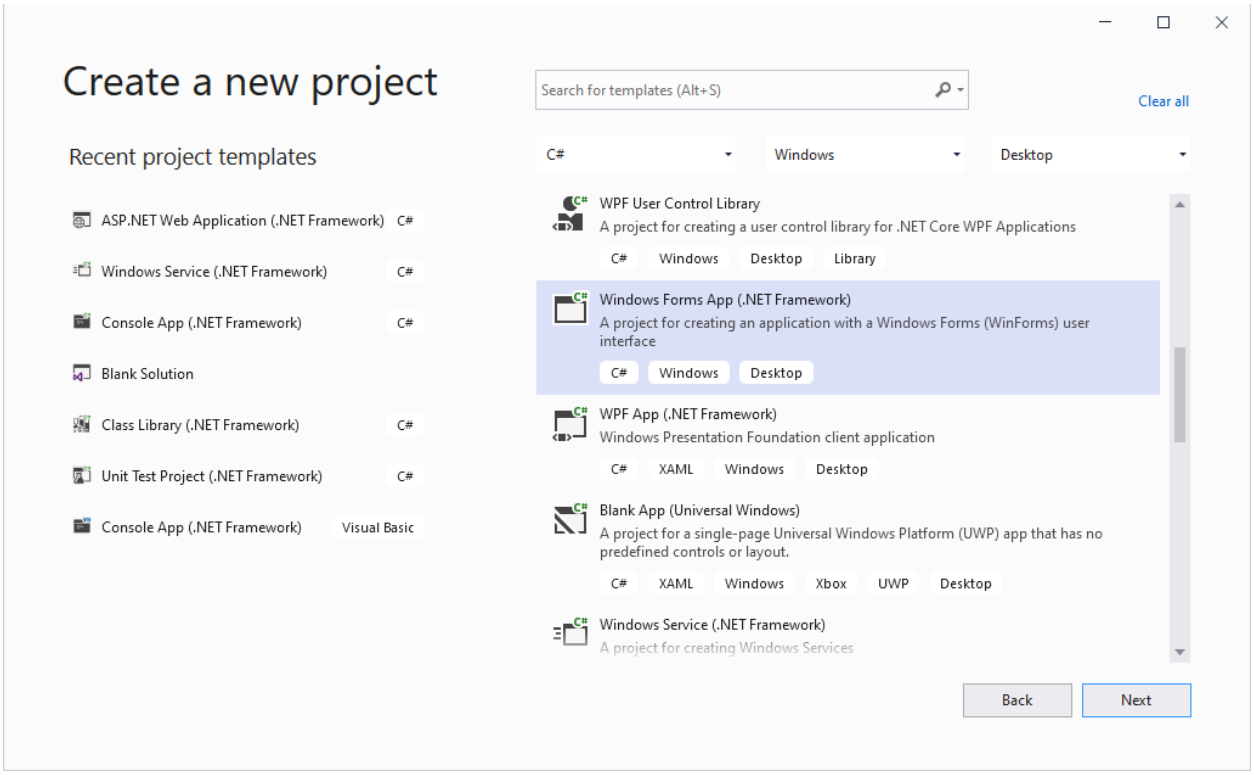
# Hands-On: Job Scheduling through Quartz in C#

### Objectives

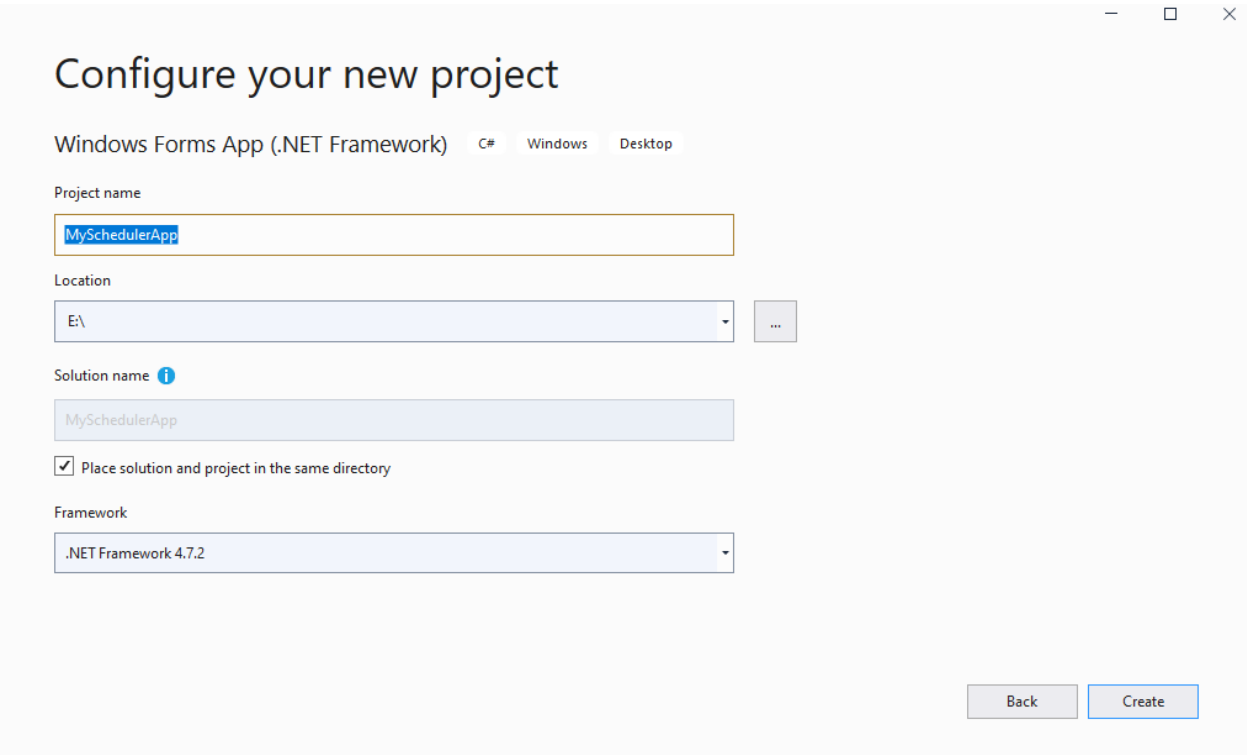
- After completing this self-guided hands-on, you will be able to implement the IJob interface to design a custom job class using the Quartz.Net library.

### Step 1 - : Create a new Windows Forms application

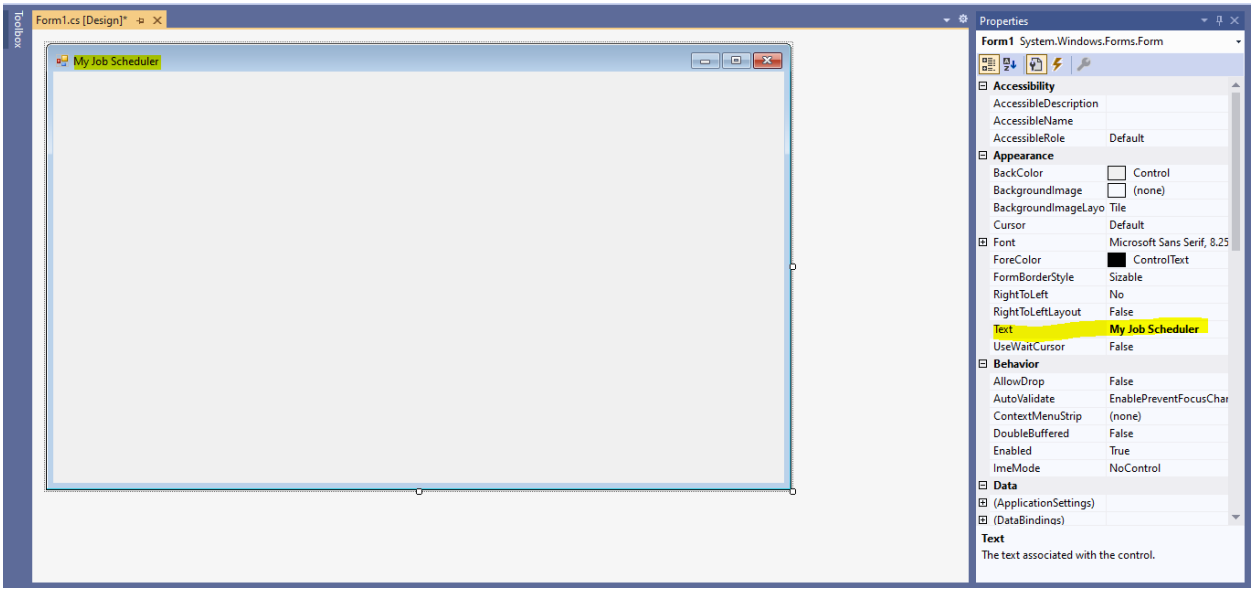
Launch the Visual Studio IDE (Version 2019 is used in this demo). Choose Windows Forms App (.NET Framework) from Create a new project dialog window. Click Next



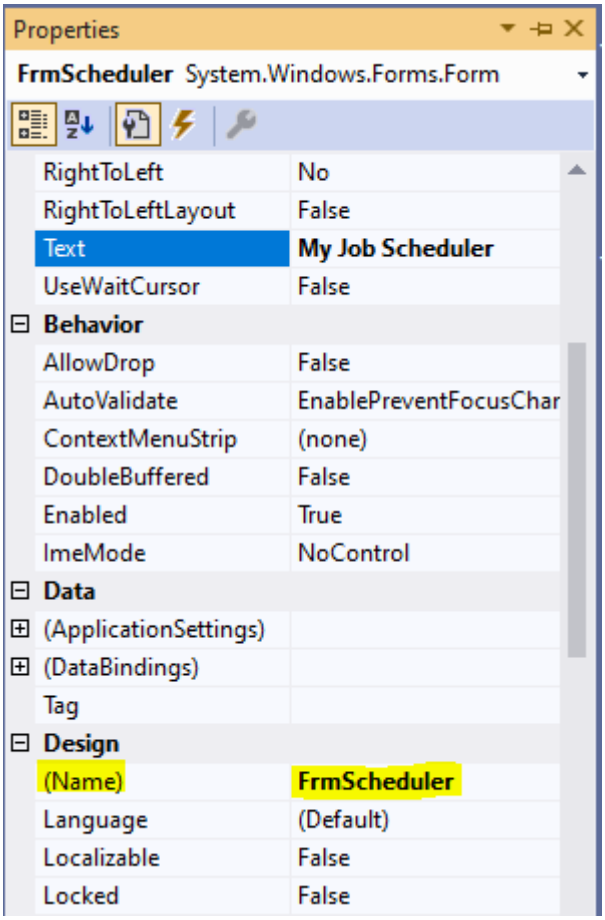

Give appropriate name as shown in the below screen. Click Create to finish creating the new project.



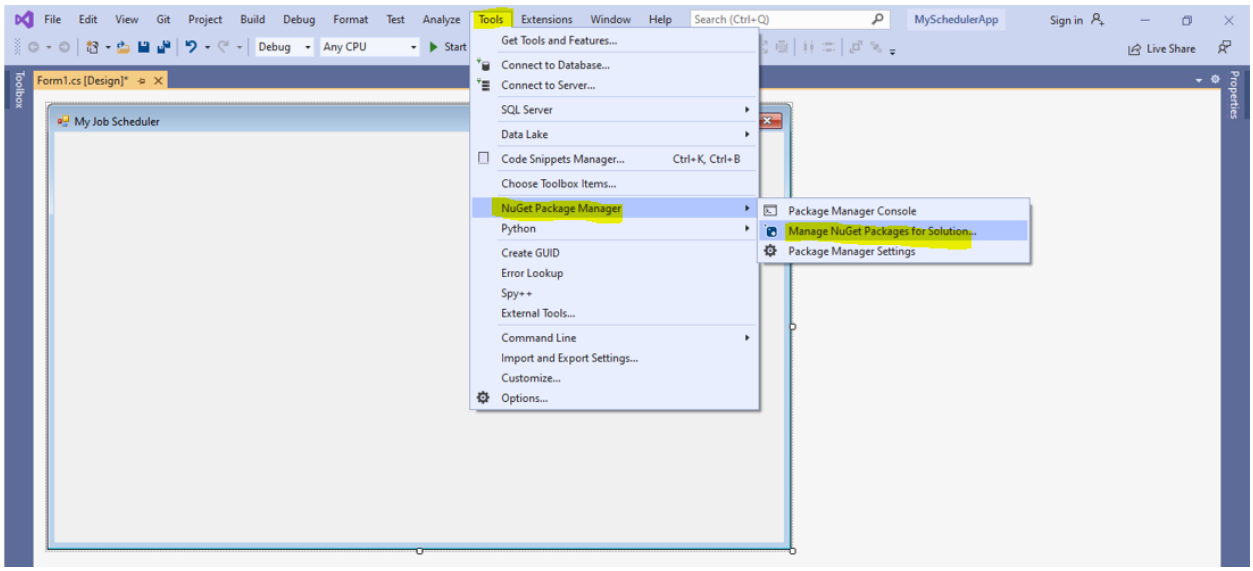

Change the form title as “My Job Scheduler”. You can set the **Text** property value using **Properties** window (Right-click on the form and choose Properties).




Also, rename the **Name** property value as “FrmScheduler”

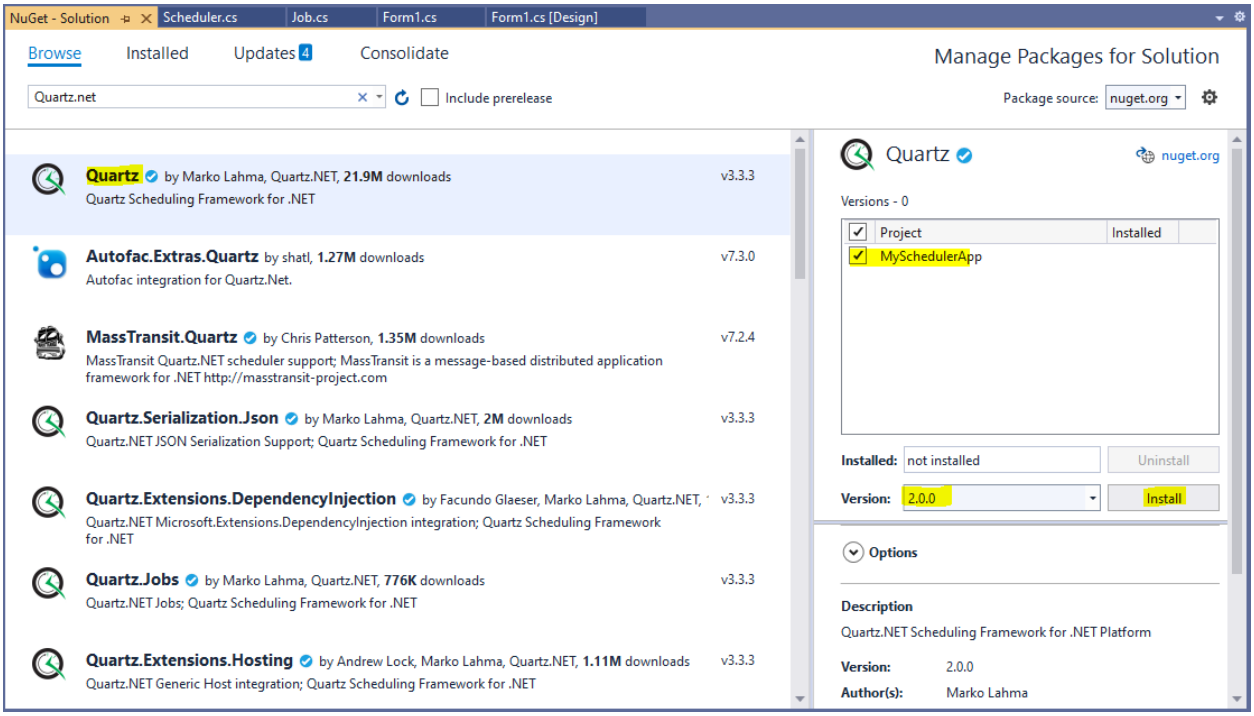



Now we will add Quartz.NET to our project. In order to do that, add the reference of it using NuGet Package Manager.

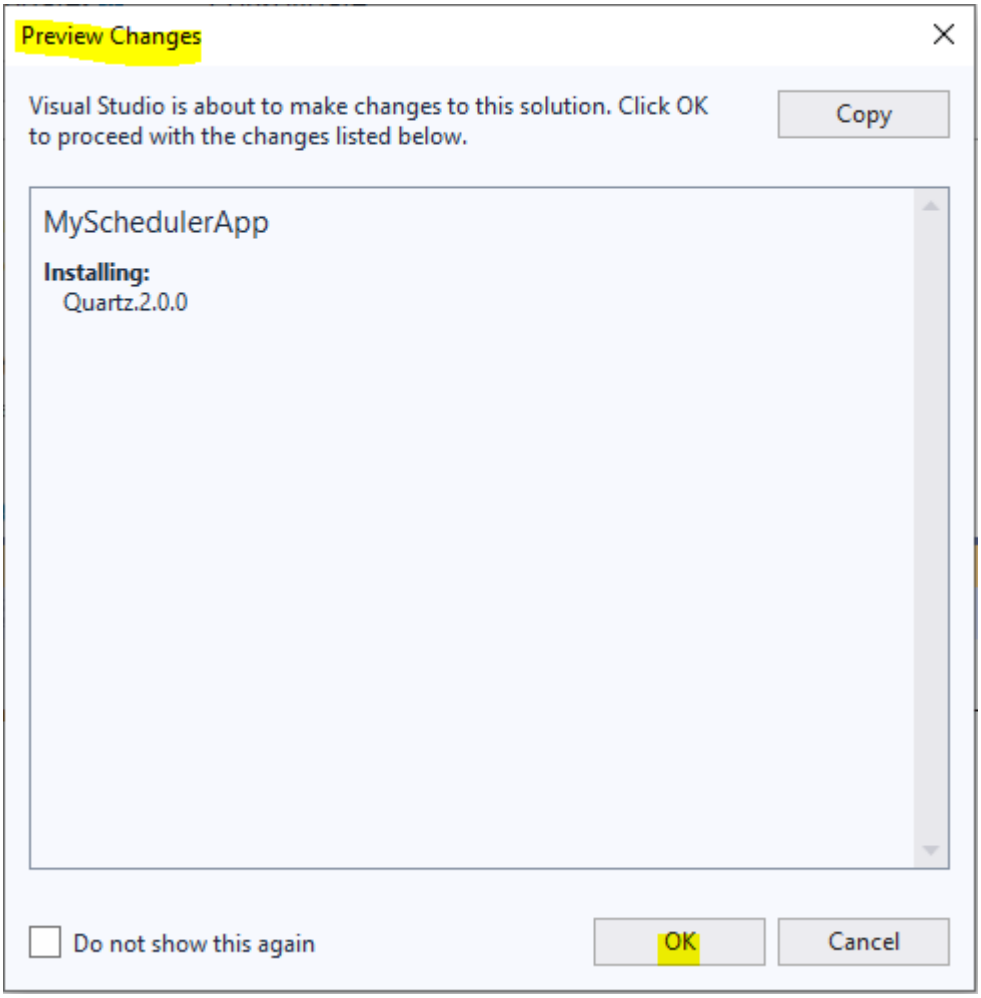


Browse for Quartz.NET and install it. See the below screenshot for implementing the same at your end.

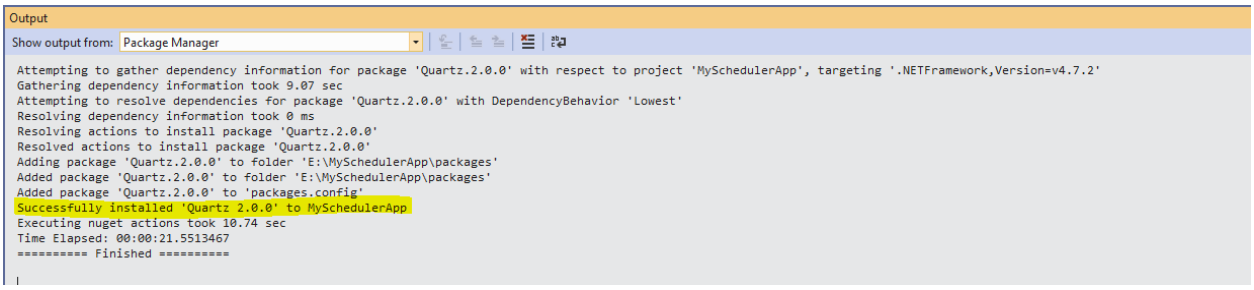
**Note:** Choose **version 2.0.0** from the Version dropdown list.




Click **Install** and press **OK** when the **Preview Changes** dialog window shows up. Further press **I Agree** button in the next window (If shows up).



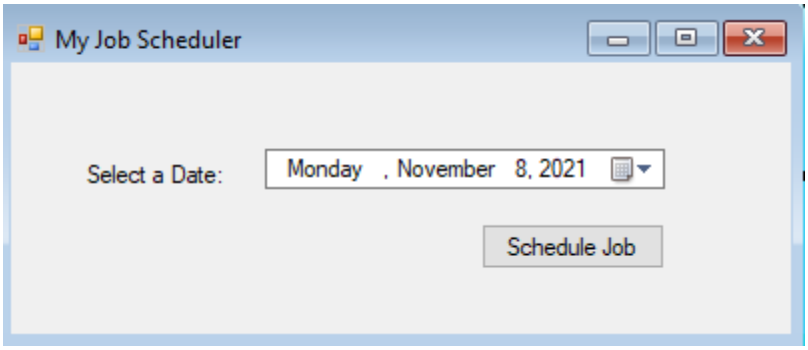

Make sure you get the success message in the Output window as shown below.



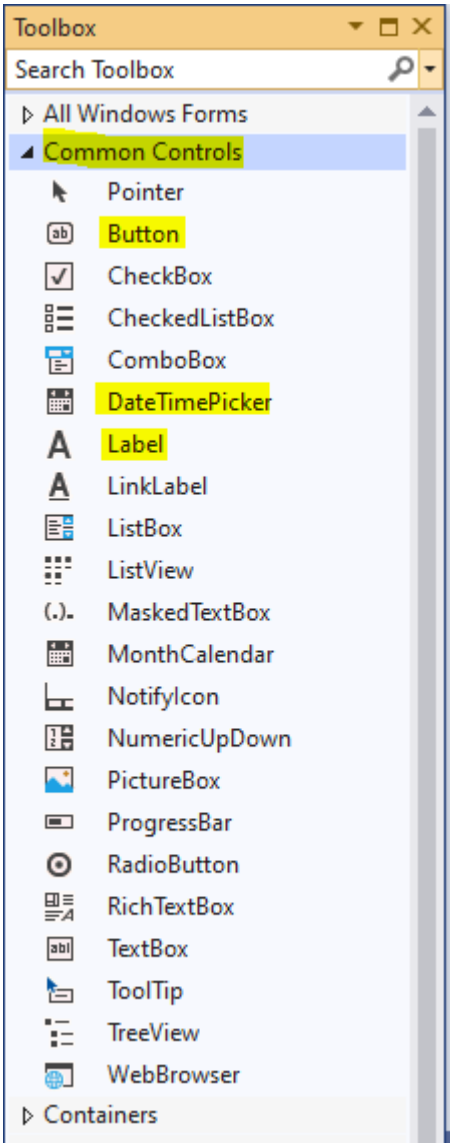

### Step 2 – Create a Scheduler Program

Here, we are building a Windows Forms application in which the first form contains a **DateTime picker** and a **Schedule** button. When a user clicks on the button, a job is scheduled at the time he/she sets in DateTime picker. And, this scheduled job starts doing its work at the given time and date.

Design your forms as shown below.

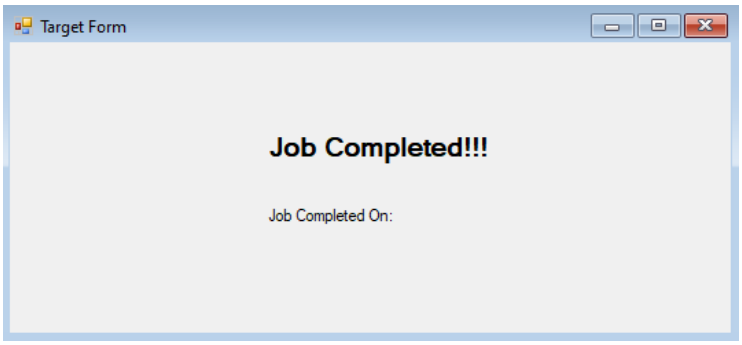



You can add controls using **Toolbox** window. These controls can be found under **Common Controls** section of the Toolbox window.




Change the Name property of the Button as “BtnScheduleJob”

Add second form to the project and design it as follows:




Drag and drop three label controls and change the text properties of first two labels as given below.

Label1 -> Job Completed!!! (Make the font size bigger as seen in the form)

Label2-> Job Completed On:

Also, change the name property of Label3 as “LblTime”

In the Form2.cs class write the following code.

```
using System;
using System.Windows.Forms;

namespace MySchedulerApp
{
    4 references
    public partial class Form2 : Form
    {
        1 reference
        public Form2()
        {
            InitializeComponent();

            LblTime.Text = DateTime.Now.ToString();
        }
    }
}
```

**Step 3 – Create Job and Scheduler classes**

Create the following two classes for achieving the scheduling task. First one is of Job and the second one is of the Scheduler.

**Job Class**

This is the job which we want to execute at given time. Here, just a new form is opened when the job starts.

```
using Quartz;

namespace MySchedulerApp
{
    1 reference
    public class Job : IJob
    {
        0 references
        public void Execute(IJobExecutionContext context)
        {
            Form2 form2 = new Form2();
            form2.ShowDialog();
        }
    }
}
```

## Scheduler Class

This class will schedule the job written in the Job class at the given time and the job will start executing with high priority. When the given time comes, the scheduler will trigger the given job and start working on it.

```
using System;

namespace MySchedulerApp
{
    0 references
    public class Scheduler
    {
        0 references
        public void Start(DateTime date)
        {

            IScheduler scheduler = (IScheduler)StdSchedulerFactory.GetDefaultScheduler();
            scheduler.Start();
            IJobDetail job = JobBuilder.Create<Job>().Build();
            ITrigger trigger = TriggerBuilder.Create()
             .WithIdentity("IDGJob", "IDG")
               .StartAt(date)
               .WithPriority(1)
               .Build();
            scheduler.ScheduleJob(job, trigger);

        }
    }
}
```

### Step 4 – Handle Click Event of the Schedule Button

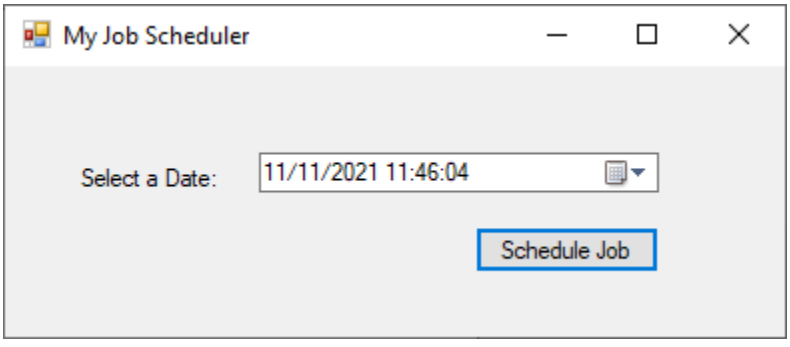
Go to **Form1** in design mode and double click on the **Schedule Job** button in order to handle the **Click** event. Write the following code in the Form1 class.

```
using System;
using System.Windows.Forms;

namespace MySchedulerApp
{
    3 references
    public partial class FrmScheduler : Form
    {
        1 reference
        public FrmScheduler()
        {
            InitializeComponent();

            dateTimePicker1.Format = DateTimePickerFormat.Custom;
            dateTimePicker1.CustomFormat = "MM/dd/yyyy hh:mm:ss";
        }

        1 reference
        private void BtnScheduleJob_Click(object sender, EventArgs e)
        {
            DateTime date = dateTimePicker1.Value;
            Scheduler sc = new Scheduler();
            sc.Start(date);
        }
    }
}
```

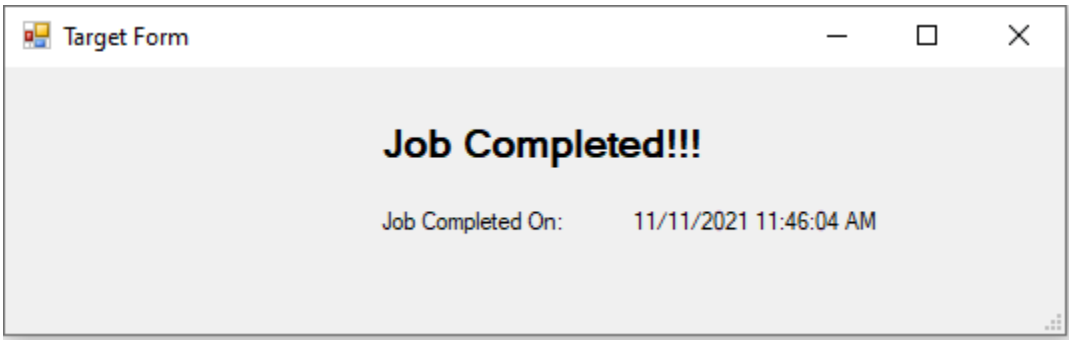
When the form is initialized, the date time picker format is changed and contains hours, minutes, and second along with the date so the user will start the scheduler at a more precise time. And, in click handler of a button, the **Scheduler** class is started with the given date.

### Step 5 – Executing the Application

Finally Run the application and schedule the job in order to complete the task. In this demo, I have set the time in which the job has to be completed as follows:




And you will see the second form will launch on that scheduled time.


Target Form
Job Completed!!!
Job Completed On:
11/11/2021 11:46:04 AM