

## Lect4+ Lect5

### Advance programming

### Level 3 IT

### IBB UN

بناء زر يتغير لونه بشكل عشوائي عند التاشير عليه في حالة تحويل الخاصية المضافة له  
Light الى True من الخصائص

```
using System;
using System.Windows.Forms;
using System.Drawing;
namespace lecturer4
{
    class BLight : Button
    {
        Random r = new Random();
        Timer timer = new Timer();
        private bool _Light;
        //خاصية جديدة تظهر بين الخصائص
        public bool Light
        {
            get{return _Light; }// get => _Light;
            set {_Light = value;}
        }
        public BLight()
        {
            this.Light = false;

            this.Width = 90;

            this.Height = 60;

            timer.Interval = 200;

            timer.Tick += new EventHandler(Timer_Tick);

            this.MouseMove += new MouseEventHandler
                ((object sender, MouseEventArgs e) =>
```

```

        {
            timer.Enabled = Light;
        });

        this.MouseLeave += new EventHandler
            ((object sender, EventArgs e) =>
            {
                timer.Enabled = false;
            });
    }

    private void Timer_Tick(object sender, EventArgs
e)
    {

        int rr = r.Next(0, 255);

        int g = r.Next(0, 255);

        int b = r.Next(0, 255);

        this.BackColor = Color.FromArgb(rr, g, b);

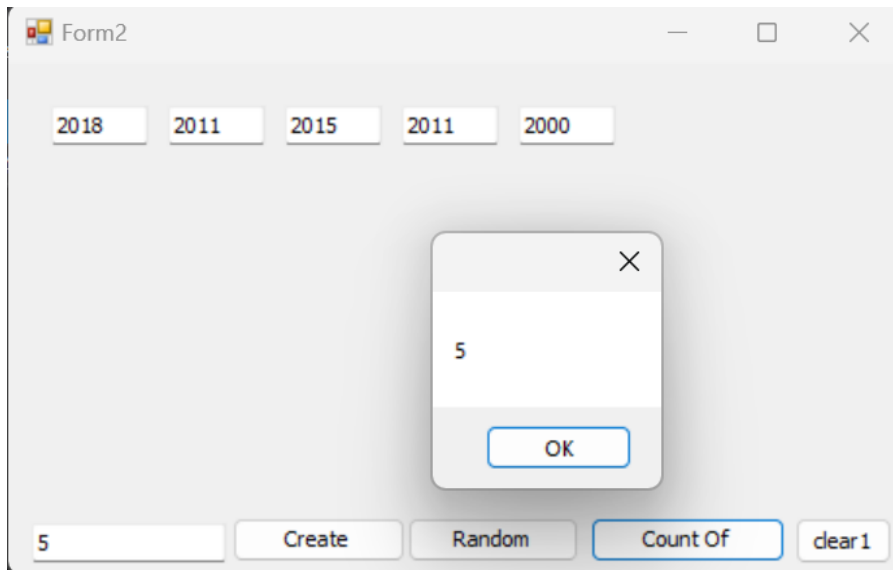
    }

}

}

```

إنشاء مصفوفة TextBox مع اعطائهم قيم عشوائية كذلك عددهم مع زر التنضيف



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace lecter4
{
    public partial class Form2 : Form
    {
        public Form2()
        {
            InitializeComponent();
        }
    }
}
```

```

        // TextBox t;
        //int i = 0;
        TextBox[] T = new TextBox[20];
        Random r = new Random();
        int x ;
        int y ;
        private void button1_Click(object sender,
EventArgs e)
        {
            x = 10;
            y = 10;
            int Count =int.Parse(textBox1.Text);
            panel1.Controls.Clear();
            for(int i=0;i< Count;i++)
            {

                T[i] = new TextBox()
                {
                    Location = new Point(x, y),
                    Size = new Size(50, 20)
                };

                x += 10 + T[i].Width;

                panel1.Controls.Add(T[i]);
            }
        }

        private void button2_Click(object sender,
EventArgs e)
        {
            foreach(TextBox t in
panel1.Controls.OfType<TextBox>())
            {
                t.Text = r.Next(2000, 2019).ToString();
            }
        }

```

```

        private void button3_Click(object sender,
EventArgs e)
        {
            MessageBox.Show((panel1.Controls.OfType<Text
Box>().Count()).ToString());
        }

        private void Form2_Load(object sender, EventArgs
e)
        {
            button1.Enabled = button2.Enabled =
button3.Enabled = false;
        }

        private void textBox1_TextChanged(object sender,
EventArgs e)
        {
            if (textBox1.Text.Trim() == "")
                button1.Enabled = button2.Enabled =
button3.Enabled = false;
            else
                button1.Enabled = button2.Enabled =
button3.Enabled = true;
        }
    }
}

```

بناء زر يعمل على تنظيف أي TextBox داخل Panel أو GroupBox

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

```

Eng: Eissa AL-Gumaei

```

using System.Threading.Tasks;
using System.Windows.Forms;

namespace lecturer4
{
    class clear : Button
    {
        public clear() { this.Click +=
clearButton_Click; }
        void clearButton_Click(object sender,
EventArgs e)
        {
            // Loop through all the controls in the
form
            foreach (Control control in
this.Parent.Controls)
            {
                // Check if the control is a Panel
or a GroupBox
                if (control is Panel || control is
GroupBox)
                {
                    // Loop through all the TextBox
controls in the control
                    foreach (Control childControl in
control.Controls)
                    {
                        if (childControl is TextBox)
                        {
                            // Clear the Text property
of the TextBox control
                            ((TextBox)childControl).Clear();
                            //((TextBox)childControl).Text="";
                        }
                    }
                }
            }
        }
    }
}

```

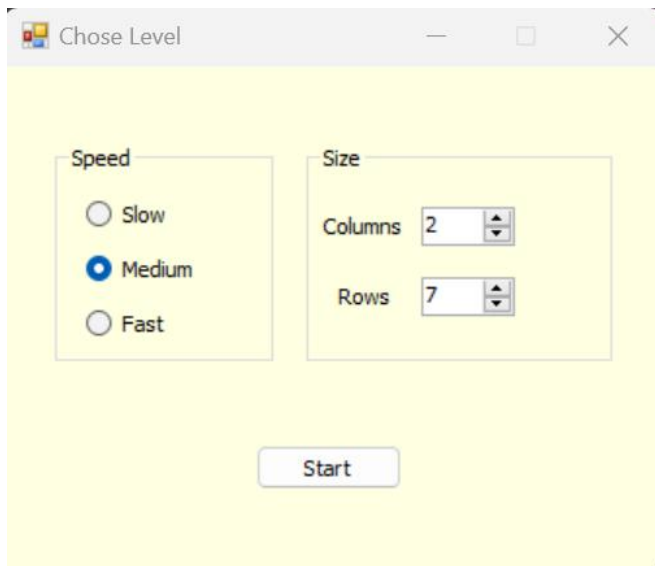
```

    }
  }
}

```

بناء لعبة المضرب مع تحديد السرعة والحجم

ملاحظة الأدوات داخل **GroupBox** سحب من بين الأدوات



```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

```

namespace لعبة\_التحطيم

Eng: Eissa AL-Gumaei

```

{
    public partial class ChoseLevel : Form
    {
        public ChoseLevel()
        {
            InitializeComponent();

            //===== اختيار المرحلة
            =====
            private void button1_Click(object sender,
            EventArgs e)
            {
                if (SlowRadio.Checked)
                    FormPlay.interval = 100;

                else if (MediumRadio.Checked)
                    FormPlay.interval = 30;

                else
                    FormPlay.interval = 1;

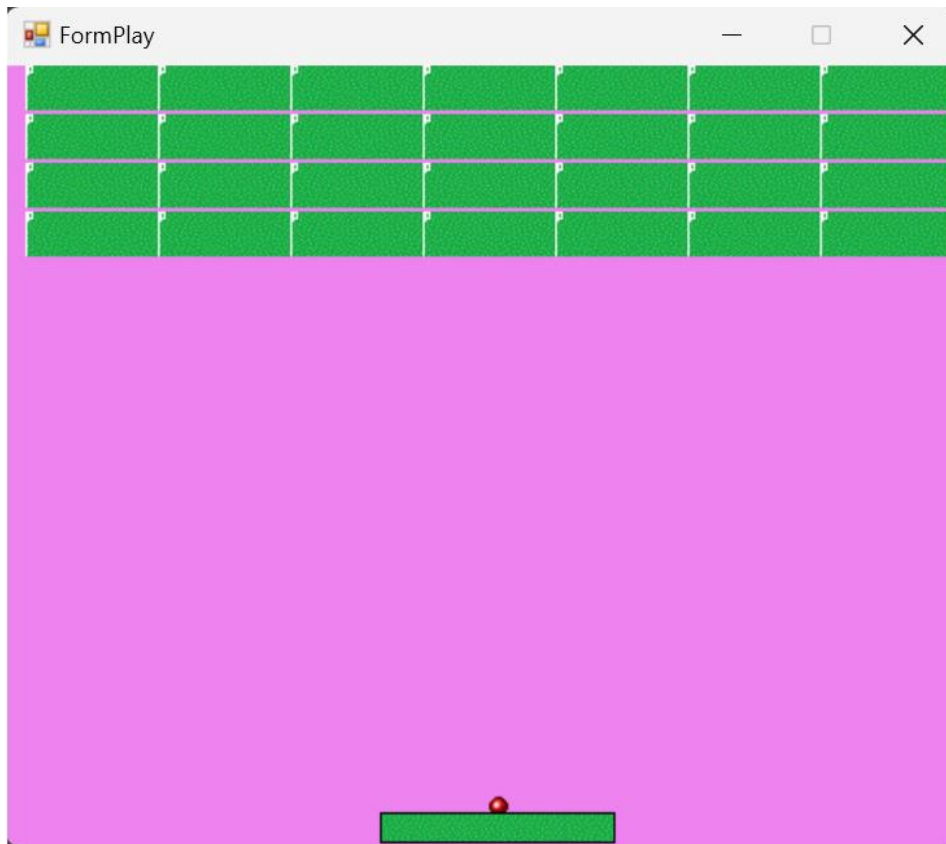
                FormPlay.n = int.Parse(numericUpDown1.Text);
                FormPlay.m = int.Parse(numericUpDown2.Text);
                this.Hide();
                FormPlay f1 = new FormPlay();
                f1.Show();
            }

            //=====
            =====

        }
    }
}

```





```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
```

```
namespace لعبة_التحطيم
{
```

Eng: Eissa AL-Gumaei

```

public partial class FormPlay : Form
{

//=====
=====
    static public int n; //عدد الصفوف
    static public int m; //عدد الأعمدة
    bool BallUp = true; //ارتفاع او انخفاض الكرة
    bool BallRight = false; //تحريك الكرة لليمين او اليسار
    PictureBox[,] toub; //مجموعة الطوب
    static public int interval; //السرعة
    int formWidth; //للاحتفاظ بحجم الفورم الابتدائي

//=====
=====
//===== عند
فتح الفورم =====
    public FormPlay()
    {
        InitializeComponent();

        Paddle.BackgroundImage =
Image.FromFile("paddle.gif");
        timer1.Interval = interval;
        this.BackColor = Color.White;
        Ball.SendToBack();
        Creattoub();
        Ball.Image = Image.FromFile("ball.gif");
        Paddle.Top = this.Height + 170;
        Ball.Top = Paddle.Top - 10;
        Paddle.Left = (this.Width / 2) -
(Paddle.Width / 2);
        Ball.Left = (this.Width / 2) - (Ball.Width /
2);
        this.Focus();
    }

//=====
=====

```

```

//===== انشاء مجموعة الطوب =====
void Creattoub()
{
    toub = new PictureBox[n, m];
    int top = 25, width = 68, height = 23;

    for (int i = 0; i < n; i++)
        for (int j = 0; j < m; j++)
        {
            toub[i, j] = new PictureBox();
            toub[i, j].Size = new Size(width,
height);
            toub[i, j].Location = new
Point(width * j+10, top * i);
            toub[i, j].Image =
Image.FromFile("brick.gif");
            toub[i, j].Enabled = false;
            this.Controls.Add(toub[i, j]);
        }
}

//=====
//===== عند الضغط من لوحة المفاتيح =====
private void FormPlay_KeyDown(object sender,
KeyEventArgs e)
{
    string result = e.KeyData.ToString();

    if (result == "Right")
        if (Paddle.Right <= formWidth - 20)
            Paddle.Left += 20;

    if (result == Keys.Left.ToString()) // if
(result == "Left")
        if (Paddle.Left >= 0)
            Paddle.Left -= 20;

```

```

        if (result == "Space")
            timer1.Start();

        else if (result == "Return")
            timer1.Stop();
    }

//=====
//=====
    bool f = true;
    //===== عند تشغيل التايمر
//=====
    private void timer1_Tick(object sender,
EventArgs e)
    {
        if (f)
        {
            formWidth = this.Width;
            f = false;
        }
        BallMove();
        DisposePaddle();
        if (Ball.Top >= Paddle.Bottom)
        {
            timer1.Stop();
            MessageBox.Show("Game Over \n try
again..");
            Ball.Left = Paddle.Left + Paddle.Width /
2;
            Ball.Top = Paddle.Top - Ball.Height;
            Creattoub();
        }
    }

//=====
//=====
    //===== تحريك الكرة
//=====

```

```

void BallMove()
{
    // ----- عند ضرب الكرة بالـPaddle -----
    -----
    if (Ball.Bottom >= Paddle.Top)
        if (Ball.Left >= Paddle.Left &&
Ball.Right <= Paddle.Right)
            BallUp = true;
    //-----
    -----

    //----- عند ضرب الكرة بأعلى الحائط -----
    -----
    if (Ball.Top <= 0)
        BallUp = false;
    //-----
    -----

    //----- عند ضرب الكرة بيمين الحائط -----
    -----
    if (Ball.Right >= this.Width - 20)
        BallRight = false;
    //-----
    -----

    //----- عند ضرب الكرة بيسار الحائط -----
    -----
    if (Ball.Left <= 0)
        BallRight = true;
    //-----
    -----

    //----- تحريك الكرة -----
    -----
    Ball.Top += (BallUp) ? -5 : 5; // ارتفاع الكرة
    عند الاصطدام بالمضرب أو انخفاضها عند الاصطدام بأعلى الحائط
    Ball.Left += (BallRight) ? 5 : -5; // تحريك
    الكرة لليمين عند الاصطدام بيسار الحائط والعكس
    //-----
    -----

```

```

    }

//=====
//=====
//===== تدمير طوب =====
//=====
// Object1.Bounds.Intersect(Object2.Bounds) او
استخدام الدالة الجاهزة
void DisposePaddle()
{
    foreach (PictureBox mytoub in
this.Controls.OfType<PictureBox>())
    {
        //
        if (Ball.Top <= mytoub.Bottom &&
Ball.Bottom >= mytoub.Bottom - 3)
            if (Ball.Right <= mytoub.Right &&
Ball.Left >= mytoub.Left)
            {
                mytoub.Dispose();
                BallUp = false;
            }
        else if (Ball.Bottom >= mytoub.Top
&& Ball.Top < mytoub.Top - 3)
            if (Ball.Right <= mytoub.Right
&& Ball.Left >= mytoub.Left)
            {
                mytoub.Dispose();
                BallUp = true;
            }

        //من جهة اليسار
        else if (Ball.Right >=
mytoub.Left && Ball.Left > mytoub.Left - 3)
            if (Ball.Bottom <=
mytoub.Bottom && Ball.Top >= mytoub.Top)
            {
                mytoub.Dispose();
                BallRight = false;
            }
    }
}

```

```

    }
    //من جهة اليمين
    else if (Ball.Left <=
mytoub.Right && Ball.Right > mytoub.Right - 3)
        if (Ball.Bottom <=
mytoub.Bottom && Ball.Top >= mytoub.Top)
        {
            mytoub.Dispose();
            BallRight = true;
        }

    }
}

//=====
=====
}
}

```