# TUGAS 1

## Soal

1. Buat fungsional baru pada form1 untuk melakukan zoom in pada formPict yang aktif. (Jika tekan Zoom in, maka picturebox1 pada formpict yang aktif akan berukuran height bertambah 50 dan width bertambah 50).

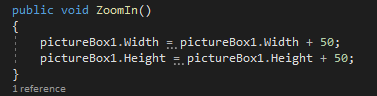
## Source Code

|  |  |
| --- | --- |
|  | Form1.cs |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127  128  129  130  131  132  133  134  135  136  137  138  139  140  141  142  143  144  145  146  147  148  149  150  151  152  153  154  155  156 | 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 Bab3  {  public partial class Form1 : Form, INotifyPropertyChanged  {  string ActiveForm;  List<FormPict> frmPict = new List<FormPict>();  List<ToolStripMenuItem> TsItem = new List<ToolStripMenuItem>();  ToolStripMenuItem Ts;  public event PropertyChangedEventHandler PropertyChanged;  private void OnPropertyChanged(string v)  {  PropertyChanged?.Invoke(this, new PropertyChangedEventArgs(v));  ToolStripMenuItem item = (ToolStripMenuItem)menuStrip1.Items["WindowToolStripMenuItem"];  foreach (ToolStripMenuItem itemchecked in item.DropDownItems)  {  if (!itemchecked.Text.Equals(v))  itemchecked.Checked = false;  else  itemchecked.Checked = true;  }  }  public String ActiveForm1  {  get  {  return ActiveForm;  }  set  {  ActiveForm = value;  OnPropertyChanged(ActiveForm1);  }  }  public Form1()  {  InitializeComponent();  FormPict FP;  }  private void OpenToolStripButton\_Click(object sender, EventArgs e)  {  if (openFileDialog1.ShowDialog() == DialogResult.OK)  {  frmPict.Add(new FormPict(this)  {  Text = System.IO.Path.GetFileName(openFileDialog1.FileName),  MdiParent = this  });  frmPict[frmPict.Count - 1].setpic(Image.FromFile(openFileDialog1.FileName));  frmPict[frmPict.Count - 1].Show();  ToolStripMenuItem item = (ToolStripMenuItem)menuStrip1.Items["windowToolStripMenuItem"];  foreach (ToolStripMenuItem itemchecked in item.DropDownItems)  {  itemchecked.Checked = false;  }  TsItem.Add(new ToolStripMenuItem(null, null, ItemClick)  {  Text = System.IO.Path.GetFileName(openFileDialog1.FileName),  Checked = true  });  item.DropDownItems.Add(TsItem[TsItem.Count - 1]);  }  }  private void ItemClick(object sender, EventArgs e)  {  var tms = (ToolStripMenuItem)sender;  for (int i = 0; i < frmPict.Count; i++)  {  if (String.Equals(frmPict[i].Text, tms.ToString(), StringComparison.OrdinalIgnoreCase))  {  frmPict[i].Activate();  }  }  }  private void ToolStripButton1\_Click(object sender, EventArgs e)  {  FormPict tempChild = (FormPict)this.ActiveMdiChild;  tempChild.PutarKanan();  }  private int AllDegree;  public int AllDegree1  {  get  {  return AllDegree;  }  set  {  AllDegree = value;  }  }  private void ToolStripButton2\_Click(object sender, EventArgs e)  {  FormPict tempChild = (FormPict)this.ActiveMdiChild;  tempChild.PutarKiri();  }  private void ToolStripButton3\_Click(object sender, EventArgs e)  {  FormPict tempChild = (FormPict)this.ActiveMdiChild;  tempChild.ZoomIn();  }  }  } |

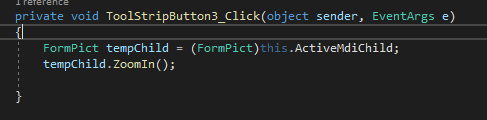
|  |  |
| --- | --- |
|  | FormPict.cs |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99 | 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 Bab3  {  public partial class FormPict : Form  {  int deg, n;  Image gbrAsli;  Form1 FrmParent;  public FormPict(Form1 F1)  {  InitializeComponent();  FrmParent = F1;  }  public void setpic(Image img)  {  pictureBox1.Image = img;  gbrAsli = pictureBox1.Image;  }  public void PutarKanan()  {  Image flipImage = pictureBox1.Image;  flipImage.RotateFlip(RotateFlipType.Rotate90FlipNone);  deg += 90;  if (deg > 360)  {  deg = deg - 360;  }  pictureBox1.Image = flipImage;  textBox1.Text = deg.ToString();  }  private void button1\_Click(object sender, EventArgs e)  {  FrmParent.AllDegree1 = deg;  }  private void button2\_Click(object sender, EventArgs e)  {  int a1, a2;  a1 = FrmParent.AllDegree1 / 90;  a2 = deg / 90;  if (a2 < a1)  {  for(int i = a2; i<a1; i++)  {  PutarKanan();  }  }else if (a2 > a1)  {  for(int i=a1; i < a2; i++)  {  PutarKiri();  }  }  }  public void PutarKiri()  {  Image flipImage = pictureBox1.Image;  flipImage.RotateFlip(RotateFlipType.Rotate270FlipNone);  pictureBox1.Image = flipImage;  deg -= 90;  if (deg < 0)  {  textBox1.Text = deg.ToString();  }  }  public void ZoomIn()  {  pictureBox1.Width = pictureBox1.Width + 50;  pictureBox1.Height = pictureBox1.Height + 50;  }  public void ZoomOut()  {  pictureBox1.Width = pictureBox1.Width - 50;  pictureBox1.Height = pictureBox1.Height - 50;  }  }  } |

## Screenshot

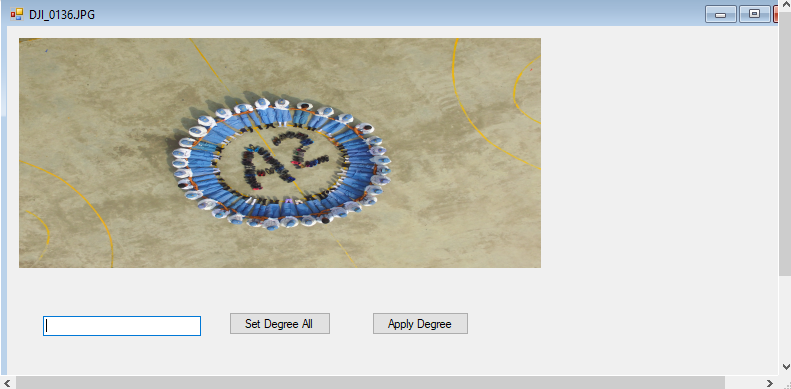
* + - 1. Pembuatan Method Zoom in

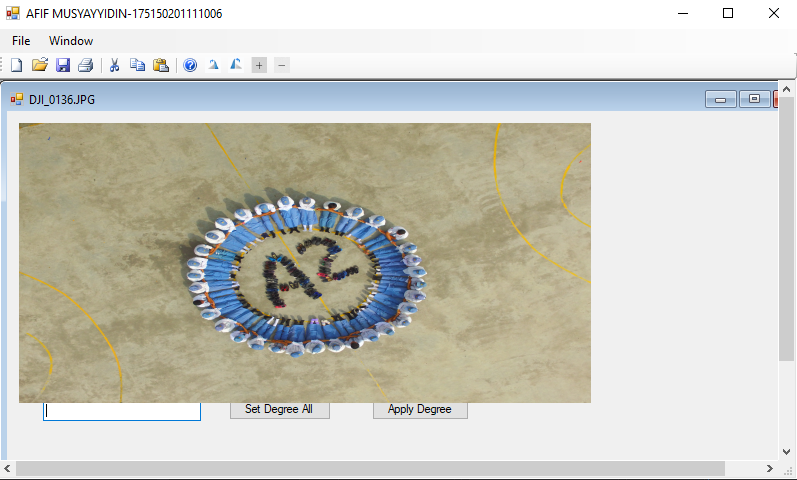


* + - 1. Pemanggilan Method ZoomIn pada Form1



* + - 1. Kondisi Awal



* + - 1. Saat Di Zoom 

## Penjelasan

Dalam soal berikut, diperintahkan untuk melakukan pembuatan button zoom in agar gambar dapat diperbesar ukurannya pada FormPict dengan menggunakan button tersebut. Awalnya seperti pembuatan event klik kanan dan klik kiri pada latihan yaitu membuat method Zoom In pada FormPict yang berisi dimana ukuran lebar dan tinggi pada gambar ditambah dengan 50. Kemudian setelah itu dilakukan pembuatan method Zoom In pada FormPict, dimasukkan method tersebut pada Form1 yang merupakan parent dari FormPict yaitu dengan memberikan event pada tombol Zoom In yang sudah disediakan sebelumnya (Melakukan double klik pada button zoom in). Kemudian isi dari evebt button tersebut adalah pembuatan objek tempChild dari kelas FormPict. Kemudian memanggil method yang ada di FormPict (Method Zoom In) dengan menggunakan method tempChild tersebut. Form1 dapat dijalankan dan memasukkan gambar yang ingin ditampilkan. Lakukan klik pada button zoom in.

# TUGAS 2

## Soal

1. Buat fungsional baru pada form1 untuk melakukan zoom out pada formPict yang aktif. (Jika tekan Zoom in, maka picturebox1 pada formpict yang aktif akan berukuran height bertambah 50 dan width bertambah 50).

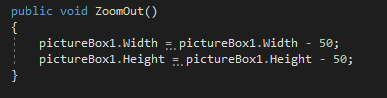
## Source Code

|  |  |
| --- | --- |
|  | Form1.cs |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127  128  129  130  131  132  133  134  135  136  137  138  139  140  141  142  143  144  145  146  147  148  149  150  151  152  153  154  155  156  157  158  159  160  161  162  163  164 | 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 Bab3  {  public partial class Form1 : Form, INotifyPropertyChanged  {  string ActiveForm;  List<FormPict> frmPict = new List<FormPict>();  List<ToolStripMenuItem> TsItem = new List<ToolStripMenuItem>();  ToolStripMenuItem Ts;  public event PropertyChangedEventHandler PropertyChanged;  private void OnPropertyChanged(string v)  {  PropertyChanged?.Invoke(this, new PropertyChangedEventArgs(v));  ToolStripMenuItem item = (ToolStripMenuItem)menuStrip1.Items["WindowToolStripMenuItem"];  foreach (ToolStripMenuItem itemchecked in item.DropDownItems)  {  if (!itemchecked.Text.Equals(v))  itemchecked.Checked = false;  else  itemchecked.Checked = true;  }  }  public String ActiveForm1  {  get  {  return ActiveForm;  }  set  {  ActiveForm = value;  OnPropertyChanged(ActiveForm1);  }  }  public Form1()  {  InitializeComponent();  FormPict FP;  }  private void OpenToolStripButton\_Click(object sender, EventArgs e)  {  if (openFileDialog1.ShowDialog() == DialogResult.OK)  {  frmPict.Add(new FormPict(this)  {  Text = System.IO.Path.GetFileName(openFileDialog1.FileName),  MdiParent = this  });  frmPict[frmPict.Count - 1].setpic(Image.FromFile(openFileDialog1.FileName));  frmPict[frmPict.Count - 1].Show();  ToolStripMenuItem item = (ToolStripMenuItem)menuStrip1.Items["windowToolStripMenuItem"];  foreach (ToolStripMenuItem itemchecked in item.DropDownItems)  {  itemchecked.Checked = false;  }  TsItem.Add(new ToolStripMenuItem(null, null, ItemClick)  {  Text = System.IO.Path.GetFileName(openFileDialog1.FileName),  Checked = true  });  item.DropDownItems.Add(TsItem[TsItem.Count - 1]);  }  }  private void ItemClick(object sender, EventArgs e)  {  var tms = (ToolStripMenuItem)sender;  for (int i = 0; i < frmPict.Count; i++)  {  if (String.Equals(frmPict[i].Text, tms.ToString(), StringComparison.OrdinalIgnoreCase))  {  frmPict[i].Activate();  }  }  }  private void ToolStripButton1\_Click(object sender, EventArgs e)  {  FormPict tempChild = (FormPict)this.ActiveMdiChild;  tempChild.PutarKanan();  }  private int AllDegree;  public int AllDegree1  {  get  {  return AllDegree;  }  set  {  AllDegree = value;  }  }  private void ToolStripButton2\_Click(object sender, EventArgs e)  {  FormPict tempChild = (FormPict)this.ActiveMdiChild;  tempChild.PutarKiri();  }  private void ToolStripButton3\_Click(object sender, EventArgs e)  {  FormPict tempChild = (FormPict)this.ActiveMdiChild;  tempChild.ZoomIn();  }  private void ToolStripButton4\_Click(object sender, EventArgs e)  {  FormPict tempChild = (FormPict)this.ActiveMdiChild;  tempChild.ZoomOut();  }  }  } |

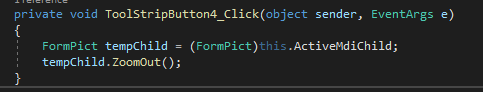
|  |  |
| --- | --- |
|  | FormPict.cs |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99 | 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 Bab3  {  public partial class FormPict : Form  {  int deg, n;  Image gbrAsli;  Form1 FrmParent;  public FormPict(Form1 F1)  {  InitializeComponent();  FrmParent = F1;  }  public void setpic(Image img)  {  pictureBox1.Image = img;  gbrAsli = pictureBox1.Image;  }  public void PutarKanan()  {  Image flipImage = pictureBox1.Image;  flipImage.RotateFlip(RotateFlipType.Rotate90FlipNone);  deg += 90;  if (deg > 360)  {  deg = deg - 360;  }  pictureBox1.Image = flipImage;  textBox1.Text = deg.ToString();  }  private void button1\_Click(object sender, EventArgs e)  {  FrmParent.AllDegree1 = deg;  }  private void button2\_Click(object sender, EventArgs e)  {  int a1, a2;  a1 = FrmParent.AllDegree1 / 90;  a2 = deg / 90;  if (a2 < a1)  {  for(int i = a2; i<a1; i++)  {  PutarKanan();  }  }else if (a2 > a1)  {  for(int i=a1; i < a2; i++)  {  PutarKiri();  }  }  }  public void PutarKiri()  {  Image flipImage = pictureBox1.Image;  flipImage.RotateFlip(RotateFlipType.Rotate270FlipNone);  pictureBox1.Image = flipImage;  deg -= 90;  if (deg < 0)  {  textBox1.Text = deg.ToString();  }  }  public void ZoomIn()  {  pictureBox1.Width = pictureBox1.Width + 50;  pictureBox1.Height = pictureBox1.Height + 50;  }  public void ZoomOut()  {  pictureBox1.Width = pictureBox1.Width - 50;  pictureBox1.Height = pictureBox1.Height - 50;  }  }  } |

## Screenshot

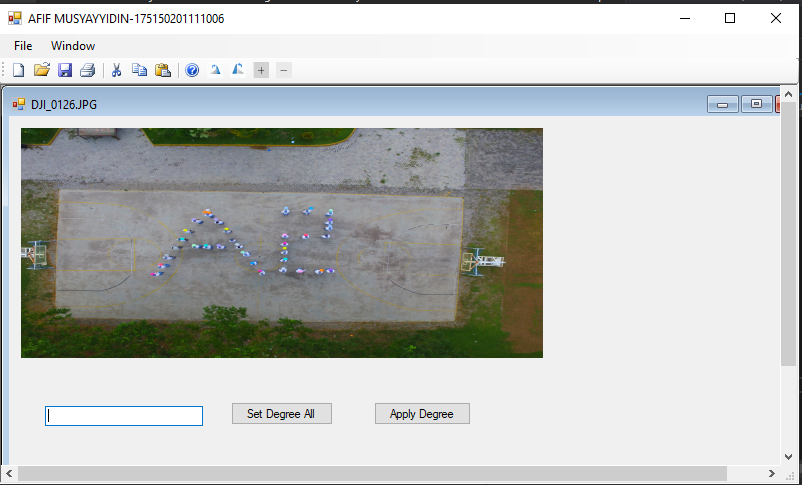
* + - 1. Pembuatan Method Zoom Out



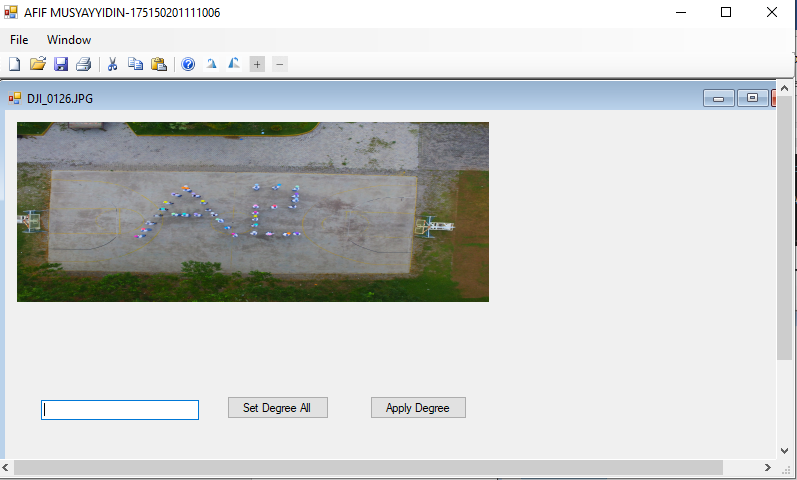
* + - 1. Pemanggilan Method Zoom Out



* + - 1. Kondisi Awal



* + - 1. Setelah di Zoom Out



## Penjelasan

Dalam soal berikut, diperintahkan untuk melakukan pembuatan button zoom Out agar gambar dapat diperkecil ukurannya pada FormPict dengan menggunakan button tersebut. Tidak jauh berbeda dengan pembuatan event pada button zoom In di soal pertama. Awalnya membuat method Zoom Out pada FormPict yang berisi dimana ukuran lebar dan tinggi pada gambar dikurangi dengan 50. Kemudian setelah dilakukan pembuatan method Zoom Out pada FormPict, dimasukkan method tersebut pada Form1 yang merupakan parent dari FormPict yaitu dengan memberikan event pada tombol Zoom Out yang sudah disediakan sebelumnya (Melakukan double klik pada button zoom out). Kemudian isi dari event button tersebut adalah pembuatan objek tempChild dari kelas FormPict. Kemudian memanggil method yang ada di FormPict (Method Zoom Out) dengan menggunakan method tempChild tersebut. Form1 dapat dijalankan dan memasukkan gambar yang ingin ditampilkan. Lakukan klik pada button zoom out

# TUGAS 3

## Soal

1. Buat fungsional baru pada formPict untuk mengatur ukuran yang akan diaplikasikan pada formpict lain yang aktif. (Jika tekan tombol Atur Zoom Level pada satu formpict, maka semua picturebox1 pada formpict lain2 akan berukuran sama dengan ukuran picturebox1 yang melakukan setting).

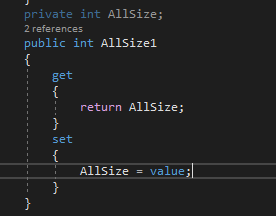
## Source Code

|  |  |
| --- | --- |
|  | Form1.cs |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127  128  129  130  131  132  133  134  135  136  137  138  139  140  141  142  143  144  145  146  147  148  149  150  151  152  153  154  155  156  157  158  159  160  161  162  163  164  165  166  167  168  169  170  171  172  173  174  175  176  177 | 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 Bab3  {  public partial class Form1 : Form, INotifyPropertyChanged  {  string ActiveForm;  List<FormPict> frmPict = new List<FormPict>();  List<ToolStripMenuItem> TsItem = new List<ToolStripMenuItem>();  ToolStripMenuItem Ts;  public event PropertyChangedEventHandler PropertyChanged;  private void OnPropertyChanged(string v)  {  PropertyChanged?.Invoke(this, new PropertyChangedEventArgs(v));  ToolStripMenuItem item = (ToolStripMenuItem)menuStrip1.Items["WindowToolStripMenuItem"];  foreach (ToolStripMenuItem itemchecked in item.DropDownItems)  {  if (!itemchecked.Text.Equals(v))  itemchecked.Checked = false;  else  itemchecked.Checked = true;  }  }  public String ActiveForm1  {  get  {  return ActiveForm;  }  set  {  ActiveForm = value;  OnPropertyChanged(ActiveForm1);  }  }  public Form1()  {  InitializeComponent();  FormPict FP;  }  private void OpenToolStripButton\_Click(object sender, EventArgs e)  {  if (openFileDialog1.ShowDialog() == DialogResult.OK)  {  frmPict.Add(new FormPict(this)  {  Text = System.IO.Path.GetFileName(openFileDialog1.FileName),  MdiParent = this  });  frmPict[frmPict.Count - 1].setpic(Image.FromFile(openFileDialog1.FileName));  frmPict[frmPict.Count - 1].Show();  ToolStripMenuItem item = (ToolStripMenuItem)menuStrip1.Items["windowToolStripMenuItem"];  foreach (ToolStripMenuItem itemchecked in item.DropDownItems)  {  itemchecked.Checked = false;  }  TsItem.Add(new ToolStripMenuItem(null, null, ItemClick)  {  Text = System.IO.Path.GetFileName(openFileDialog1.FileName),  Checked = true  });  item.DropDownItems.Add(TsItem[TsItem.Count - 1]);  }  }  private void ItemClick(object sender, EventArgs e)  {  var tms = (ToolStripMenuItem)sender;  for (int i = 0; i < frmPict.Count; i++)  {  if (String.Equals(frmPict[i].Text, tms.ToString(), StringComparison.OrdinalIgnoreCase))  {  frmPict[i].Activate();  }  }  }  private void ToolStripButton1\_Click(object sender, EventArgs e)  {  FormPict tempChild = (FormPict)this.ActiveMdiChild;  tempChild.PutarKanan();  }  private int AllDegree;  public int AllDegree1  {  get  {  return AllDegree;  }  set  {  AllDegree = value;  }  }  private int AllSize;  public int AllSize1  {  get  {  return AllSize;  }  set  {  AllSize = value;  }  }  private void ToolStripButton2\_Click(object sender, EventArgs e)  {  FormPict tempChild = (FormPict)this.ActiveMdiChild;  tempChild.PutarKiri();  }  private void ToolStripButton3\_Click(object sender, EventArgs e)  {  FormPict tempChild = (FormPict)this.ActiveMdiChild;  tempChild.ZoomIn();  }  private void ToolStripButton4\_Click(object sender, EventArgs e)  {  FormPict tempChild = (FormPict)this.ActiveMdiChild;  tempChild.ZoomOut();  }  }  } |

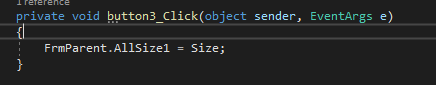
|  |  |
| --- | --- |
|  | FormPict.cs |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127  128  129  130  131  132 | 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 Bab3  {  public partial class FormPict : Form  {  int deg, n,Size;  Image gbrAsli;  Form1 FrmParent;  public FormPict(Form1 F1)  {  InitializeComponent();  FrmParent = F1;  }  public void setpic(Image img)  {  pictureBox1.Image = img;  gbrAsli = pictureBox1.Image;  }  public void PutarKanan()  {  Image flipImage = pictureBox1.Image;  flipImage.RotateFlip(RotateFlipType.Rotate90FlipNone);  deg += 90;  if (deg > 360)  {  deg = deg - 360;  }  pictureBox1.Image = flipImage;  textBox1.Text = deg.ToString();  }  private void button1\_Click(object sender, EventArgs e)  {  FrmParent.AllDegree1 = deg;  }  private void button2\_Click(object sender, EventArgs e)  {  int a1, a2;  a1 = FrmParent.AllDegree1 / 90;  a2 = deg / 90;  if (a2 < a1)  {  for(int i = a2; i<a1; i++)  {  PutarKanan();  }  }else if (a2 > a1)  {  for(int i=a1; i < a2; i++)  {  PutarKiri();  }  }  }  public void PutarKiri()  {  Image flipImage = pictureBox1.Image;  flipImage.RotateFlip(RotateFlipType.Rotate270FlipNone);  pictureBox1.Image = flipImage;  deg -= 90;  if (deg < 0)  {  textBox1.Text = deg.ToString();  }  }  public void ZoomIn()  {  pictureBox1.Width = pictureBox1.Width + 50;  pictureBox1.Height = pictureBox1.Height + 50;  Size += 50;  textBox1.Text = Size.ToString();  }  private void button4\_Click(object sender, EventArgs e)  {  int a1, a2;  a1 = FrmParent.AllSize1 / 50;  a2 = Size / 50;  if (a2 < a1)  {  for (int i = a2; i < a1; i++)  {  ZoomIn();  }  }  else if (a2 > a1)  {  for (int i = a1; i < a2; i++)  {  ZoomOut();  }  }  }  private void button3\_Click(object sender, EventArgs e)  {  FrmParent.AllSize1 = Size;  }  public void ZoomOut()  {  pictureBox1.Width = pictureBox1.Width - 50;  pictureBox1.Height = pictureBox1.Height - 50;  Size -= 50;  textBox1.Text = Size.ToString();  }  }  } |

## Screenshot

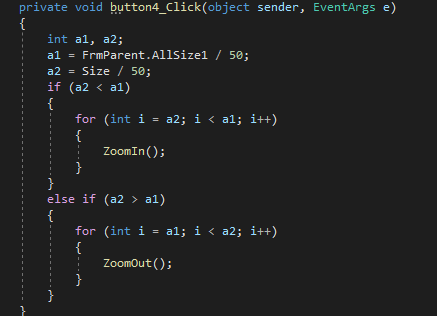
* + - 1. Membuat method untuk mengambill semua ukuran



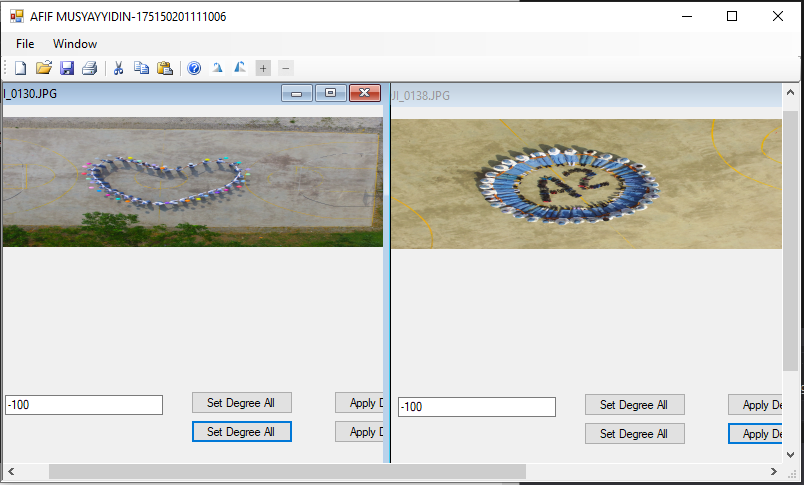
* + - 1. Membuat method untuk button



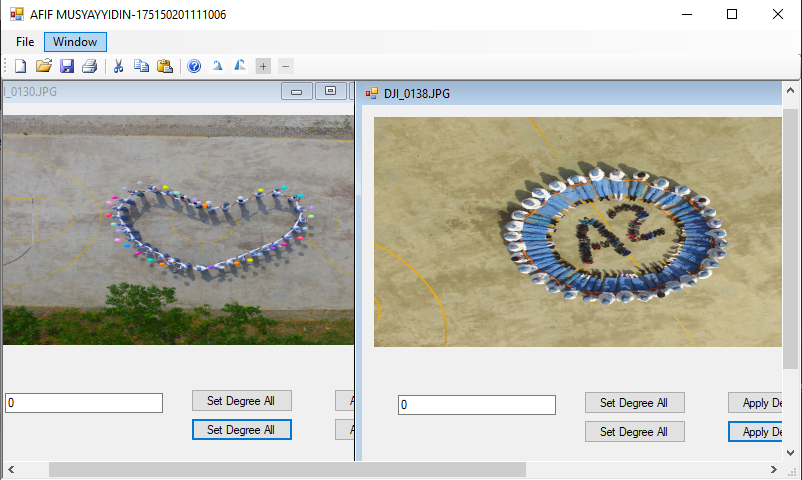
* + - 1. Membuat method untuk button Apply



* + - 1. Kondisi Awal



* + - 1. Kondisi Akhir



## Penjelasan

Pada percobaan diatas hamper sama seperti percobaan pada latihan, disini dibuat terlebih dahulu 2 button untuk Set Sizenya dan button untuk Apply Sizenya. Kemudian membuat method AllSize1 pada Form1.cs dan sebelumnya deklarasikan variable AllSize dan membuat variable global Size yang bertipe data integer. Setelah itu melakukan pembuatan event pada button baru yang telah dibuat tadi pada FormPict.cs. Sedikit perubahan pada Method ZoomIn dan ZoomOut pada FormPict.cs agar perubahanan ukurannya dapat diketahui dan dilihat pada textbox. Kemudian jalankan Form1.cs tersebut. Setelah itu buka gambar dengan mengklik button open file. Buka gambar sebanyak 2 untuk mengetest program tersebut. Lakukan perubahan pada gambar pertama yaitu dengan melakukan zoom in atau zoom out pada salah satu gambar tersebut lalu klik Set All Size. Setelah itu pindah pada gambar lainnya atau gambar kedua, klik apply size agar ukuran gambarnya dapat mengikuti ukuran gambar yang ada di gambar pertama.