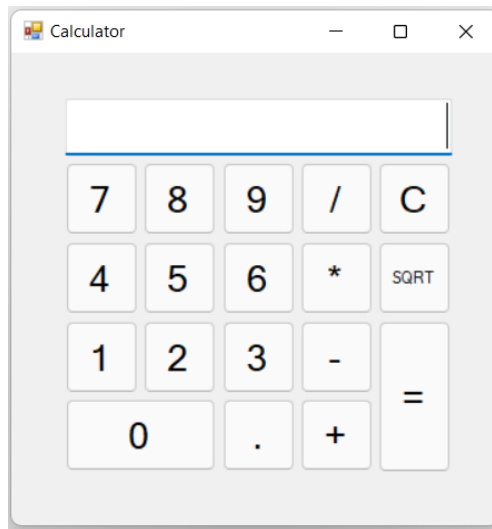


## BÀI THỰC HÀNH CÔNG NGHỆ .NET #1

- Làm các bài tập 1, 2, 3, 4 trong slide “Các điều khiển cơ bản của C#”.

## BÀI THỰC HÀNH CÔNG NGHỆ .NET #2

- Xây dựng phần mềm máy tính bỏ túi (Calculator)
- Thực hiện:
  1. Tạo form mới đặt tên Calculator
  2. Thiết kế giao diện như sau



Yêu cầu:

- Đặt tên (thuộc tính name trong properties) cho các điều khiển (control) như sau:
  - Các nút (button) được đặt bắt đầu bằng btn, ví dụ
    - Nút số 0 đặt tên (name) là btn0
    - Nút số 1 đặt tên là btn1
    - Tương tự cho các nút khác
    - Nút dấu chấm đặt tên btnDot
    - Nút dấu + đặt tên btnAdd
    - Nút dấu – đặt tên btnSubtract
    - Nút dấu \* đặt tên btnMultiple
    - Nút dấu / đặt tên btnDivide
    - Nút C đặt tên btnClear
    - Nút SQLT đặt btnSQRT
    - Nút dấu = đặt tên btnEqual

- Textbox được đặt tên txtResult, căn phải (phải chuột vào textbox, chọn properties, đặt thuộc tính RightToLeft=Yes)
3. Lập trình thực hiện các chức năng
- Nhấp phải chuột vào vị trí trống trên form, chọn View Code
  - Thêm một số biến toàn cục như sau

```
string sign;    //lưu dấu +, -, *, /
double val1;    //giá trị 1
double val2;    //giá trị 2

public Calculator()
{
    InitializeComponent();
}
```

- Quay lại trang giao diện của Calculator, nhấp đúp vào nút 1 (btn1), xuất hiện

```
private void btn1_Click(object sender, EventArgs e)
{
    ...
}
```

- Thêm nội dung sau

```
private void btn1_Click(object sender, EventArgs e)
{
    txtResult.Text += btn1.Text;
}
```

**Ghi chú: btn1.Text có thể thay bằng “1”;**

- Làm tương tự các nút khác
- Vào giao diện design, nhấp đúp chuột vào nút “C”

```
private void btnClear_Click(object sender, EventArgs e)
{
    txtResult.Text = "";
    val1 = 0;
    val2 = 0;
    sign = "";
}
```

- Nhấp đúp vào nút cộng (+)

```
private void btnAdd_Click(object sender, EventArgs e)
{
    sign = "+";
    val1 = double.Parse(txtResult.Text);
    txtResult.Text = "";
}
```

---

- Tương tự cho -, \*, /
- Nhấp đúp vào nút bằng (=)

```
private void btnEqual_Click(object sender, EventArgs e)
{
    val2 = double.Parse(txtResult.Text);
    double result;
    if (sign == "+")
    {
        result = val1 + val2;
        txtResult.Text = result.ToString();
    }
}
```

- Tương tự cho các nút khác

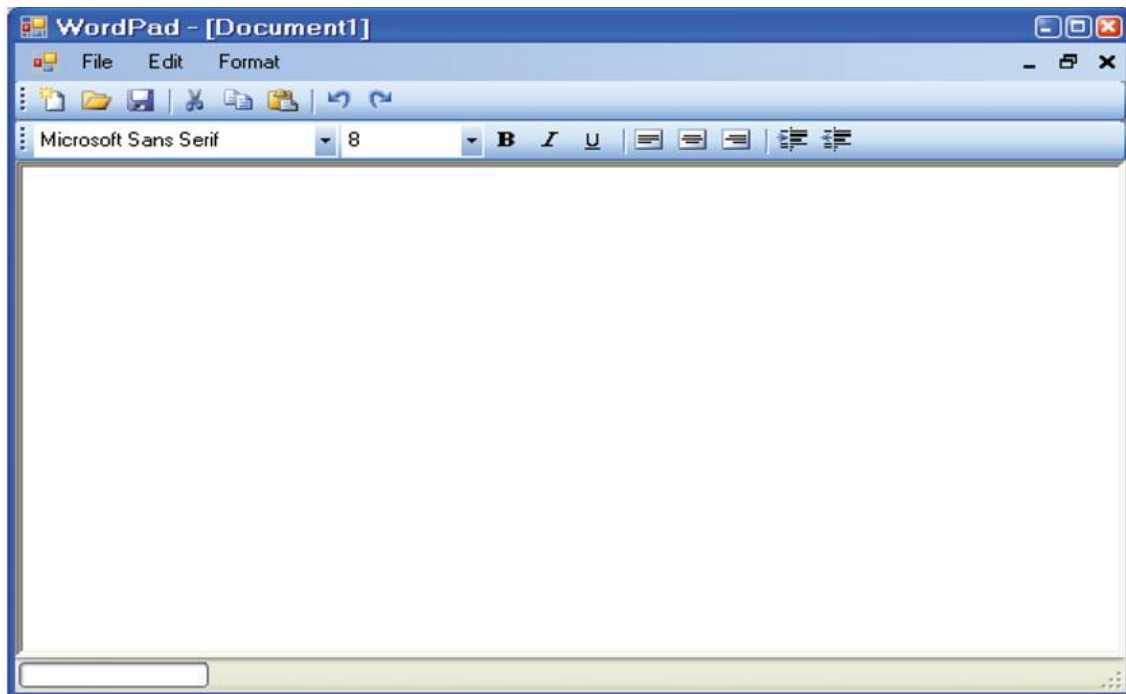
## BÀI THỰC HÀNH CÔNG NGHỆ .NET #3

### Mục đích:

- Thực hành các điều khiển cơ bản và nâng cao

### Yêu cầu:

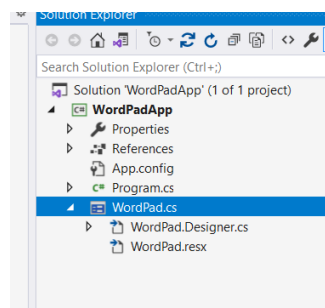
- Xây dựng phần mềm WordPad như sau



### Hướng dẫn (**Tạo một giao diện gần giống trên**):

#### 1. Tạo Form

- Mở Visual Studio 2019, tạo project mới (Windows Forms App (.NET Framework))
- Đặt tên cho Project là WordPadApp
- Đổi tên Form1 thành WordPad

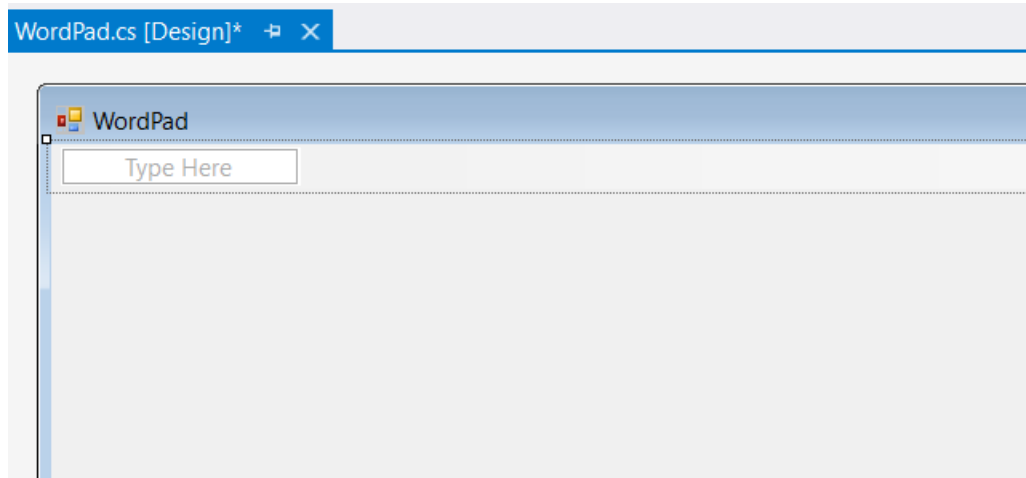


- Nhấp chuột phải vào WordPad, chọn Properties

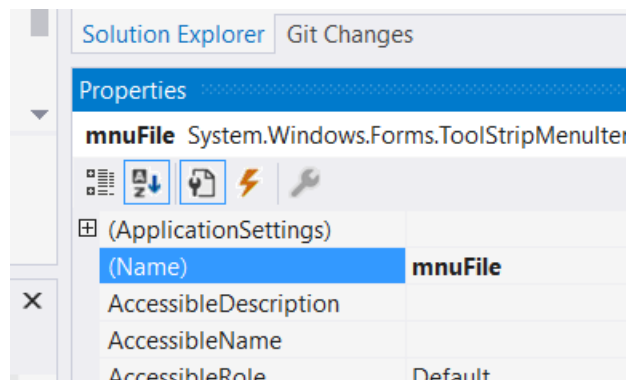
- Đặt một số thuộc tính của form như sau:
  - o Text: WordPad
  - o WindowsState: Maximized

## 2. Tạo menu

- Trong Toolbox, chọn điều khiển MenuStrip (nhấp đúp chuột vào MenuStrip hoặc nhấp chuột và kéo thả trên form)



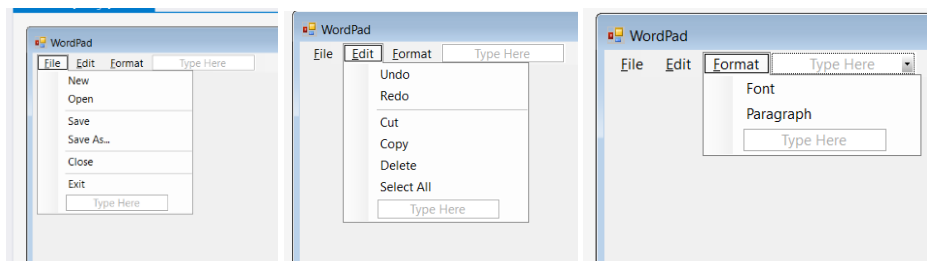
- Tạo các Menu File, Edit, Format,... Trong properties của các menu này, đặt tên (thuộc tính Name) là mnuFile, mnuEdit, mnuFormat...



Lưu ý: nếu cần sử dụng phím tắt nào, ta đặt & ở trước ký tự đó

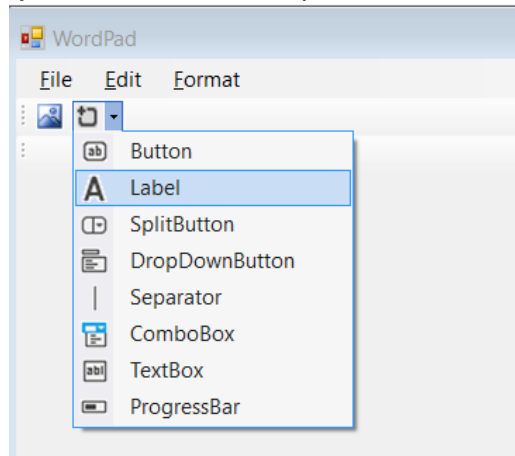
Ví dụ: &Edit thì phím tắt sẽ sử dụng là ALT+E

- Tạo các menu con cho các menu chính, sử dụng dấu – để tạo ngăn cách, đặt tên cho các menu tương ứng



### 3. Tạo các thanh thực đơn (ToolStrip)

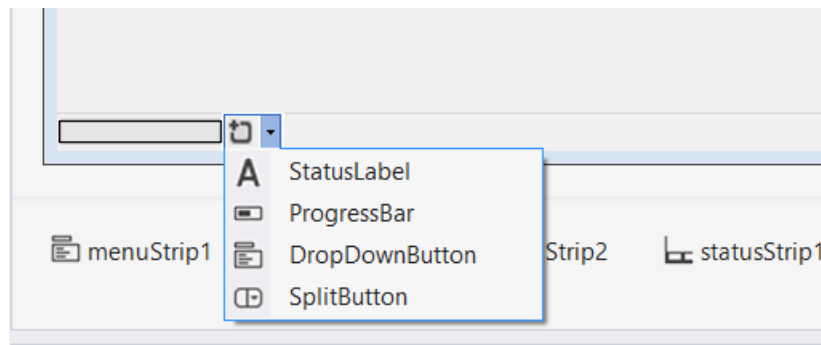
- Trong Toolbox, chọn ToolStrip (nhấp đúp vào ToolStrip, mỗi lần nhấp đúp thì một ToolStrip được thêm vào)



- Nhấp vào từng Nút của ToolStrip và chọn các điều khiển tương ứng.

### 4. Tạo StatusBar

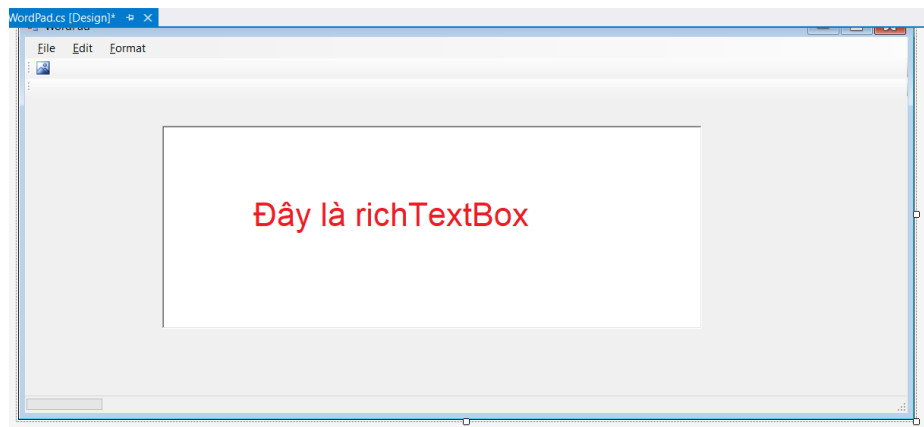
- Trong Toolbox, chọn StatusStrip (nhấp đúp chuột)



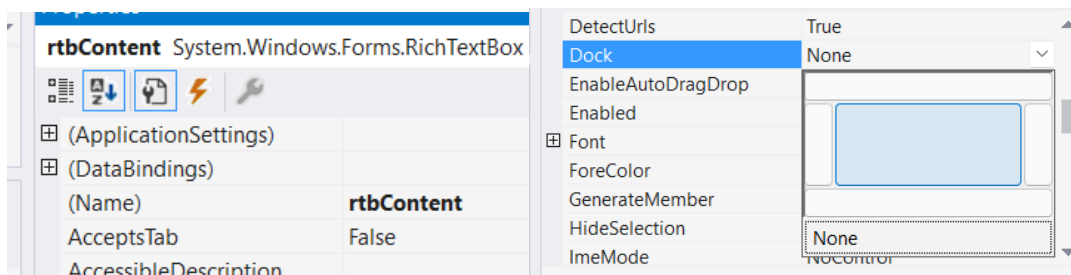
- Chọn vào biểu tượng để thêm các nội dung tương ứng
- Trong StatusStrip có các điều khiển sau Label, ProgressBar, DropDownButton, SplitButton

### 5. Vùng nội dung

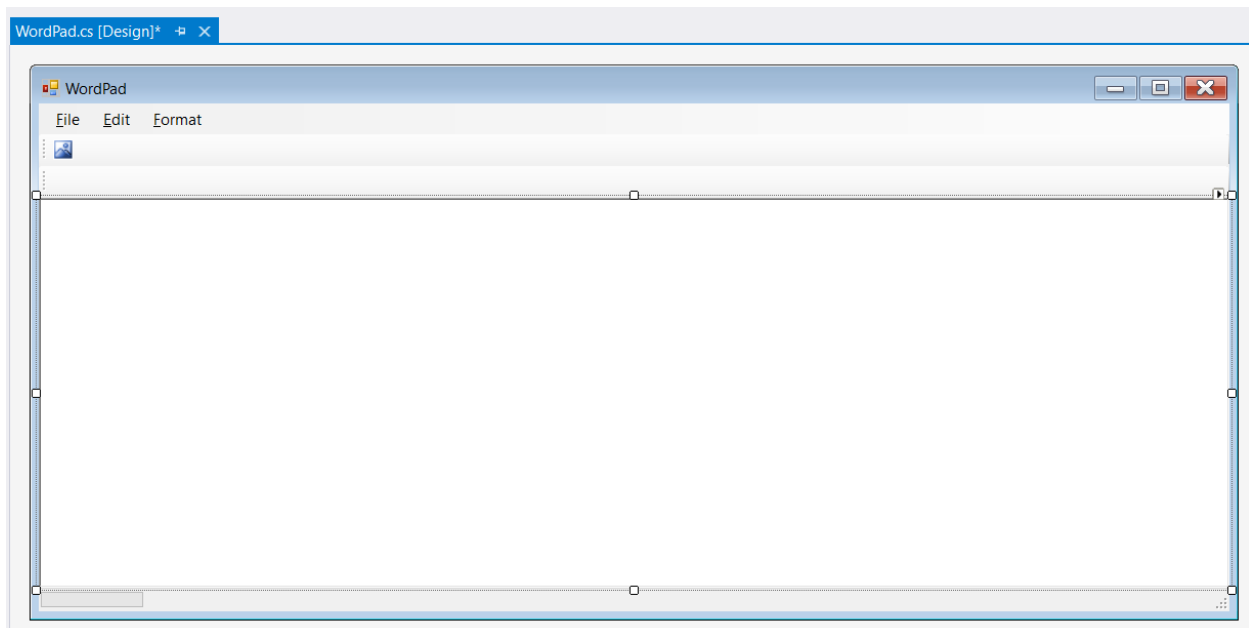
- Chèn RichTextBox vào vùng nội dung (trong Toolbox)



- Đặt thuộc tính cho richTextBox
  - o Name: rtbContent
  - o Dock: Fill



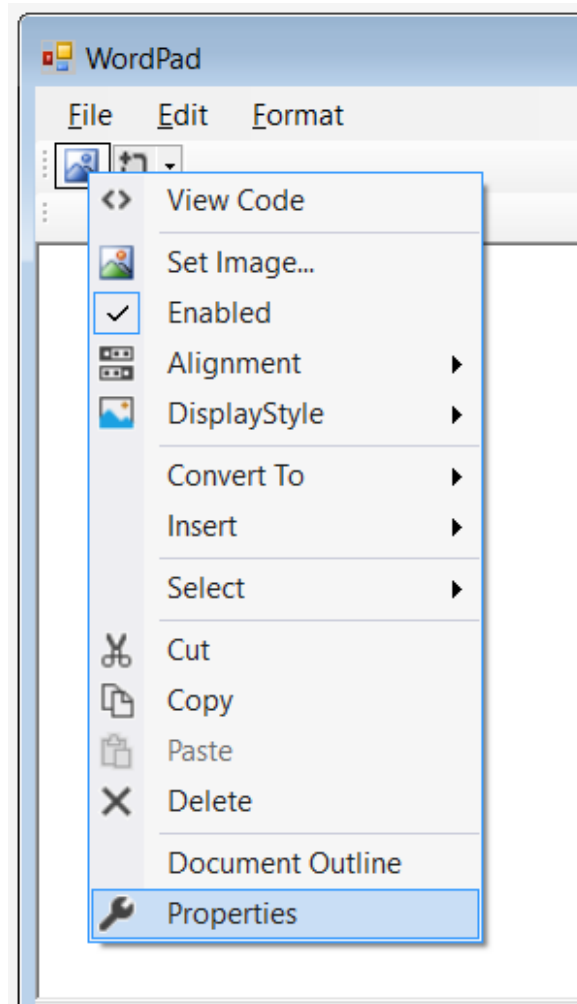
- Màn hình sau khi thiết lập Fill cho thuộc tính Dock



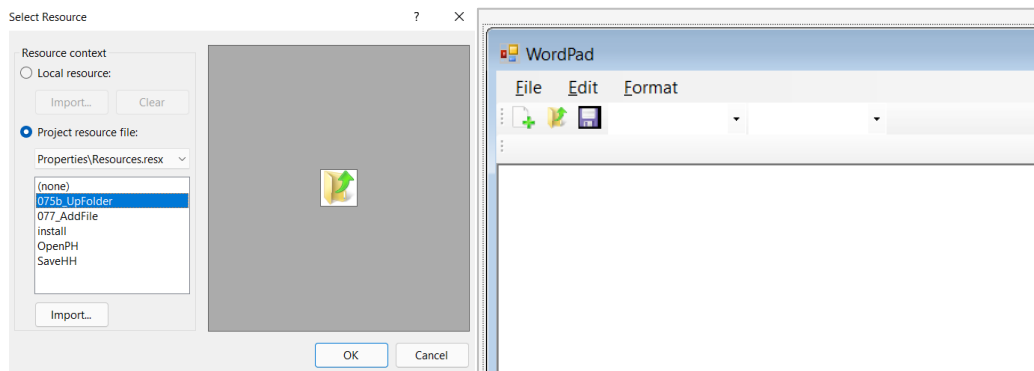
## 6. Thêm ảnh vào các nút

- Copy các ảnh vào thư mục project với link sau  
<https://vinhuni.edu.vn/pages/assets/images/icon.rar>

- Nhấp chuột phải vào nút, chọn Properties

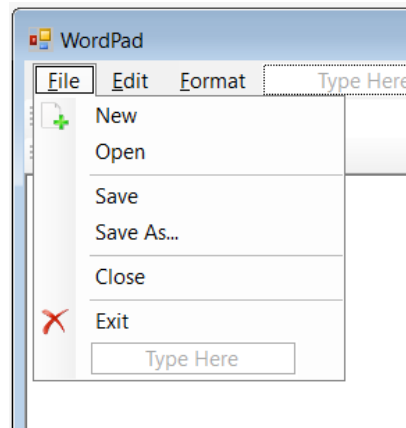


- Chọn thuộc tính Image, xuất hiện hộp thoại
- Chọn Import, chọn đường dẫn ảnh, OK



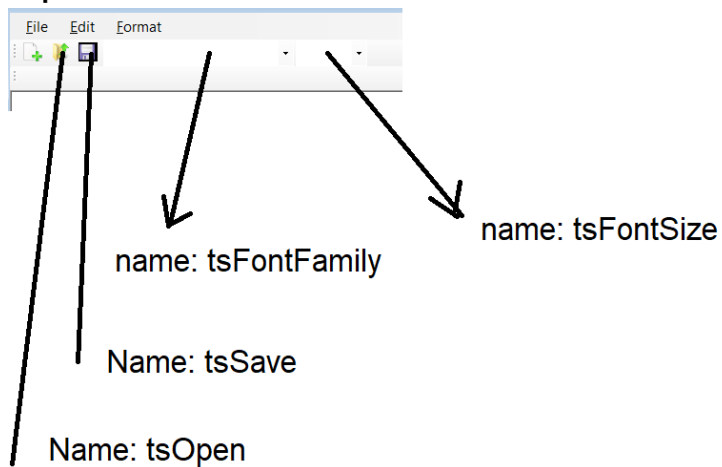
- Làm tương tự cho menu





## 7. Lập trình thực hiện các chức năng

- Đặt tên các điều khiển



- Nhấp đúp vào thanh tiêu đề của form để tạo sự kiện load cho Form WordPad

```
private void WordPad_Load(object sender, EventArgs e)
{
}

```

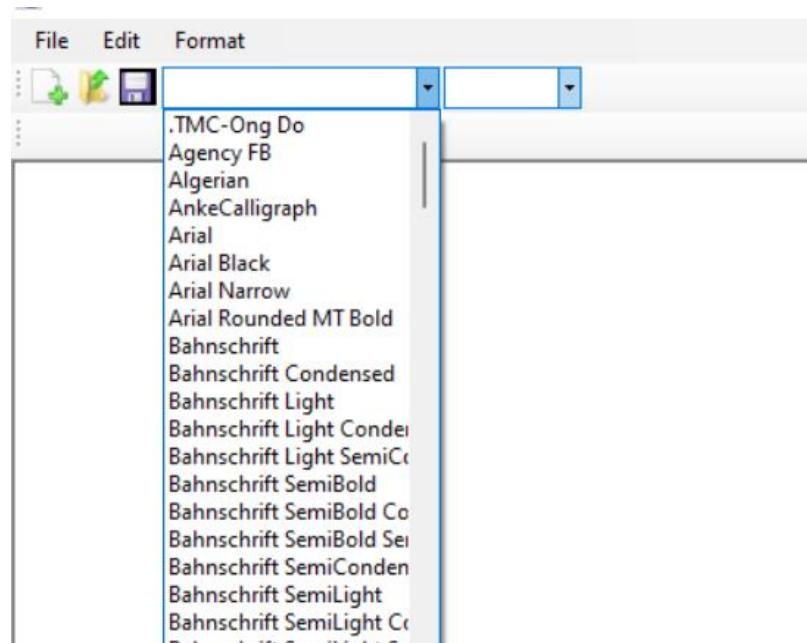
- Thêm các nội dung, trong đó

```

private void WordPad_Load(object sender, EventArgs e)
{
    //Load font hệ thống
    FontFamily[] ff = FontFamily.Families;
    for (int i = 0; i < ff.Length; i++)
    {
        tsFontFamily.Items.Add(ff[i].Name);
    }
    //Thêm vào font-size
    for (int i = 8; i <= 72; i++)
    {
        tsFontSize.Items.Add(i);
    }
    //Đặt mặc định
    tsFontFamily.Text = "Arial";
    tsFontSize.Text = "14";
}

```

- Sau khi thực hiện ta có




- Thêm hàm ChangeFont

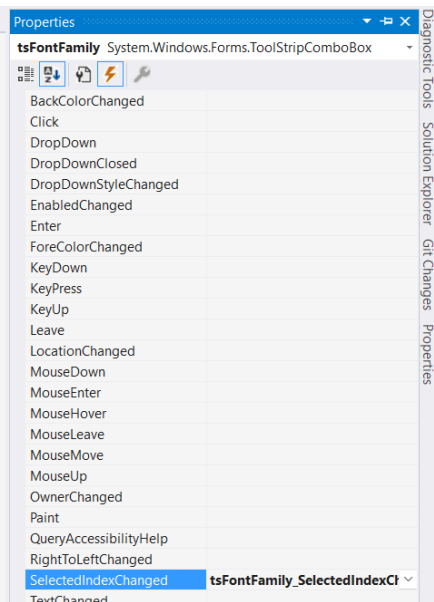
```

private void ChangeFont()
{
    //nếu xoá ô font family đưa về mặc định
    if (tsFontFamily.Text == "")
        tsFontFamily.Text = "Arial";
    //nếu xoá size đưa về mặc định
    if (tsFontSize.Text == "")
        tsFontSize.Text = "14";

    rtbContent.SelectionFont = new Font(tsFontFamily.Text, float.Parse(tsFontSize.Text));
}

```

- Tạo các sự kiện thay đổi font
  - Nhấp chuột phải vào tsFontFamily, chọn Properties
  - Chọn sự kiện 
  - Nhấp đúp vào SelectedIndexChanged



- Thêm hàm ChangeFont() trong sự kiện này, làm tương tự cho tsFontSize

```

private void tsFontFamily_SelectedIndexChanged(object sender, EventArgs e)
{
    ChangeFont();
}

```

1 reference

```

private void tsFontSize_SelectedIndexChanged(object sender, EventArgs e)
{
    ChangeFont();
}

```

- Thêm nút có biểu tượng B (bold), đặt tên tsBold
- Chọn sự kiện Click cho tsBold

```
private void tsBold_Click(object sender, EventArgs e)
{
    rtbContent.SelectionFont = new Font(rtbContent.SelectionFont,
        rtbContent.SelectionFont.Style ^ FontStyle.Bold);
}
```

- Làm tương tự cho các nút I, U
- Ghi lại tệp theo định dạng \*.doc cho nút Ghi (tsSave)

```
private void tsbSave_Click(object sender, EventArgs e)
{
    //Khởi tạo hộp thoại lưu tệp
    SaveFileDialog dr = new SaveFileDialog();
    dr.Filter = "Doc File|*.doc";

    if (dr.ShowDialog() == DialogResult.OK)
    {
        rtbContent.SaveFile(dr.FileName);
        MessageBox.Show("save successfully", "Address File : " +
            dr.FileName, MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
}
```

- Mở tệp theo định dạng \*.doc cho nút Open (tsOpen)

```
private void tsbOpen_Click(object sender, EventArgs e)
{
    //Khởi tạo hộp thoại mở tệp
    OpenFileDialog file = new OpenFileDialog();
    file.Filter = "DOC Files|*.doc";

    //Nếu ấn ok và có chọn tệp
    if (file.ShowDialog() == System.Windows.Forms.DialogResult.OK &&
        file.FileName.Length > 0)
    {
        // Nạp nội dung lên rbtContent
        rtbContent.LoadFile(file.FileName);
    }
}
```

- Các chức năng khác tham khảo trên Internet.

<https://www.youtube.com/watch?v=0CdRhNXJTAE>