Principles

Cory House
PRINCIPAL CONSULTANT

@housecor www.cleancsharp.net

Agenda

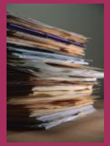
Clean Code Principles

Three Clean Code Principles

Clean Code Principles

1) Right tool for the job

2) High signal to noise ratio

3) Self-documenting 

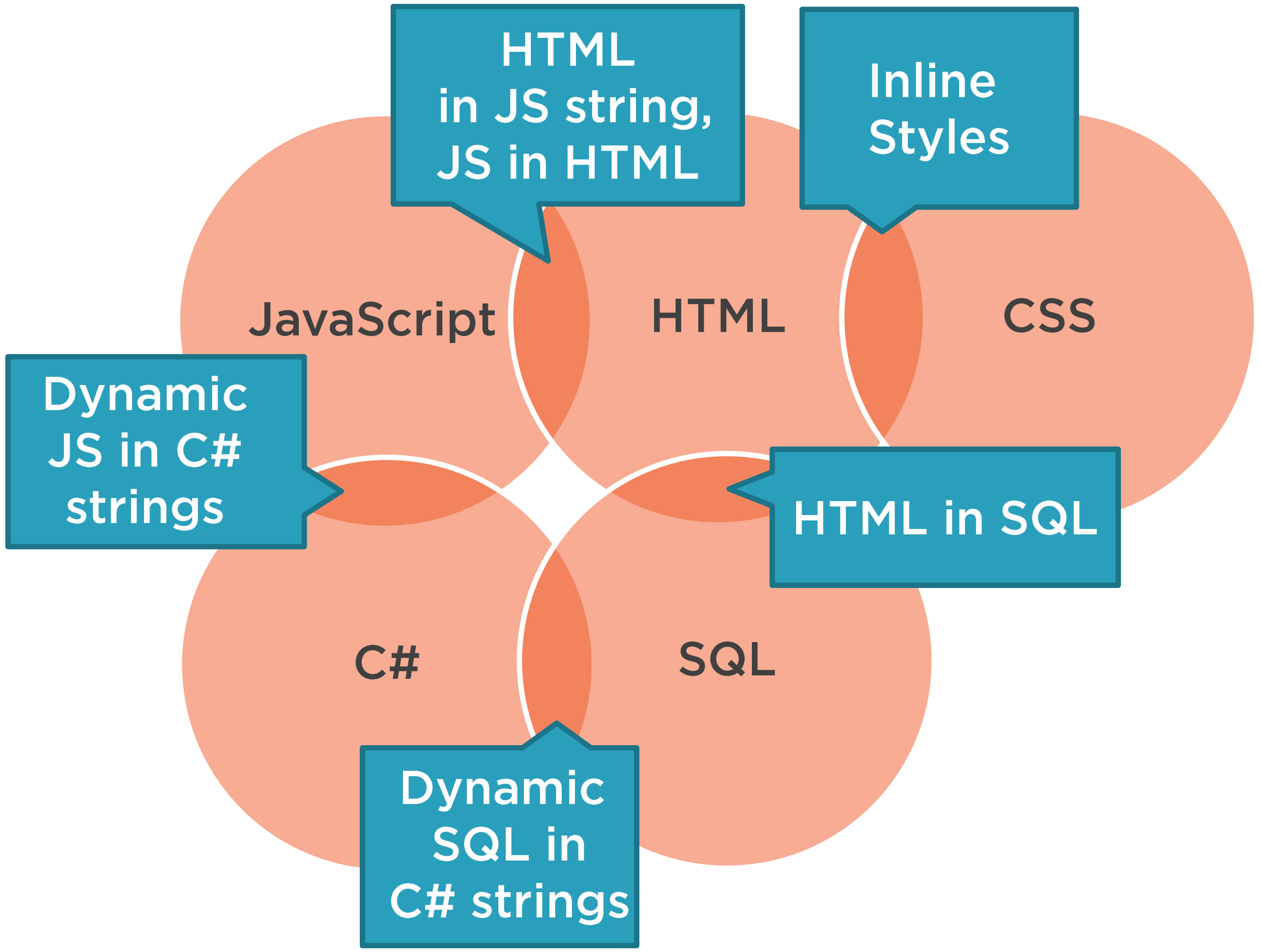
1) The Right Tool for the Job

```
(?:(?:\r\n)?[ \t])*(?:(?:(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t] )+|\Z|(?=[\["()<>@,;:\\".\[\]]))|"(?:[^\"\r\\]|\\.|(?:(?:\r\n)?[ \t]))*"(?:(?: \r\n)?[ \t])*)(?:\.(?:(?:\r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-
\031]+(?:(?:( ?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|"(?:[^\"\r\\]|\\.|(?:(?:\r\n)?[ \t]))*"(?:(?:\r\n)?[ \t])*))*@(?:(?:\r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-\0 31]+(?:(?:(?:\r\n)?[
\t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|\[([^\[\]\r\\]|\\.)*\ ](?:(?:\r\n)?[ \t])*)(?:\.(?:(?:\r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-\031]+ (?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|\[([^\[\]\r\\]|\\.)*\](?: (?:\r\n)?[
\t])*))*|(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z |(?=[\["()<>@,;:\\".\[\]]))|"(?:[^\"\r\\]|\\.|(?:(?:\r\n)?[ \t]))*"(?:(?:\r\n) ?[ \t])*)*\<(?:(?:\r\n)?[ \t])*(?:@(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\
r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|\[([^\[\]\r\\]|\\.)*\](?:(?:\r\n)?[ \t])*)(?:\.(?:(?:\r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n) ?[ \t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|\[([^\[\]\r\\]|\\.)*\](?:(?:\r\n)?[
\t] )*))*(?:,@(?:(?:\r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|\[([^\[\]\r\\]|\\.)*\](?:(?:\r\n)?[ \t])* )(?:\.(?:(?:\r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-
\031]+(?:(?:(?:\r\n)?[ \t] )+|\Z|(?=[\["()<>@,;:\\".\[\]]))|\[([^\[\]\r\\]|\\.)*\](?:(?:\r\n)?[ \t])*))*) *:(?:(?:\r\n)?[ \t])*)?(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+
|\Z|(?=[\["()<>@,;:\\".\[\]]))|"(?:[^\"\r\\]|\\.|(?:(?:\r\n)?[ \t]))*"(?:(?:\r \n)?[ \t])*)(?:\.(?:(?:\r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?: \r\n)?[
\t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|"(?:[^\"\r\\]|\\.|(?:(?:\r\n)?[ \t ]))*"(?:(?:\r\n)?[ \t])*))*@(?:(?:\r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-\031 ]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|\[([^\[\]\r\\]|\\.)*\](
?:(?:\r\n)?[ \t])*)(?:\.(?:(?:\r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-\031]+(? :(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|\[([^\[\]\r\\]|\\.)*\](?:(? :\r\n)?[ \t])*))*\>(?:(?:\r\n)?[ \t])*)|(?:[^()<>@,;:\\".\[\] \000-
\031]+(?:(? :(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|"(?:[^\"\r\\]|\\.|(?:(?:\r\n)? [ \t]))*"(?:(?:\r\n)?[ \t])*)*:(?:(?:\r\n)?[ \t])*(?:(?:(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[
\t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|"(?:[^\"\r\\]| \\.|(?:(?:\r\n)?[ \t]))*"(?:(?:\r\n)?[ \t])*)(?:\.(?:(?:\r\n)?[ \t])*(?:[^()<> @,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|"
(?:[^\"\r\\]|\\.|(?:(?:\r\n)?[ \t]))*"(?:(?:\r\n)?[ \t])*))*@(?:(?:\r\n)?[ \t] )*(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\ ".\[\]]))|\[([^\[\]\r\\]|\\.)*\](?:(?:\r\n)?[ \t])*)(?:\.(?:(?:\r\n)?[
\t])*(? :[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\".\[ \]]))|\[([^\[\]\r\\]|\\.)*\](?:(?:\r\n)?[ \t])*))*|(?:[^()<>@,;:\\".\[\] \000- \031]+(?:(?:(?:\r\n)?[
\t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|"(?:[^\"\r\\]|\\.|( ?:(?:\r\n)?[ \t]))*"(?:(?:\r\n)?[ \t])*)*\<(?:(?:\r\n)?[ \t])*(?:@(?:[^()<>@,; :\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|\[([
^\[\]\r\\]|\\.)*\](?:(?:\r\n)?[ \t])*)(?:\.(?:(?:\r\n)?[ \t])*(?:[^()<>@,;:\\" .\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|\[([^\[\ ]\r\\]|\\.)*\](?:(?:\r\n)?[ \t])*))*(?:,@(?:(?:\r\n)?[
\t])*(?:[^()<>@,;:\\".\ [\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|\[([^\[\]\ r\\]|\\.)*\](?:(?:\r\n)?[ \t])*)(?:\.(?:(?:\r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[
\t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|\[([^\[\]\r\\ |\\.)*\](?:(?:\r\n)?[ \t])*))*)*:(?:(?:\r\n)?[ \t])*)?(?:[^()<>@,;:\\".\[\] \0 00-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|"(?:[^\"\r\\]|\\ .|(?:(?:\r\n)?[
\t]))*"(?:(?:\r\n)?[ \t])*)(?:\.(?:(?:\r\n)?[ \t])*(?:[^()<>@, ;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|"(? :[^\"\r\\]|\\.|(?:(?:\r\n)?[ \t]))*"(?:(?:\r\n)?[ \t])*))*@(?:(?:\r\n)?[ \t])*
(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\". \[\]]))|\[([^\[\]\r\\]|\\.)*\](?:(?:\r\n)?[ \t])*)(?:\.(?:(?:\r\n)?[ \t])*(?:[ ^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[
\t])+|\Z|(?=[\["()<>@,;:\\".\[\] ]))|\[([^\[\]\r\\]|\\.)*\](?:(?:\r\n)?[ \t])*))*\>(?:(?:\r\n)?[ \t])*)(?:,\s*( ?:(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\
".\[\]]))|"(?:[^\"\r\\]|\\.|(?:(?:\r\n)?[ \t]))*"(?:(?:\r\n)?[ \t])*)(?:\.(?:( ?:\r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[ \["()<>@,;:\\".\[\]]))|"(?:[^\"\r\\]|\\.|(?:(?:\r\n)?[ \t]))*"(?:(?:\r\n)?[
\t ])*))*@(?:(?:\r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t ])+|\Z|(?=[\["()<>@,;:\\".\[\]]))|\[([^\[\]\r\\]|\\.)*\](?:(?:\r\n)?[ \t])*)(? :\.(?:(?:\r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-
\031]+(?:(?:(?:\r\n)?[ \t])+| \Z|(?=[\["()<>@,;:\\".\[\]]))|\[([^\[\]\r\\]|\\.)*\](?:(?:\r\n)?[ \t])*))*|(?: [^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\".\[\ ]]))|"(?:[^\"\r\\]|\\.|(?:(?:\r\n)?[
\t]))*"(?:(?:\r\n)?[ \t])*)*\<(?:(?:\r\n) ?[ \t])*(?:@(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\[" ()<>@,;:\\".\[\]]))|\[([^\[\]\r\\]|\\.)*\](?:(?:\r\n)?[ \t])*)(?:\.(?:(?:\r\n) ?[ \t])*(?:[^()<>@,;:\\".\[\]
\000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<> @,;:\\".\[\]]))|\[([^\[\]\r\\]|\\.)*\](?:(?:\r\n)?[ \t])*))*(?:,@(?:(?:\r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,
;:\\".\[\]]))|\[([^\[\]\r\\]|\\.)*\](?:(?:\r\n)?[ \t])*)(?:\.(?:(?:\r\n)?[ \t] )*(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\ ".\[\]]))|\[([^\[\]\r\\]|\\.)*\](?:(?:\r\n)?[ \t])*))*)*:(?:(?:\r\n)?[
\t])*)? (?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z|(?=[\["()<>@,;:\\". \[\]]))|"(?:[^\"\r\\]|\\.|(?:(?:\r\n)?[ \t]))*"(?:(?:\r\n)?[ \t])*)(?:\.(?:(?: \r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[
\t])+|\Z|(?=[\[ "()<>@,;:\\".\[\]]))|"(?:[^\"\r\\]|\\.|(?:(?:\r\n)?[ \t]))*"(?:(?:\r\n)?[ \t]) *))*@(?:(?:\r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])
+|\Z|(?=[\["()<>@,;:\\".\[\]]))|\[([^\[\]\r\\]|\\.)*\](?:(?:\r\n)?[ \t])*)(?:\ .(?:(?:\r\n)?[ \t])*(?:[^()<>@,;:\\".\[\] \000-\031]+(?:(?:(?:\r\n)?[ \t])+|\Z |(?=[\["()<>@,;:\\".\[\]]))|\[([^\[\]\r\\]|\\.)*\](?:(?:\r\n)?[
\t])*))*\>(?:( ?:\r\n)?[ \t])*))*)?;\s*)
```

http://ex-parrot.com/~pdw/Mail-RFC822-Address.html

Let's use
<lunatic's favorite tool>
for *everything*

Watch for Boundaries



```
string script = @"<script type=""text/javascript"" defer=""defer"">
                    //<![CDATA[
                        var _gaq = _gaq || [];
                        _gaq.push(['_setAccount', '" + ws.GoogleAnalyticsID + @"']);
                        _gaq.push(['_trackPageview']);

                        (function() {
                        var ga = document.createElement('script');
                        ga.src = ('https:' == document.location.protocol ? 'https://ssl' : 'http://www') +
                        '.google-analytics.com/ga.js';
                        ga.setAttribute('async', 'true');
                        document.documentElement.firstChild.appendChild(ga);
                        })();
                    //]]>
                    </script>";
this.Header.Controls.Add(new LiteralControl("\r\n" + script));
```

✓

```
//In GoogleAnalytics.js
var _gaq = _gaq || [];
_gaq.push(['_setAccount', WebSiteSetup.GoogleAnalyticsKey]);
_gaq.push(['_trackPageview']);

(function () {
   var ga = document.createElement('script');
   ga.src = ('https:' == document.location.protocol ? 'https://ssl' : 'http://www') +
   '.google-analytics.com/ga.js';
   ga.setAttribute('async', 'true');
   document.documentElement.firstChild.appendChild(ga);
})();
```

```
<!--In document head-->
<script type="text/javascript">
    var WebSiteSetup = { "GoogleAnalyticsKey": "JDSGI832JDUG9831" };
</script>
```

Stay Native - Advantages

Cached

Code colored

Syntax checked

Separation of concerns

Reusable

Avoids string parsing

Can minify & obfuscate

Avoid using one language to write another language/format via strings.

Avoid using C# strings to create:

- JavaScript
- XML
- HTML
- JSON
- CSS

Leverage libraries

One language per file.

Every Tech is Potential Evil

Tech	Potential Evil
LINQ-2-SQL	**Massive queries, outer joins**
SQL	**Embedding display logic**
C#	**Writing JS & HTML in strings**

2) Maximize Signal to Noise Ratio

Signal

Terse
Expressive
Does one thing

Noise

High cyclomatic complexity	Huge classes
Excessive indentation	Long methods
Zombie code	Repetition
Unnecessary comments	No whitespace
Poorly named structures	Overly verbose

So. Much. Noise.

🚫

```
 #region -- InitCountryDropdown Method --
 /// <summary>
 /// Summary description for CountryDropDownList.
 /// </summary>
protected override void InitCountryDropdown(EventArgs e)
{
   if (Items.Count == 0)
   {
      this.DataSource = CountryTable();
      this.DataTextField = "CountryName";

      //CH 2012-4-5 - Adding separate data value field
      //to fix bug #4535
      //We're now storing the country ID instead
      //of the country name if desired
      if (useCountryName == true)
      {
          this.DataValueField = "CountryName";
      }
      else
      {
          this.DataValueField = "CountryID";
      }

      this.DataBind();
      this.CssClass = "entryfield";
   }
}
#endregion
```

✓

```
protected override void InitCountryDropdown(EventArgs e)
{
    if (Items.Count > 0) return;

    this.DataSource = CountryTable();
    this.DataTextField = "CountryName";
    this.DataValueField = useCountryName ? "CountryName" : "CountryID";
    this.DataBind();
    this.CssClass = "entryfield";
}
```

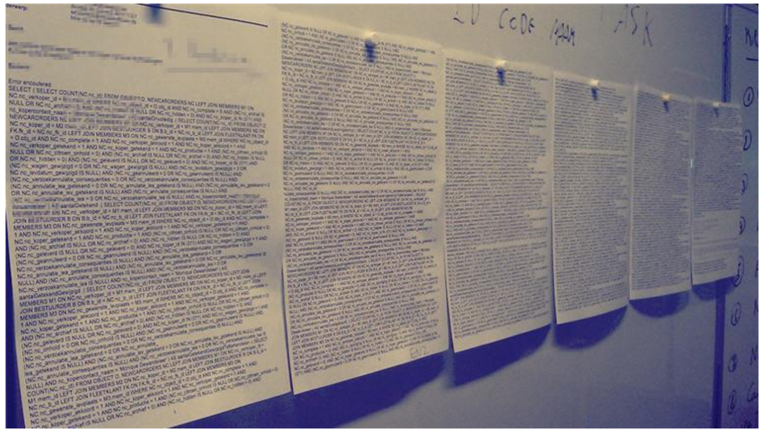
Goal: Density. More in each "eye full"

Why is Signal to Noise Ratio Important?

1. **Our brain is the compiler**

=

2. **The mess builds quietly**

=


DRY Principle: Don’t Repeat Yourself

Same principle as DB normalization

Copy and paste is often a design problem

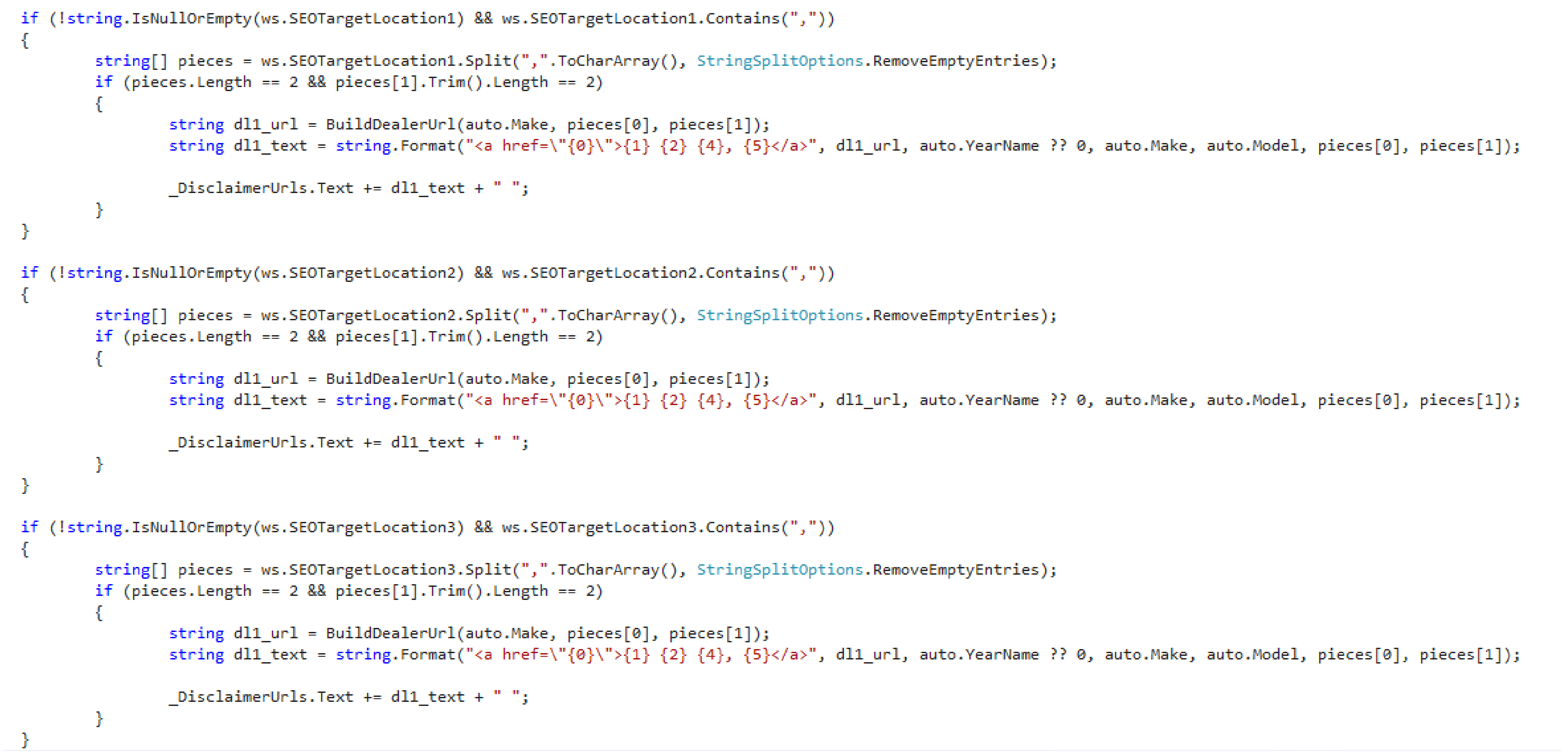
DRY Principle: Don’t Repeat Yourself

Duplication Issues

1. Decreases signal to noise ratio
2. Increases the number of lines of code
3. Creates a maintenance problem

Copy and paste is often a design problem

Look for Patterns



```
if (!string.IsNullOrEmpty(ws.SEOTargetLocation1) && ws.SEOTargetLocation1.Contains(","))
{
        string[] pieces = ws.SEOTargetLocation1.Split(",".ToCharArray(), StringSplitOptions.RemoveEmptyEntries);
        if (pieces.Length == 2 && pieces[1].Trim().Length == 2)
        {
                string dl1_url = BuildDealerUrl(auto.Make, pieces[0], pieces[1]);
                string dl1_text = string.Format("<a href=\"{0}\">{1} {2} {4}, {5}</a>", dl1_url, auto.YearName ?? 0, auto.Make, auto.Model, pieces[0], pieces[1]);

                _DisclaimerUrls.Text += dl1_text + " ";
        }
}

if (!string.IsNullOrEmpty(ws.SEOTargetLocation2) && ws.SEOTargetLocation2.Contains(","))
{
        string[] pieces = ws.SEOTargetLocation2.Split(",".ToCharArray(), StringSplitOptions.RemoveEmptyEntries);
        if (pieces.Length == 2 && pieces[1].Trim().Length == 2)
        {
                string dl1_url = BuildDealerUrl(auto.Make, pieces[0], pieces[1]);
                string dl1_text = string.Format("<a href=\"{0}\">{1} {2} {4}, {5}</a>", dl1_url, auto.YearName ?? 0, auto.Make, auto.Model, pieces[0], pieces[1]);

                _DisclaimerUrls.Text += dl1_text + " ";
        }
}

if (!string.IsNullOrEmpty(ws.SEOTargetLocation3) && ws.SEOTargetLocation3.Contains(","))
{
        string[] pieces = ws.SEOTargetLocation3.Split(",".ToCharArray(), StringSplitOptions.RemoveEmptyEntries);
        if (pieces.Length == 2 && pieces[1].Trim().Length == 2)
        {
                string dl1_url = BuildDealerUrl(auto.Make, pieces[0], pieces[1]);
                string dl1_text = string.Format("<a href=\"{0}\">{1} {2} {4}, {5}</a>", dl1_url, auto.YearName ?? 0, auto.Make, auto.Model, pieces[0], pieces[1]);

                _DisclaimerUrls.Text += dl1_text + " ";
        }
}
```

3) Self-documenting Code

Understanding the original programmer's intent is the most difficult problem.
- Fjelstad & Hamlen 1979

Well written code is self-documenting.

Clear intent

Layers of abstractions

Format for readability

Favor code over comments

Summary

Use the right tool

- Watch for boundaries
- Stay native
- One language per file

Maximize the signal to noise ratio

- TED: Terse, expressive, does one thing
- DRY: Don't repeat yourself
- Watch for patterns

Strive for self-documenting code

Next up: Naming