Семинар 8.

Примеры задач.

№1. Предположим, что список рассылки содержит имена, в которые иногда входит обращение (**Mr.**, **Mrs.**, **Miss** или **Ms.**) в дополнение к имени и фамилии. Если вы не хотите включать обращения при создании этикеток для конвертов из списка, с помощью регулярного выражения их можно удалить, как показано в примере.

Шаблон регулярного выражения **(Mr\\.? |Mrs\\.? |Miss |Ms\\.?)** сопоставляет любые вхождения строк "**Mr**", "**Mr.**", "**Mrs**", "**Mrs.**", "**Miss**", "**Ms**" и (или) "**Ms.**". После вызова метода **Regex.Replace** сопоставленная строка заменяется на **String.Empty**; другими словами, она удаляется из исходной строки.

№2. Случайный повтор слов — это распространенная ошибка при написании текстов. Регулярное выражение можно использовать для определения повторяющихся слов, как показано в следующем примере.

Шаблон регулярного выражения **\b(\w+?)\s\1\b** интерпретируется следующим образом:

|  |  |
| --- | --- |
| **\b** | Начало на границе слова. |
| **(\w+?)** | Соответствует одному или нескольким символам слова (как можно меньшему количеству). Вместе они формируют группу, к которой можно обращаться как к **\1**. |
| **\s** | Соответствует пробелу. |
| **\1** | Сопоставление подстроки, равной группе с именем **\1**. |
| **\b** | Соответствует границе слова. |

Метод **Regex.Matches** вызывается с параметрами регулярного выражения **RegexOptions.IgnoreCase**. Поэтому операция сопоставления учитывает регистр, а пример указывает, что подстрока "**This this**" является повтором.

Обратите внимание, что входная строка содержит подстроку "**this? This**". Но из-за знака пунктуации она не считается повторением.