[**Тестирование полей ввода**](http://blog.shumoos.com/archives/67)

Для полей ввода проверяются:

1. Обработка корректных данных
2. Обработка граничных условий
3. Обработка некорректных данных внутри граничных условий
4. Обработка некорректных данных вне граничных условий
5. Обработка сложных данных

Все просто и многократно описано в книгах. Так если можно вводить только целые числа от 0 до 100 включительно, то будем проверять обработку:

1. 13 (любое целое число от 0 до 100)
2. -1, 0, 100, 101. Рекомендуют еще проверить обработку 1 и 99, но, на мой взгляд это излишество. Вот если бы в условиях стояло “исключая 0 и 100″, то тогда нужно было бы проверить 0, 1, 99, 100.
3. 3.14
4. -8, 115. Хотя, если -1 и 101 были обработаны нормально, то эта проверка уже лишняя. Стоит попробовать ввести буквы, знаки препинания.
5. в данном случае нет.

Неужели это все? Нет, есть еще несколько тонкостей.

**Различия в обработке некорректных данных между Web и Win32 приложениями.**Win32 приложения не должны давать вводить в поле ввода некорректные символы. Реализация может быть различной. Может быть просто блокировка ввода, может выскакивать маленький предупреждающий символ слева от поля ввода. Иногда подают звуковой сигнал или выводят модальное окно- предупреждение. Но последние два способа не эргономичны и я считаю их ошибками проектирования интерфейса.  
В Web-приложениях обработка, как правило, ведется на сервере. Соответственно, пользователю должна вернуться та же самая форма ввода, содержащая: все, что пользователь ввел (не заставляйте человека вводить данные дважды!), пометки около всех неправильных полей, и сообщение, что же было неверным.

**Сложные символы**  
Пусть у нас есть поле ввода названия фирмы. Обычно, я проверяю поля на обработку строки типа такой:  
“[|]’~< !--@/\*$%^&#\*/()?>,.\*/\

Внимание! При использования этой строки уберите пробел между < и !--. Пробел вставлен, т.к. движок блога падает на этом сочетании.

Здесь интересно понять какие ошибки где могут встретиться.

* Апостроф – символ, встречающийся в SQL выражениях, и при неправильной его обработке, упадет операция вставки в базу данных.
* % и ? – части LIKE SQL запроса. Ошибки, связанные с ними в моей практике не встречались.
* Кавычки и угловые скобки – части кода html. Ошибки, связанные с ними, скорее встретятся в ыеб приложении
* & - часто «проглатывается» в Win32 приложениях.
* /\* и \*/ и — . Для MS SQL server иногда очень убийственно (прислано Петраш А.Ю.).
* | вместе с символами \ / : \* ? ” < > запрещён к использованию в именах файлов и папок(прислано Fyz).
* < !--, очень часто получается, что после сохранения страница отображается неверно (прислано Смирновым Александром). На эту ошибку, в частности я натолкнулся в движке этого блога.
* Символы # и & по отдельности воспринимались нормально, а вот если их ввести последовательно… То есть желательно проверять еще иногда и разные комбинации спец.символов (прислано Triss).

**Обработка белых пробелов**

* В однострочных полях следует обрезать «белые пробелы» в начале и конце строки.
* Символ табуляции внутри строки заменять пробелами  
  Дополнение. Изредка встречается копирование пользователем из буфера обмена в поле ввода нескольких строк. Для этого случая применяем правило:
* Информация после первого встреченного символа конца строки должна обрезаться. Данное правило накладывается после правила обрезки белых пробелов.

**Соответствие длин поля ввода и хранимых данных**

* Длина поля ввода с неизменяемым размером не должна быть больше максимальной длины ввода. Если из соображений оформления длина поля больше, то следует не позволять пользователю выходить за пределы. В любом случае недопустима потеря данных.
* Введенные и отображаемые впоследствии данные должны быть одинаковыми.
* Обязательность заполнения поля должна контролироваться не на уровне БД, а на уровне клиента, в случае Win32 приложений, или сервера, в случае Web приложения.

Разберем эти правила более подробно. Пусть на форме ввода нужно заполнить три поля: Фамилию, Имя и Отчество. Из соображений красоты дизайнер задал длину этих полей одинаковыми, и равными 20 символов. Все хорошо, но ДБА задал длину полей в БД 32, 16 и 16 соответственно.

* Первая характерная ошибка. Программист не контролирует длину ввода. При вводе имени в 17 символов программа падает.
* Вторая характерная ошибка. Ставится ограничение по вводу в 20, 16, 16 символов соответственно. Часть поля в БД не используется. Ошибка непринципиальная.
* Третья характерная ошибка. Поля Фамилия и Имя указаны как обязательные к заполнению на уровне БД, но этой обработки нет на клиенте. При попытке передачи данных без имени приложение падает.
* Четвертая характерная ошибка. Для фамилии длина поля оставлена 20 символов, но разрешен ввод 32. При вводе длинной фамилии оператор не видит начала поля. Так иногда делают, но это не очень хорошо.