

ИНСТРУКЦИЯ ПО РАЗРАБОТКЕ ВАЛИДАТОРОВ

Валидаторы нужны для того, чтобы ограничить вводимый пользователем текст. Они требуются вам, если в условии вашей лабораторной работы заданы ограничения на значения строковых полей.

1. Создание класса :

1. Выберите имя, отражающее суть валидатора, и добавьте к нему слово валидатор — в нашем примере мы назовем его ExampleValidator
2. В меню Project выберите Add Qt Class
3. В появившемся окне в качестве базового класса введите QValidator, в качестве имени своего класса — ExampleValidator

Welcome to the Qt4 Class Wizard

This wizard will add a new Qt4 class to your project. The wizard creates a .h and .cpp file.

Class name:	.h file:	.cpp file:
ExampleValidator	examplevalidator.h	examplevalidator.cpp
Base class:	Constructor signature:	<input checked="" type="checkbox"/> Insert Q_OBJECT
QValidator	QObject *parent	<input checked="" type="checkbox"/> Lower case file names

4. В вашем проекте появятся файлы h и cpp для созданного класса.
- ### 2. Описание класса

1. В public часть описания класса (h файл) добавьте функцию проверки `QValidator::State validate(QString & input, int & pos) const;`
2. Также в описание класса добавьте конструктор по умолчанию (без параметров).
3. В файле cpp скопируйте код конструктора с параметрами и отредактируйте текст копии, чтобы получилось следующее:

```
ExampleValidator::ExampleValidator()  
    : QValidator(NULL)  
{  
  
}
```

4. В файле cpp опишите код функции проверки под заголовком `QValidator::State ExampleValidator::validate(QString & input, int & pos) const`. Функция получает строку и позицию курсора в ней, и должна возвращать одно из следующих значений:
 1. `QValidator::Invalid` — строка явно нарушает ограничения
 2. `QValidator::Intermediate` — строка не удовлетворяет всем ограничениям, но если к ней добавить еще какой-либо текст, то она сможет им удовлетворить (т.е. ничего не нарушено, но до правильного содержания чего-либо не хватает)
 3. `QValidator::Acceptable` — строка полностью удовлетворяет

всем ограничениям

5. Использование валидатора

1. В заголовок класса главного окна добавьте переменную (не указатель!) с типом созданного валидатора. В этот же файл (до описания класса) не забудьте подключить заголовочный файл класса валидатора.
2. В конструкторе окна вызовите функцию `setValidator` для необходимого виджета и передайте ей адрес переменной-валидатора.
3. Правильность введенного текста можно проверить используя функцию `hasAcceptableInput()` для соответствующего виджета. Строки, для которых валидатор генерирует оценку `Invalid` будут блокироваться, ввести их пользователю невозможно.