|  |  |
| --- | --- |
| ДИСЦИПЛИНА | Технологии индустриального программирования |
| ИНСТИТУТ | ИПТИП |
| КАФЕДРА | Индустриального программирования |
| ВИД УЧЕБНОГО МАТЕРИАЛА | Методические указания к практическим занятиям по дисциплине |
| ПРЕПОДАВАТЕЛЬ | Александров Алексей Сергеевич |
| СЕМЕСТР | 3 семестр, 2024/2025 уч. год |

# Практическое занятие №3

**Представление систем счисления в Qt**

При выполнении предыдущей работы, числовое значение из lineEdit считывалось напрямую, конвертируя значение текстового поля в число, без проверок:

int value\_1 = ui->lineEdit\_value\_1->text().toInt();

Для проверки необходимо записать текстовое значение в отдельную переменную, а также создать логическую переменную (bool), в которую будет производиться запись корректности результата конвертации строкового значения в число.

Строковые значения в Qt представляются типом данных (классом) QString, предоставляющий мощный дополнительный функционал работы со строками.

В данном случае необходимо записать значение из lineEdit в текстовую переменную:

QString text\_1 = ui->lineEdit\_value\_1->text();

QString text\_2 = ui->lineEdit\_value\_2->text();

После чего создать 2 логических переменных:

bool correct\_1 = false;

bool correct\_2 = false;

И выполнить конвертацию строкового значения в число, передав адрес созданной логической переменной в метод QString::toInt():

int value\_1 = text\_1.toInt(&correct\_1);

int value\_2 = text\_2.toInt(&correct\_2);

В результате вызова метода QString::toInt() в переменную “correct\_*x*” записывается значение “true”, если строка была успешно конвертирована в число, иначе – “false” – если конвертация не удалась. В случае неудачной конвертации, в переменную “value\_*x*” будет записано число “0”, независимо от того, что в ней было записано ранее.

По умолчанию, метод toInt() предполагает, что число, записанное в строке находится в десятиной системе счисления, однако, если число записано в другой системе, это необходимо указать, передав методу toInt() дополнительный аргумент, указывающий систему счисления:

int value\_1 = text\_1.toInt(&correct\_1, 16);

В данном случае число, записанное в текстовое поле в 16й системе счисления, будет переведено методом toInt() в десятичную. Таким образом, можно переводить числа из разных систем счисления в десятичную и выполнять с ними различные математические действия.

При указании базовой системы счисления корректность конвертации будет проверяться согласно указанной базовой системе – т.е., если в строке будет указано число “AF10”, то при указании 10 системы счисления метод toInt() сообщит об ошибке, в то время, как при указании 16 системы – число будет корректно конвертировано.

Важно отметить, что в строковой представлении (QString) число может быть записано в любой системе счисления, однако, тип данных int поддерживает работу только в десятичной системе счисления.

Таким образом можно проверить корректность выполнения конвертации:

if(correct\_1 && correct\_2) // если оба значения корректны

{

// код при корректных значениях

}

else

{

// код при некорректных значениях

}

Конвертировать можно, также, и число в строку, используя метод QString::number():

QString result = QString::number(sum);

А если указать основание системы счисления вторым аргументом, то метод выполнит перевод из десятичной системы счисления в указанную, например, в шестнадцатеричную:

QString result = QString::number(sum, 16);

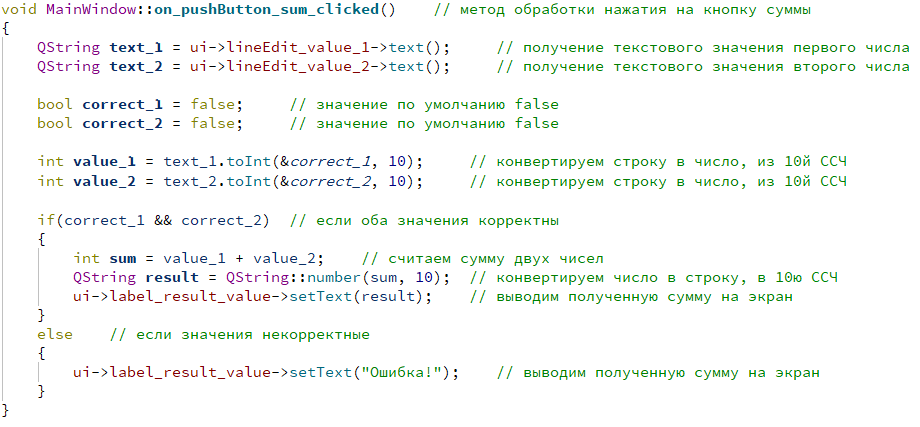
Ту же самую операцию можно выполнить, используя метод QString::setNum():

QString result;

result.setNum(sum, 16);

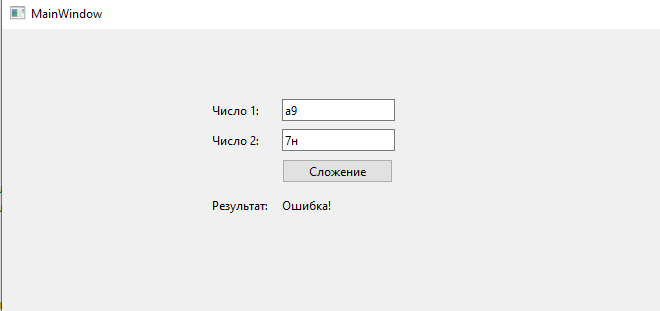
Допускается использование обоих вариантов.

Полный код выглядит следующим образом:



Полный код проверки корректности данных

Таким образом, при некорректных введённых данных, программа будет сообщать об ошибке:



Отображение ошибки