|  |  |
| --- | --- |
| 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 №8

## Двоичные потоки

Двоичные (бинарные) потоки предназначены для работы с двоичными данными, они представляют собой только последовательность 0 и 1 и никакой другой информации. В связи с этим, чтобы прочитать данные из бинарного файла, необходимо знать его структуру, т.е. какие данные, в какой последовательности в него записаны.

В данной практической работе будет рассмотрена работа с двоичными потоками на примере бинарных файлов.

Для работы с двоичными потоками в Qt используется класс QDataStream, для его подключения необходимо использовать директиву #include <QDataStream>.

Создание двоичного потока и запись данных производится аналогично с текстовым потоком:

QFile file("data.bin"); // создаем объект QFile

if(!file.open(QIODevice::WriteOnly)) // пытаемся открыть файл

{

return;

}

QDataStream ds(&file); // создание двоичного потока

QString str = "Hello World!";

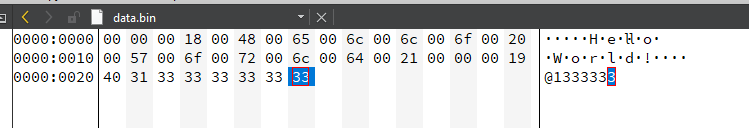
int v1 = 25;

double d1 = 17.2;

ds << str << v1 << d1; // запись в двоичный поток

file.close(); // закрываем файл

Записанный файл будет выглядеть таким образом:



Записанный бинарный файл

Чтение данных из бинарного файла производится аналогичным образом:

QFile file("data.bin"); // создаем объект QFile

if(!file.open(QIODevice::ReadOnly)) // пытаемся открыть файл

{

return;

}

QDataStream ds(&file); // создание двоичного потока

QString str;

int v1;

double d1;

ds >> str; // "Hello World!"

ds >> v1; // 25

ds >> d1; // 17.2

file.close(); // закрываем файл

Обратите внимание, что в двоичном потоке нет необходимости разделять данные пробелами – в двоичный поток записываются данные о размере блока данных, который был записан, по этому размеру поток ориентируется при чтении блока данных.

Таким образом производится чтение и запись данных в бинарные файлы с помощью бинарных потоков.