>>> Programação Orientada a Objetos (POO)

... Interface Gráfica

Prof: André de Freitas Smaira

[~]$ _

>>> Interface Gráfica (GUI)

* Uma aplicação de POO é construção de interfaces gráficas

>>> Qt5

```
#include <QApplication>
#include <QWidget>
#include <QPushButton>
#include <QVBoxLayout>
#include <QHBoxLayout>
#include <QPainter>
#include <QPen>
#include <QVector>
#include <QPoint>
```

>>> Qt5

```
class Canvas : public QWidget {
    Q_OBJECT

    int x, y;
    QVector<QPoint> pontos;

public:
    Canvas(QWidget *parent = nullptr) : QWidget(parent), x(width() / 2), y(height() / 2) {
        setMinimumSize(400, 300);
        pontos.append(QPoint(x, y));
    }

    void moveUp() { moveTo(x, y - 10); }
    void moveDown() { moveTo(x, y + 10); }
    void moveLeft() { moveTo(x - 10, y); }
    void moveRight() { moveTo(x + 10, y); }
protected:
    void paintEvent(QPaintEvent *event) override {
        QPainter painter(this);
        QPen pen(Qt::black, 2);
        painter.setPen(pen);

        for (int i = 1; i < pontos.size(); ++i)
            painter.drawLine(pontos[i - 1], pontos[i]);
    }
private:
    void moveTo(int newX, int newY) {
        x = newX;
        y = newY;
        pontos.append(QPoint(x, y));
        update();
    }
};
```

```
[~]$ _
```

>>> Qt5

```cpp
class TartarugaDesenhista : public QWidget {
    Q_OBJECT

    Canvas *canvas;

public:
    TartarugaDesenhista(QWidget *parent = nullptr) : QWidget(parent) {
        QVBoxLayout *mainLayout = new QVBoxLayout(this);

        canvas = new Canvas(this);
        mainLayout->addWidget(canvas);

        QHBoxLayout *buttonLayout = new QHBoxLayout;
        QPushButton *cimaBtn = new QPushButton("Cima", this);
        QPushButton *baixoBtn = new QPushButton("Baixo", this);
        QPushButton *esquerdaBtn = new QPushButton("Esquerda", this);
        QPushButton *direitaBtn = new QPushButton("Direita", this);

        buttonLayout->addWidget(cimaBtn);
        buttonLayout->addWidget(baixoBtn);
        buttonLayout->addWidget(esquerdaBtn);
        buttonLayout->addWidget(direitaBtn);

        mainLayout->addLayout(buttonLayout);

        connect(cimaBtn, &QPushButton::clicked, canvas, &Canvas::moveUp);
        connect(baixoBtn, &QPushButton::clicked, canvas, &Canvas::moveDown);
        connect(esquerdaBtn, &QPushButton::clicked, canvas, &Canvas::moveLeft);
        connect(direitaBtn, &QPushButton::clicked, canvas, &Canvas::moveRight);

        setWindowTitle("Tartaruga Desenhista");
    }
};
```

```
[~]$ _
```

>>> Qt5

```
int main(int argc, char *argv[]) {
    QApplication app(argc, argv);
    TartarugaDesenhista window;
    window.show();
    return app.exec();
}

#include "desenha.moc"
```

>>> Compila

Compilando com

```
qmake desenha.pro
make
```

>>> Qt5 - Outro exemplo

```cpp
#include <QApplication>
#include <QWidget>
#include <QVBoxLayout>
#include <QLabel>
#include <QLineEdit>
#include <QPushButton>
#include <QMessageBox>

class MeuPrograma : public QWidget {
    Q_OBJECT

public:
    MeuPrograma(QWidget *parent = nullptr);

private slots:
    void onButtonClicked();

private:
    QLineEdit *inputNome;
    QPushButton *botao;
};
```

>>> Qt5 - Outro exemplo

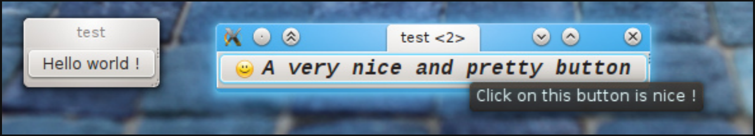
```
MeuPrograma::MeuPrograma(QWidget *parent)
    : QWidget(parent), inputNome(new QLineEdit(this)),
        botao(new QPushButton("Clique aqui", this)) {
    setWindowTitle("Exemplo de Qt5");
    setFixedSize(300, 150);

    QVBoxLayout *layout = new QVBoxLayout(this);

    QLabel *label = new QLabel("Digite seu nome e pressione o botão:", this);
    layout->addWidget(label);

    inputNome->setPlaceholderText("Seu nome");
    layout->addWidget(inputNome);

    connect(botao, &QPushButton::clicked, this, &MeuPrograma::onButtonClicked);
    layout->addWidget(botao);

    setLayout(layout);
}
```

>>> Qt5 - Outro exemplo

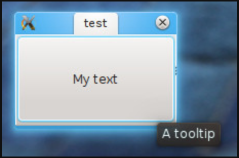
```
void MeuPrograma::onButtonClicked() {
    QString nome = inputNome->text();

    QMessageBox::information(this, "Saudação", "Olá, " + nome +
    "!\nObrigado por usar este programa.");
}

int main(int argc, char *argv[]) {
    QApplication app(argc, argv);

    MeuPrograma janela;
    janela.show();

    return app.exec();
}

#include "main.moc"
```

>>> Qt5 - Exemplos

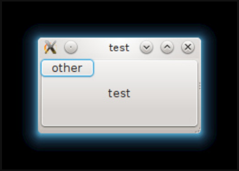
https://wiki.qt.io/Qt_for_Beginners



Click on this button is nice !

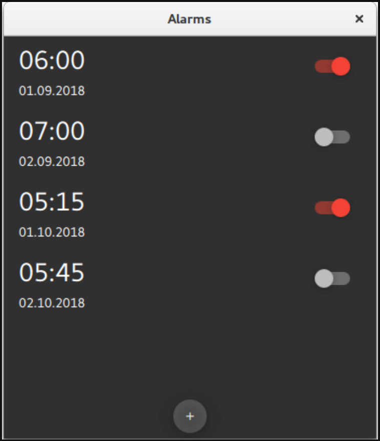


A tooltip



>>> Qt5 - Exemplos

https://doc.qt.io/qt-5/qtexamples.html



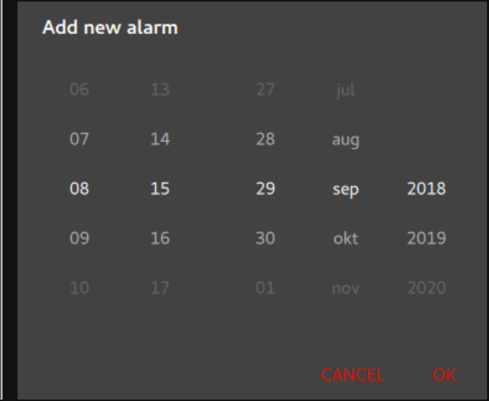
Alarms ×

06:00
01.09.2018

07:00
02.09.2018

05:15
01.10.2018

05:45
02.10.2018

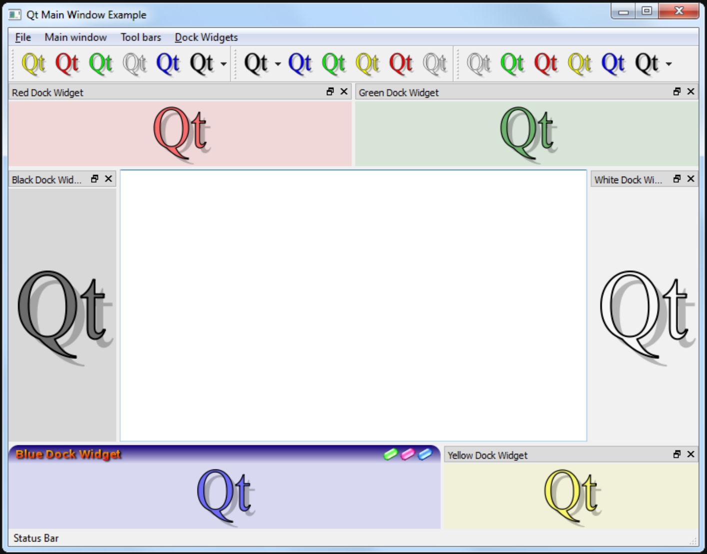


Add new alarm

```
06   13   27   jul
07   14   28   aug
08   15   29   sep   2018
09   16   30   okt   2019
10   17   01   nov   2020
```

[~]$ _

>>> Qt5 - Exemplo


Qt Main Window Example
File
Main window
Tool bars
Dock Widgets
Red Dock Widget
Green Dock Widget
Black Dock Wid...
White Dock Wi...
Blue Dock Widget
Yellow Dock Widget
Status Bar

[~]$ _

>>> GTKm

```cpp
#include <gtkmm.h>

class MeuPrograma : public Gtk::Window {
public:
    MeuPrograma();

private:
    void on_button_clicked();

    Gtk::Box vbox;
    Gtk::Label label;
    Gtk::Entry entry;
    Gtk::Button button;
};
```

>>> GTKm

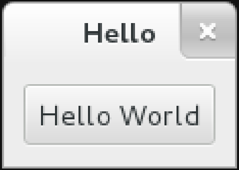
```
MeuPrograma::MeuPrograma()
    : vbox(Gtk::ORIENTATION_VERTICAL), // Caixa vertical
      button("Clique aqui")             // Texto do botão
{
    set_title("Exemplo de GTKmm");
    set_border_width(10);
    set_default_size(300, 150);

    add(vbox);

    label.set_text("Digite seu nome e pressione o botão:");
    vbox.pack_start(label, Gtk::PACK_SHRINK);

    entry.set_placeholder_text("Seu nome");
    vbox.pack_start(entry, Gtk::PACK_SHRINK);

    button.signal_clicked().connect(sigc::mem_fun(*this,
                            &MeuPrograma::on_button_clicked));
    vbox.pack_start(button, Gtk::PACK_SHRINK);

    show_all_children();
}
```

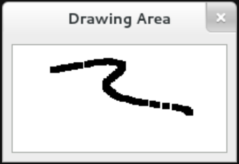
>>> GTKm

```cpp
void MeuPrograma::on_button_clicked() {
    Glib::ustring nome = entry.get_text();

    Gtk::MessageDialog dialog(*this, "Olá, " + nome + "!");
    dialog.set_secondary_text("Obrigado por usar este programa.");
    dialog.run();  // Exibe a caixa de diálogo
}

int main(int argc, char *argv[]) {
    auto app = Gtk::Application::create(argc, argv, "org.gtkmm.exemplo");
    MeuPrograma janela;
    return app->run(janela);
}
```

>>> GTKM - Exemplos

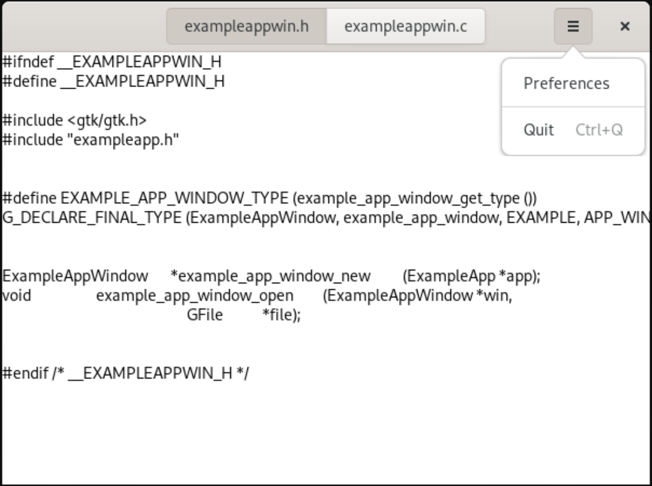
https://docs.gtk.org/gtk4/getting_started.html





>>> GTKM - Exemplos

https://docs.gtk.org/gtk4/getting_started.html



```
#ifndef __EXAMPLEAPPWIN_H
#define __EXAMPLEAPPWIN_H

#include <gtk/gtk.h>
#include "exampleapp.h"


#define EXAMPLE_APP_WINDOW_TYPE (example_app_window_get_type ())
G_DECLARE_FINAL_TYPE (ExampleAppWindow, example_app_window, EXAMPLE, APP_WIN


ExampleAppWindow     *example_app_window_new        (ExampleApp *app);
void                 example_app_window_open        (ExampleAppWindow *win,
                                                     GFile          *file);


#endif /* __EXAMPLEAPPWIN_H */
```

[~]$ _

>>> FLTK

```
#include <FL/Fl.H>
#include <FL/Fl_Window.H>
#include <FL/Fl_Box.H>
#include <FL/Fl_Input.H>
#include <FL/Fl_Button.H>
#include <FL/fl_ask.H>

class MeuPrograma {
public:
    MeuPrograma();

private:
    static void onButtonClicked(Fl_Widget *widget, void *data);

    Fl_Window *window;
    Fl_Input *inputNome;
    Fl_Button *botao;
};
```

>>> FLTK

```cpp
MeuPrograma::MeuPrograma() {
    window = new Fl_Window(300, 150, "Exemplo de FLTK");

    Fl_Box *label = new Fl_Box(20, 20, 260, 30, "Digite seu nome e pressione o botão:");
    label->align(FL_ALIGN_INSIDE | FL_ALIGN_LEFT);

    inputNome = new Fl_Input(20, 60, 260, 25, "Seu nome:");
    inputNome->align(FL_ALIGN_TOP);

    botao = new Fl_Button(100, 100, 100, 30, "Clique aqui");
    botao->callback(onButtonClicked, (void*)this);

    window->end();
    window->show();
}
```

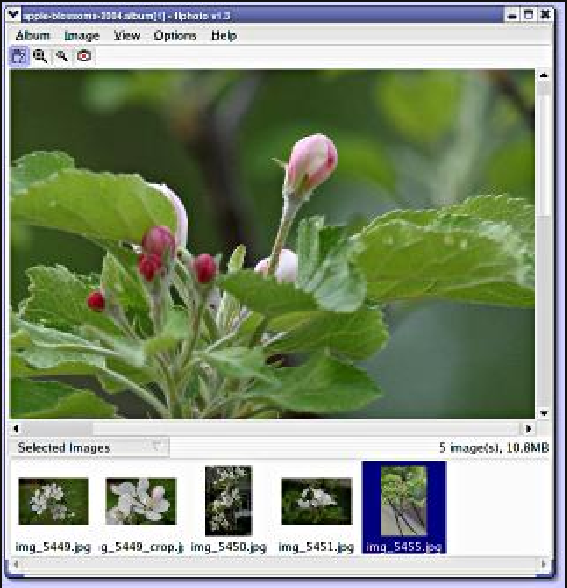
[~]$ _

>>> FLTK

```
void MeuPrograma::onButtonClicked(Fl_Widget *widget, void *data) {
    MeuPrograma *program = (MeuPrograma*)data;
    const char *nome = program->inputNome->value();

    fl_message("Olá, %s!\nObrigado por usar este programa.", nome);
}

int main(int argc, char **argv) {
    MeuPrograma programa;
    return Fl::run();
}
```

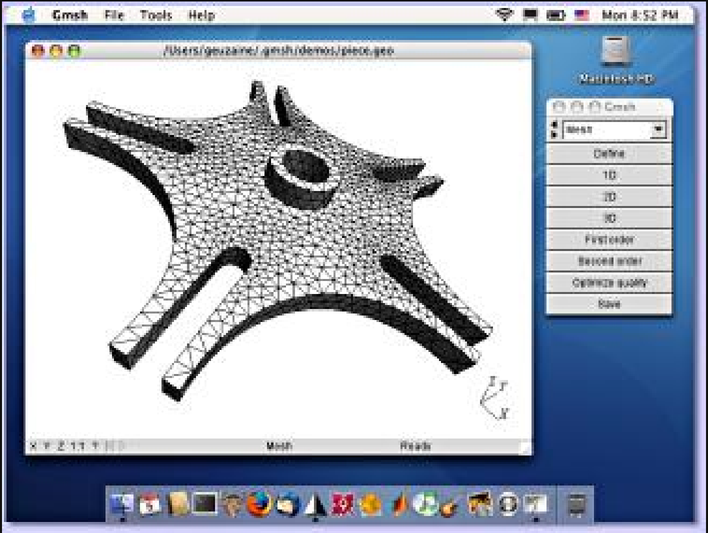
>>> FLTK - Exemplos

https://www.fltk.org/applications/shots.php



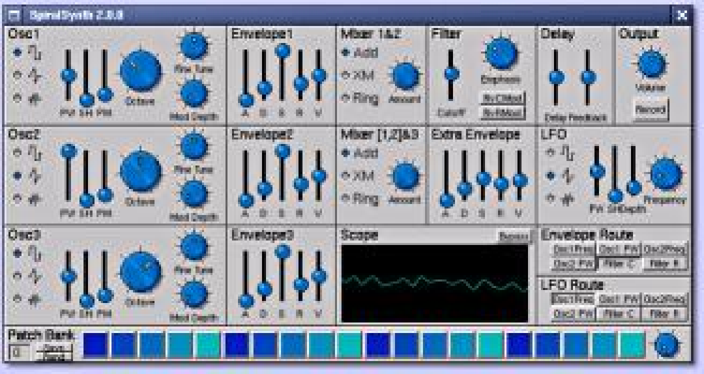
```
[~]$ _
```

>>> FLTK - Exemplos



[~]$ _

>>> FLTK - Exemplos



>>> SFML

```
#include <SFML/Graphics.hpp>
#include <SFML/Window.hpp>
#include <iostream>
#include <string>

class MeuPrograma {
public:
    MeuPrograma();
    void run();

private:
    sf::RenderWindow window;
    sf::Font font;
    sf::Text label;
    sf::RectangleShape inputBox;
    sf::RectangleShape button;
    sf::Text buttonText;
    sf::Text outputText;
    std::wstring inputString;

    void handleEvents();
    void render();
};
[~]$ _
```

>>> DFML

```cpp
MeuPrograma::MeuPrograma()
    : window(sf::VideoMode(400, 200), "Exemplo de SFML") {
    if (!font.loadFromFile("/usr/share/fonts/truetype/dejavu/DejaVuSans.ttf")) {
        std::cerr << "Erro ao carregar a fonte!" << std::endl;
    }

    label.setFont(font);
    label.setString("Digite seu nome:");
    label.setCharacterSize(20);
    label.setFillColor(sf::Color::White);
    label.setPosition(10, 10);

    inputBox.setSize(sf::Vector2f(380, 30));
    inputBox.setFillColor(sf::Color(50, 50, 50));
    inputBox.setPosition(10, 50);

    button.setSize(sf::Vector2f(150, 30));
    button.setFillColor(sf::Color::Red);
    button.setPosition(150, 100);

    buttonText.setFont(font);
    buttonText.setString("Clique aqui");
    buttonText.setCharacterSize(20);
    buttonText.setFillColor(sf::Color::White);
    buttonText.setPosition(160, 103);

    outputText.setFont(font);
    outputText.setCharacterSize(20);
    outputText.setFillColor(sf::Color::White);
    outputText.setPosition(10, 150);
}
```

>>> SFML

```cpp
void MeuPrograma::handleEvents() {
    sf::Event event;
    while (window.pollEvent(event)) {
        if (event.type == sf::Event::Closed) {
            window.close();
        } else if (event.type == sf::Event::MouseButtonPressed) {
            if (event.mouseButton.button == sf::Mouse::Left) {
                if (button.getGlobalBounds().contains(event.mouseButton.x, event.mouseButton.y)) {
                    outputText.setString(L"Olá, " + inputString + L"!");
                }
            }
        } else if (event.type == sf::Event::TextEntered) {
            // Permite digitar na caixa de entrada
            if (event.text.unicode == '\b') { // Backspace
                if (!inputString.empty()) {
                    inputString.pop_back();
                }
            } else if (event.text.unicode < 128) { // Verifica se o caractere é ASCII
                inputString += static_cast<wchar_t>(event.text.unicode);
            }
        }
    }
}
```

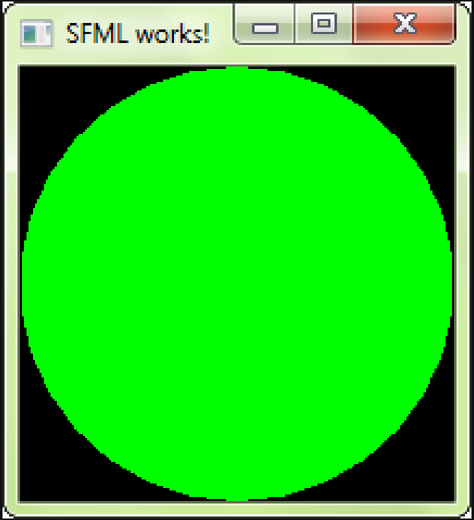
>>> SFML

```cpp
void MeuPrograma::render() {
    window.clear(sf::Color::Black);

    // Desenha o rótulo e a caixa de entrada
    window.draw(label);
    window.draw(inputBox);

    sf::Text inputText;
    inputText.setFont(font);
    inputText.setString(inputString); // Texto que o usuário digitou
    inputText.setCharacterSize(20);
    inputText.setFillColor(sf::Color::White);
    inputText.setPosition(inputBox.getPosition().x + 5, inputBox.getPosition().y + 5); // Pequena margem

    window.draw(inputText); // Desenha o texto digitado

    window.draw(button);
    window.draw(buttonText);

    window.draw(outputText);

    window.display();
}
```

>>> SFML

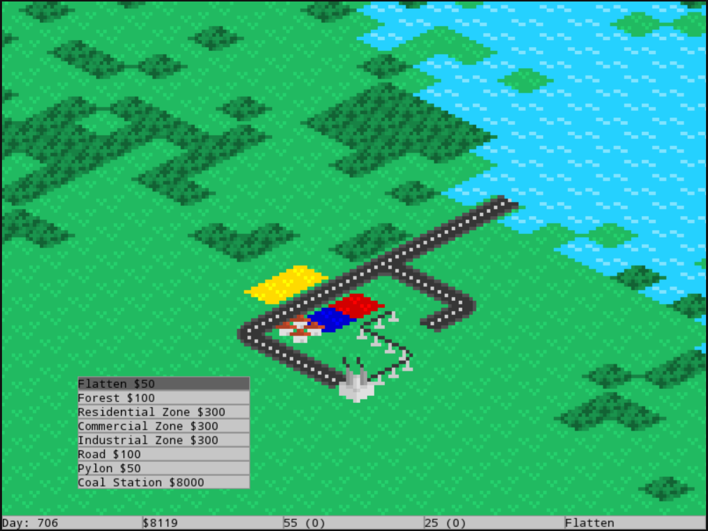
```cpp
void MeuPrograma::run() {
    while (window.isOpen()) {
        handleEvents();
        render();
    }
}

int main() {
    MeuPrograma programa;
    programa.run();
    return 0;
}
```

>>> SFML - Exemplo



```
[~]$ _
```

>>> SFML - Exemplo



[~]$ _

>>> wxWidgets

```cpp
#include <wx/wx.h>

class MeuApp : public wxApp {
public:
    virtual bool OnInit();
};

class MeuFrame : public wxFrame {
public:
    MeuFrame(const wxString& title);

private:
    void OnButtonClicked(wxCommandEvent& event);

    wxTextCtrl* inputNome;
    wxButton* botao;

    wxDECLARE_EVENT_TABLE();
};
```

[~]$ _

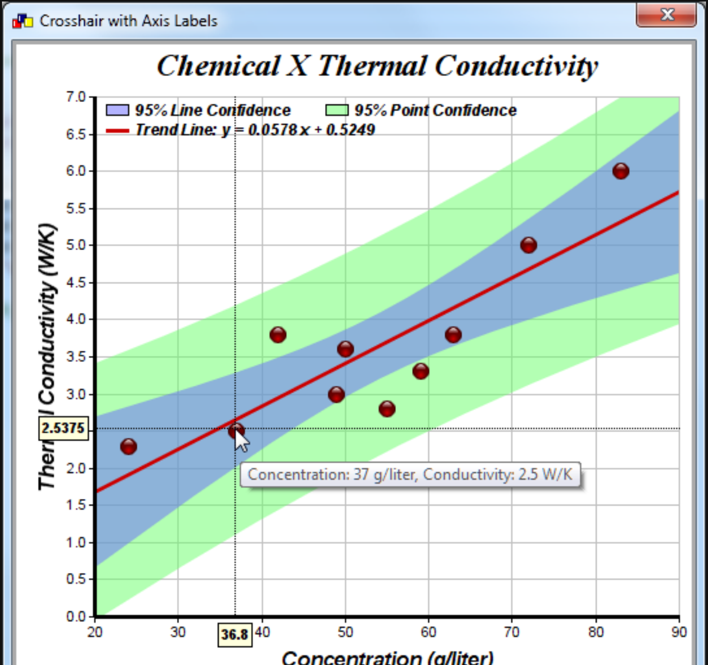
>>> wxWidgets

```cpp
wxBEGIN_EVENT_TABLE(MeuFrame, wxFrame)
    EVT_BUTTON(wxID_ANY, MeuFrame::OnButtonClicked)
wxEND_EVENT_TABLE()

wxIMPLEMENT_APP(MeuApp);

bool MeuApp::OnInit() {
    MeuFrame* frame = new MeuFrame("Exemplo wxWidgets");
    frame->Show(true);
    return true;
}

MeuFrame::MeuFrame(const wxString& title)
    : wxFrame(nullptr, wxID_ANY, title, wxDefaultPosition, wxSize(300, 150)) {
    wxPanel* panel = new wxPanel(this, wxID_ANY);

    wxBoxSizer* sizer = new wxBoxSizer(wxVERTICAL);

    inputNome = new wxTextCtrl(panel, wxID_ANY, "", wxDefaultPosition, wxSize(280, 30));
    sizer->Add(inputNome, 0, wxALL | wxCENTER, 5);

    botao = new wxButton(panel, wxID_ANY, "Clique aqui", wxDefaultPosition, wxSize(280, 30));
    sizer->Add(botao, 0, wxALL | wxCENTER, 5);

    panel->SetSizer(sizer);
}
```

>>> wxWidgets

```cpp
void MeuFrame::OnButtonClicked(wxCommandEvent& event) {
    wxString nome = inputNome->GetValue();
    wxString mensagem = "Olá, " + nome + "!";

    // Exibe a mensagem em uma caixa de diálogo
    wxMessageDialog dialog(this, mensagem, "Saudação", wxOK | wxICON_INFORMATION);
    dialog.ShowModal();  // Exibe a caixa de diálogo
}
```

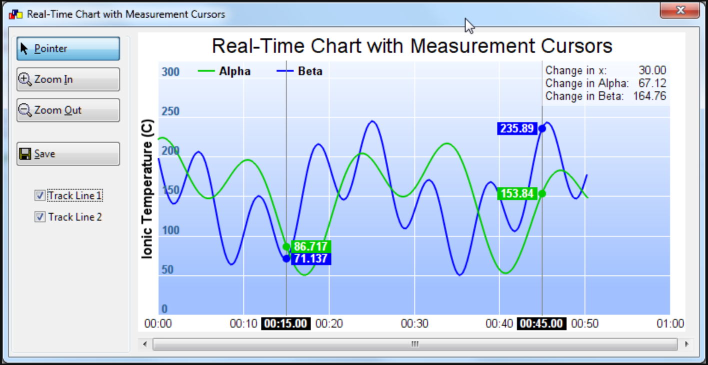
>>> wxWidgets- Exemplos



Concentration: 37 g/liter, Conductivity: 2.5 W/K

```
[~]$ _
```

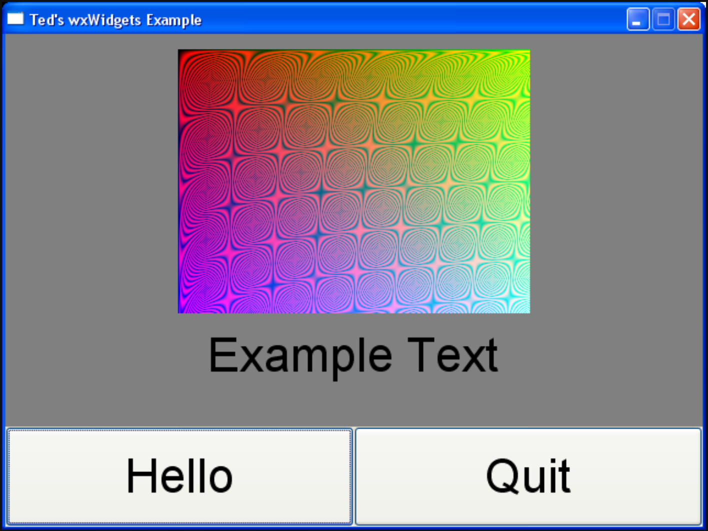
>>> wxWidgets- Exemplos



Change in x: 30.00
Change in Alpha: 67.12
Change in Beta: 164.76

```
[~]$ _
```

>>> wxWidgets- Exemplos



[~]$ _

>>> Comparação

Biblioteca	Tipo	Plataformas	Dificuldade	Uso
Qt5	GUI rica	Multiplataforma	Alta	Aplicações avançadas e multiplataforma
GTKmm	GUI tradicional	Linux, Windows, macOS	Média	Aplicações GNOME e Linux
FLTK	GUI leve	Multiplataforma	Baixa	Aplicações simples
SFML	Multimídia, 2D	Multiplataforma	Baixa	Jogos, gráficos
wxWidgets	GUI nativa	Multiplataforma	Média	Aplicações nativas multiplataforma