

ECE 462

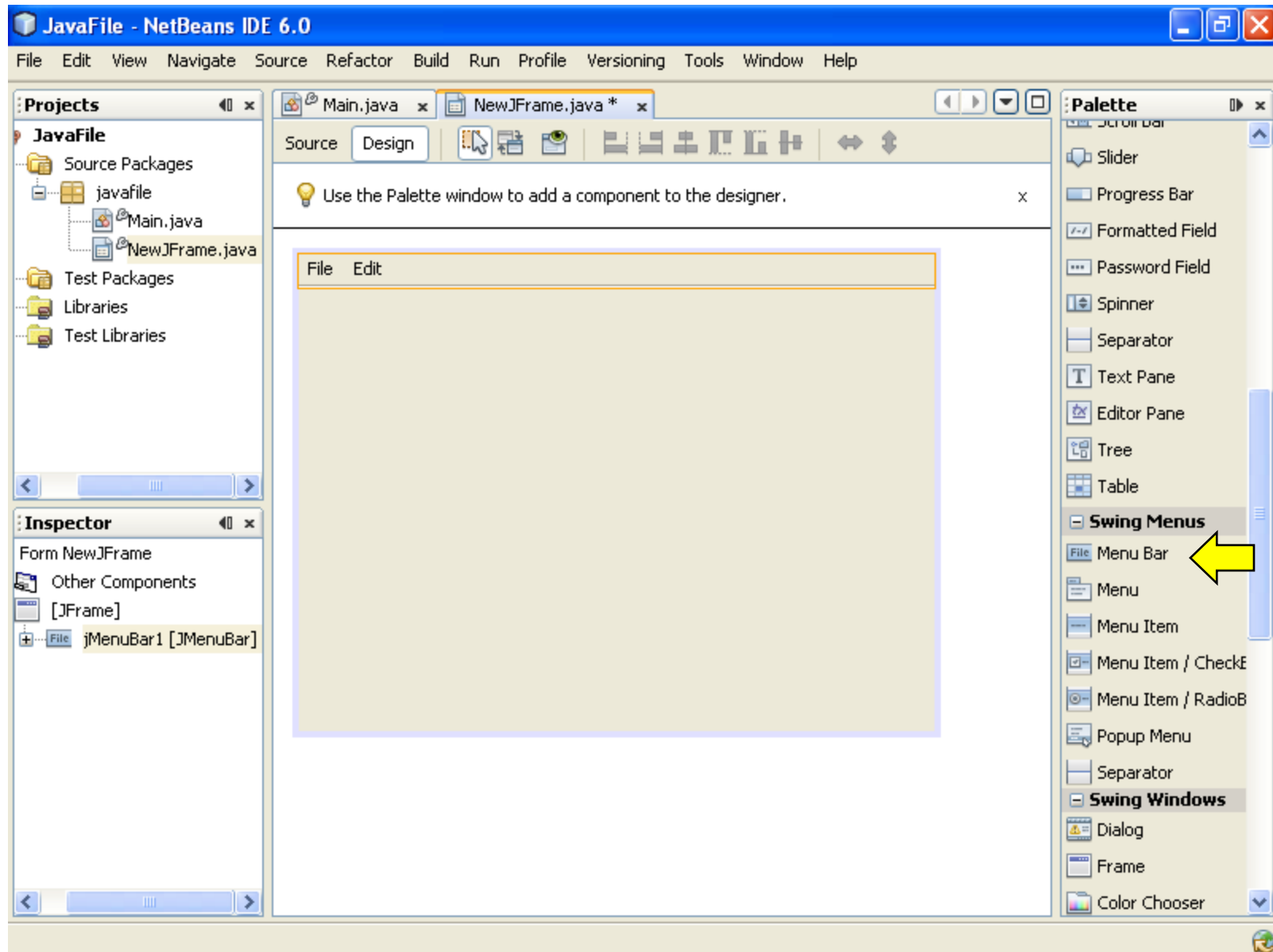
Object-Oriented Programming

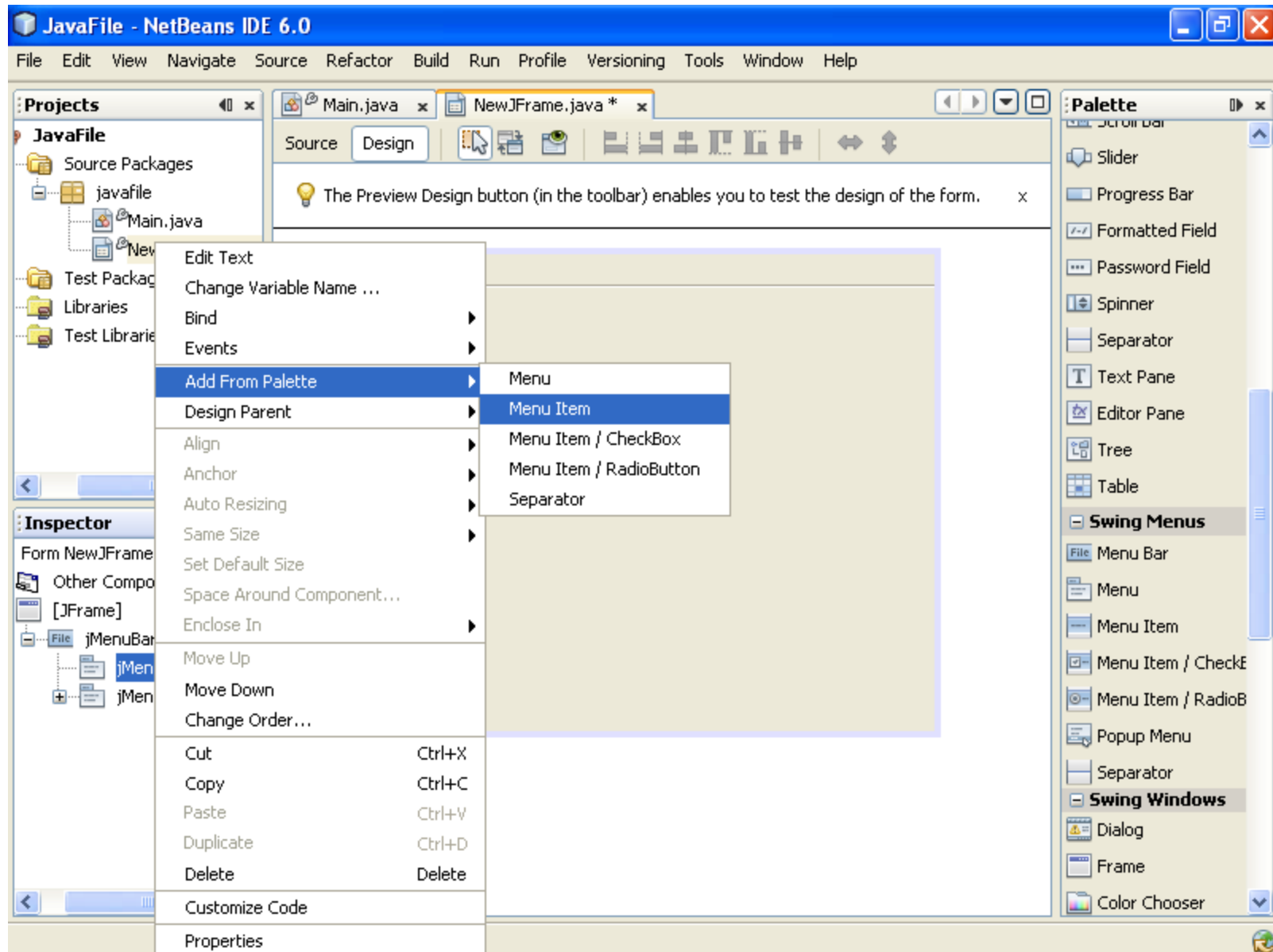
using C++ and Java

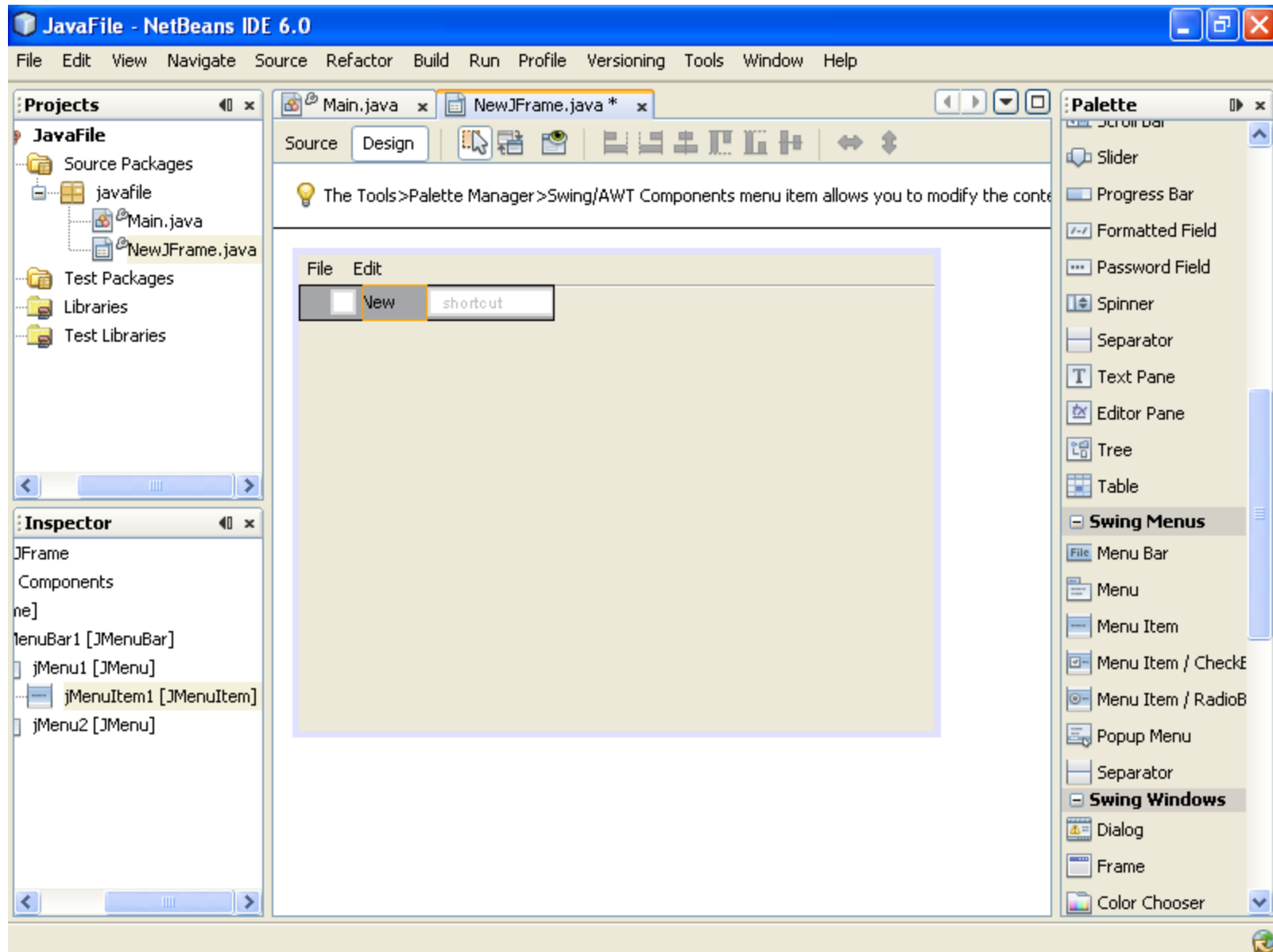
File Read and Write

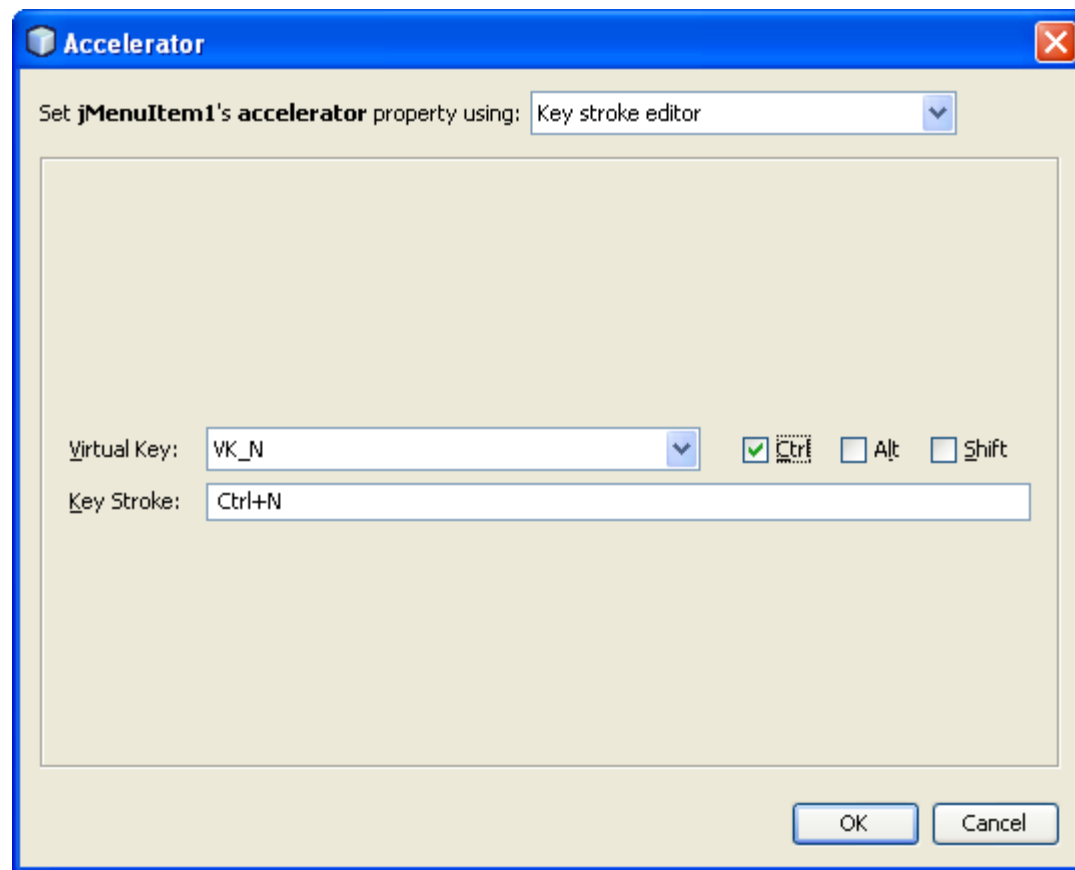
Yung-Hsiang Lu
yunlu@purdue.edu

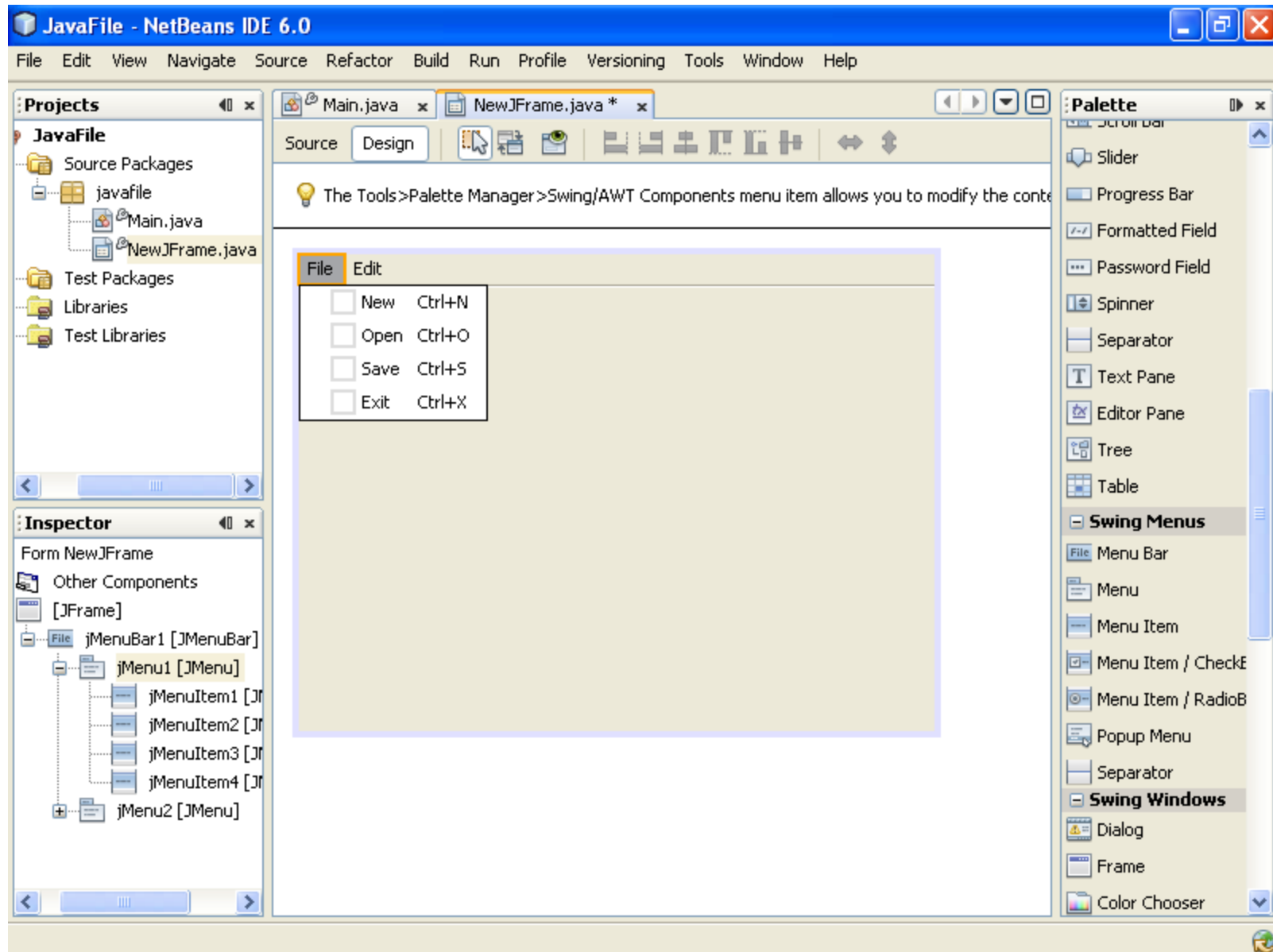
Save and Open File in Java

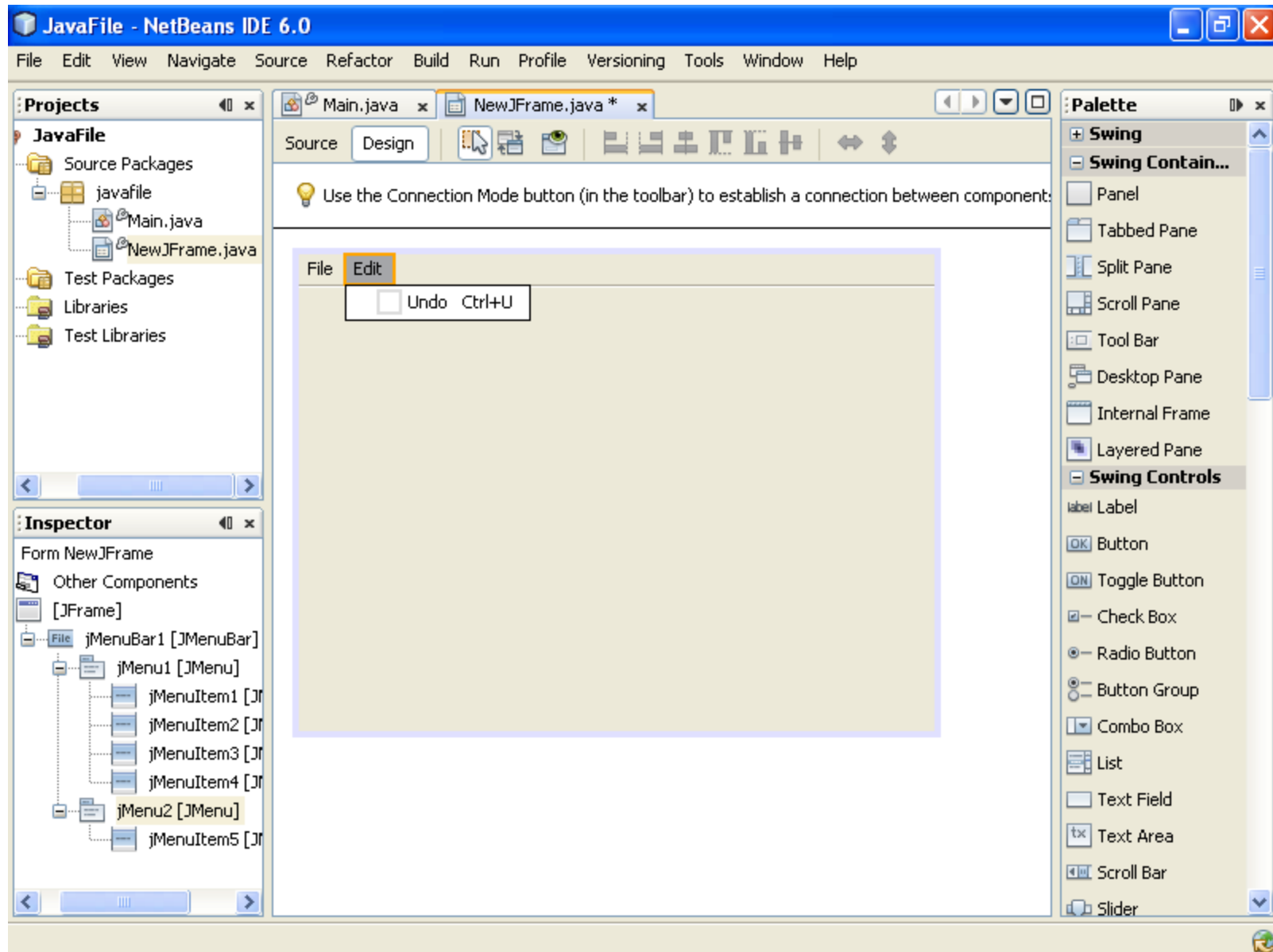


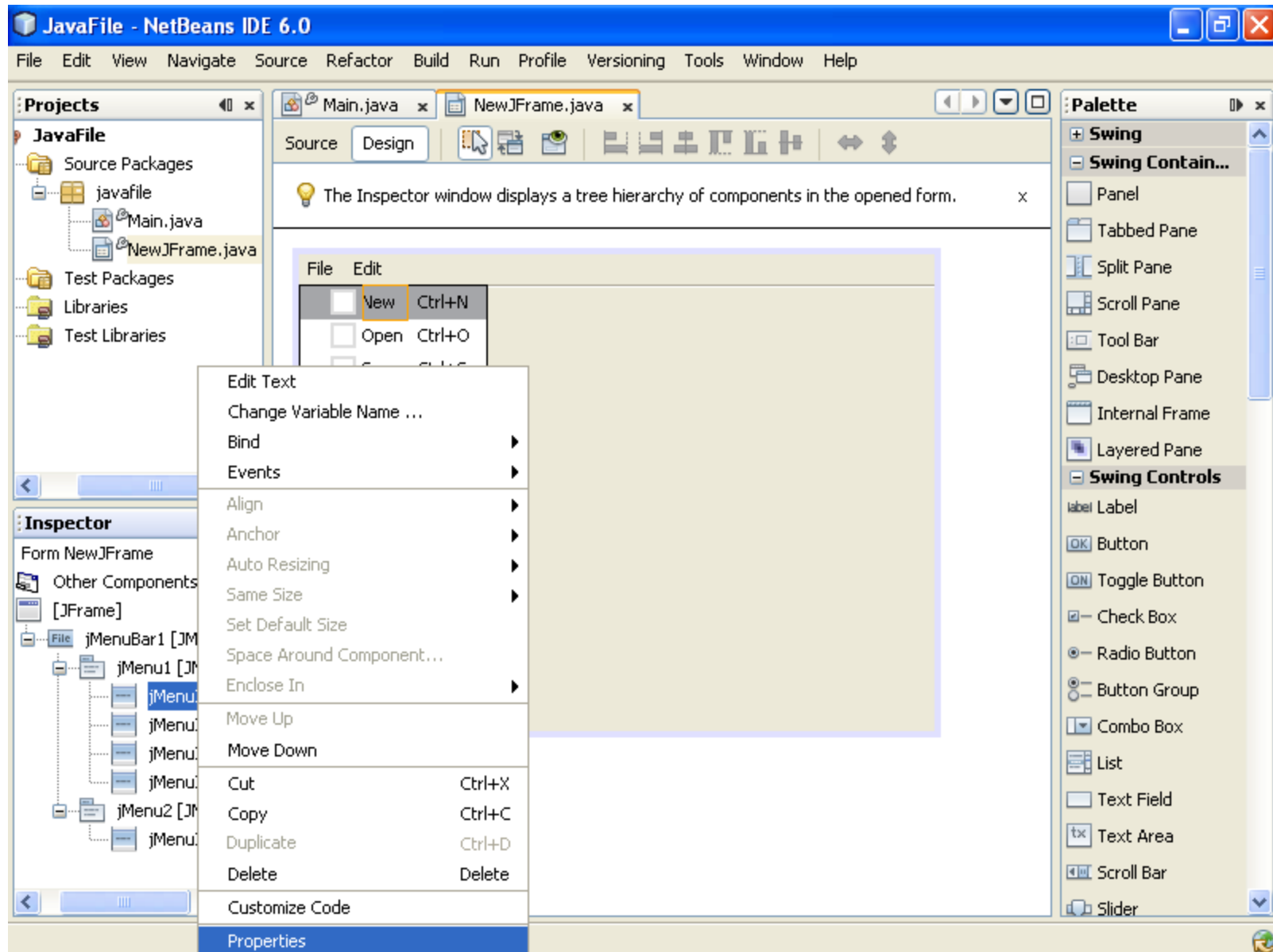


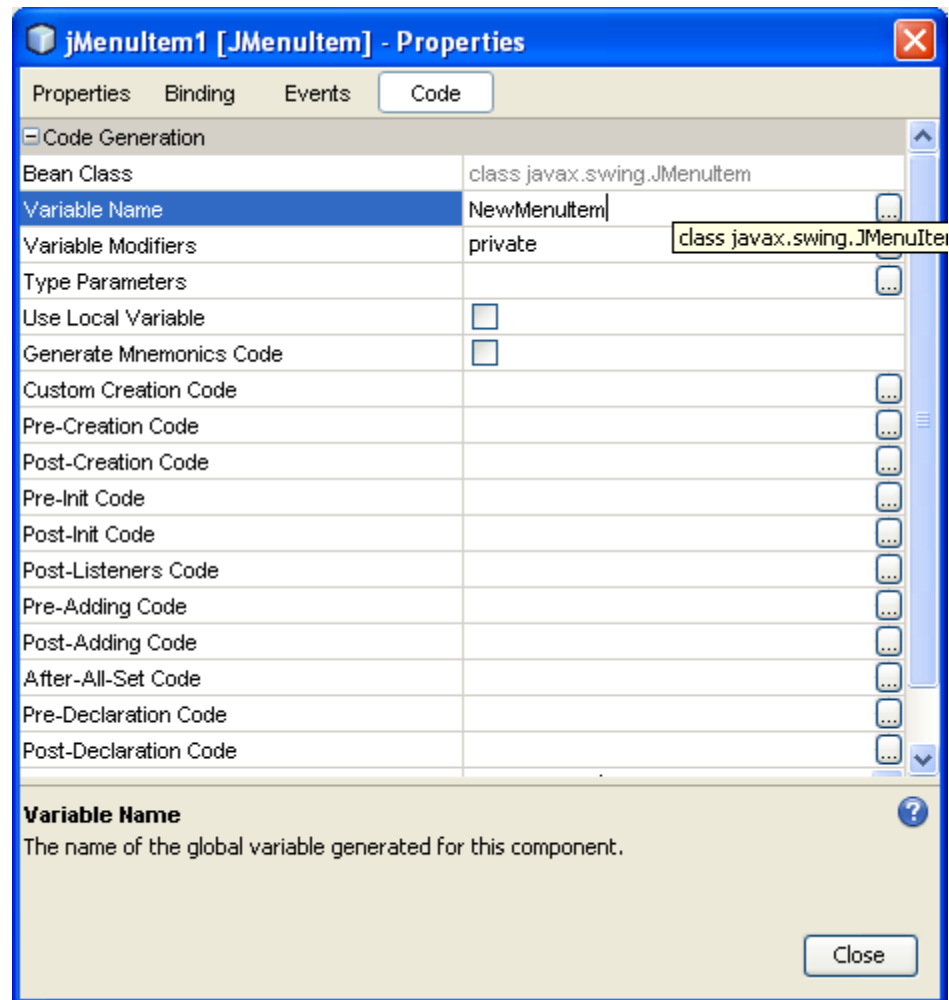


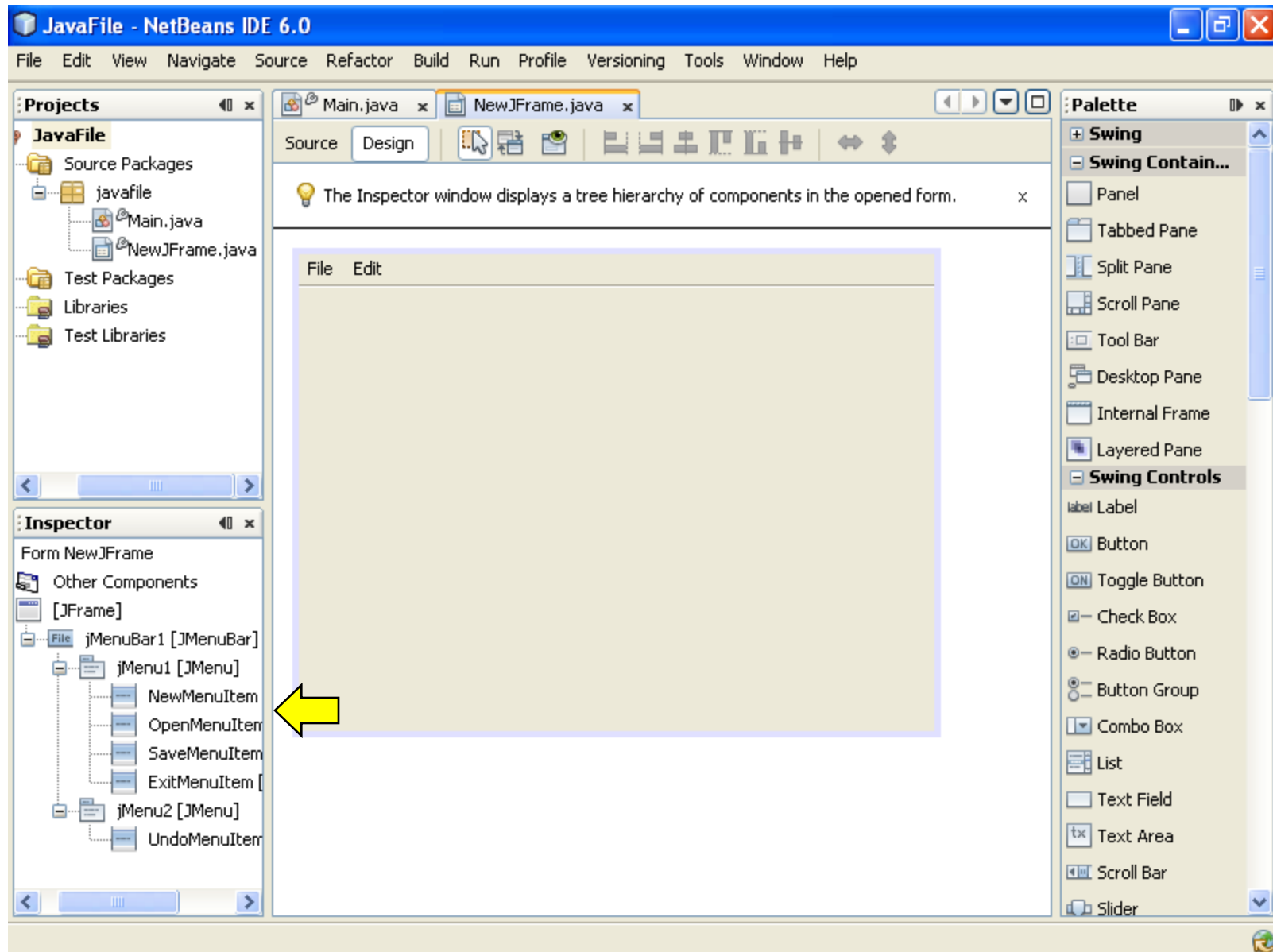


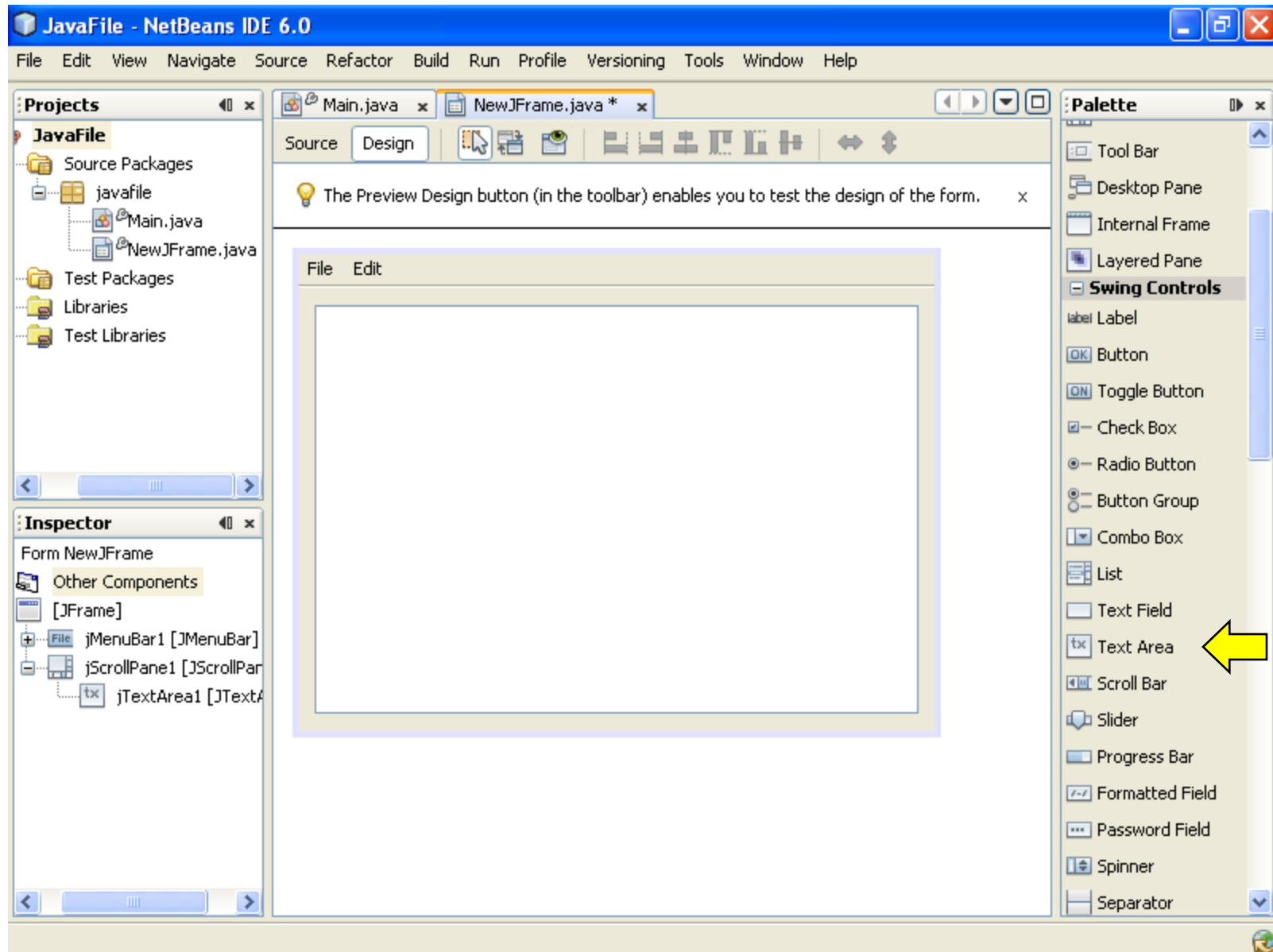


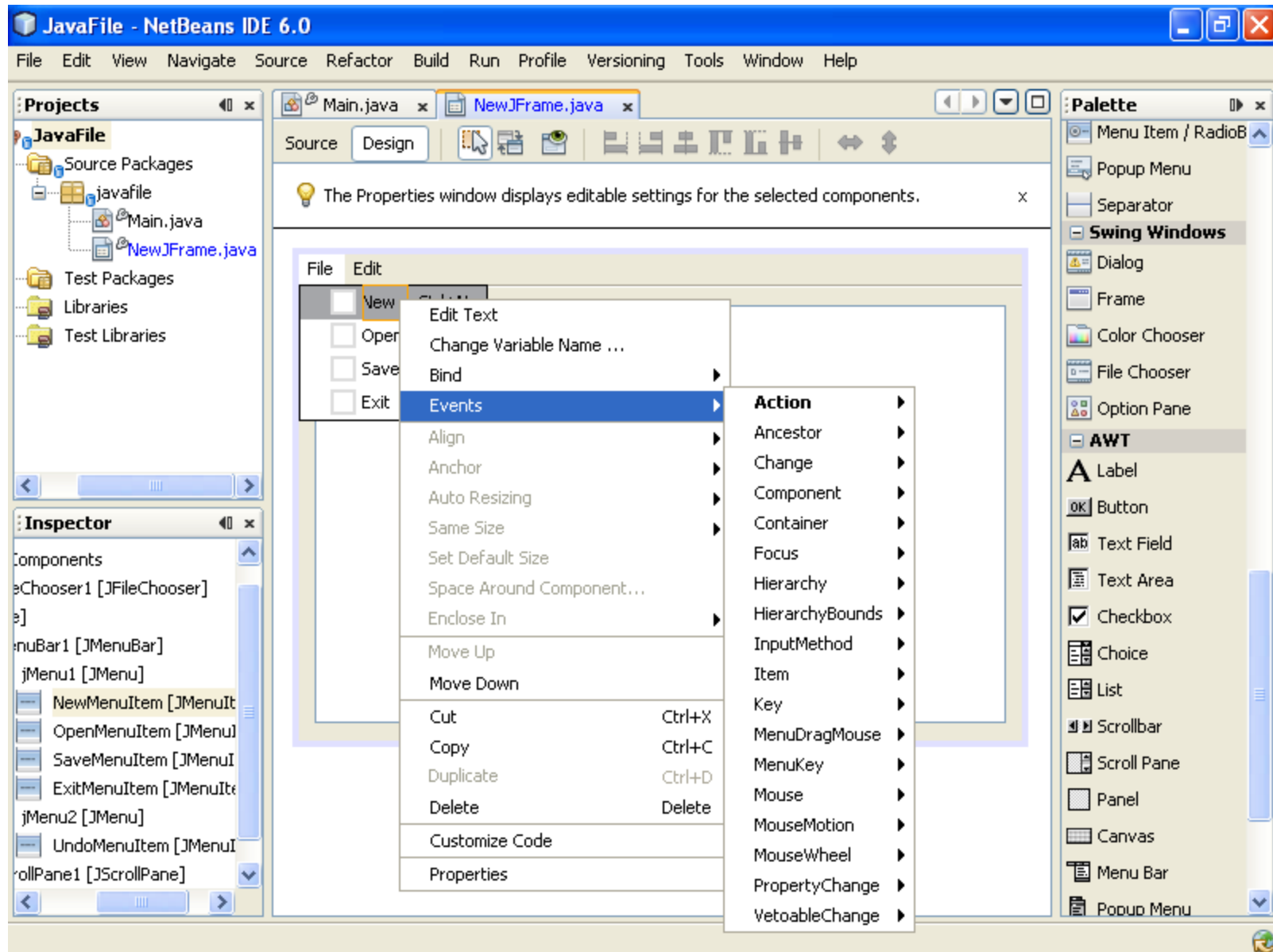


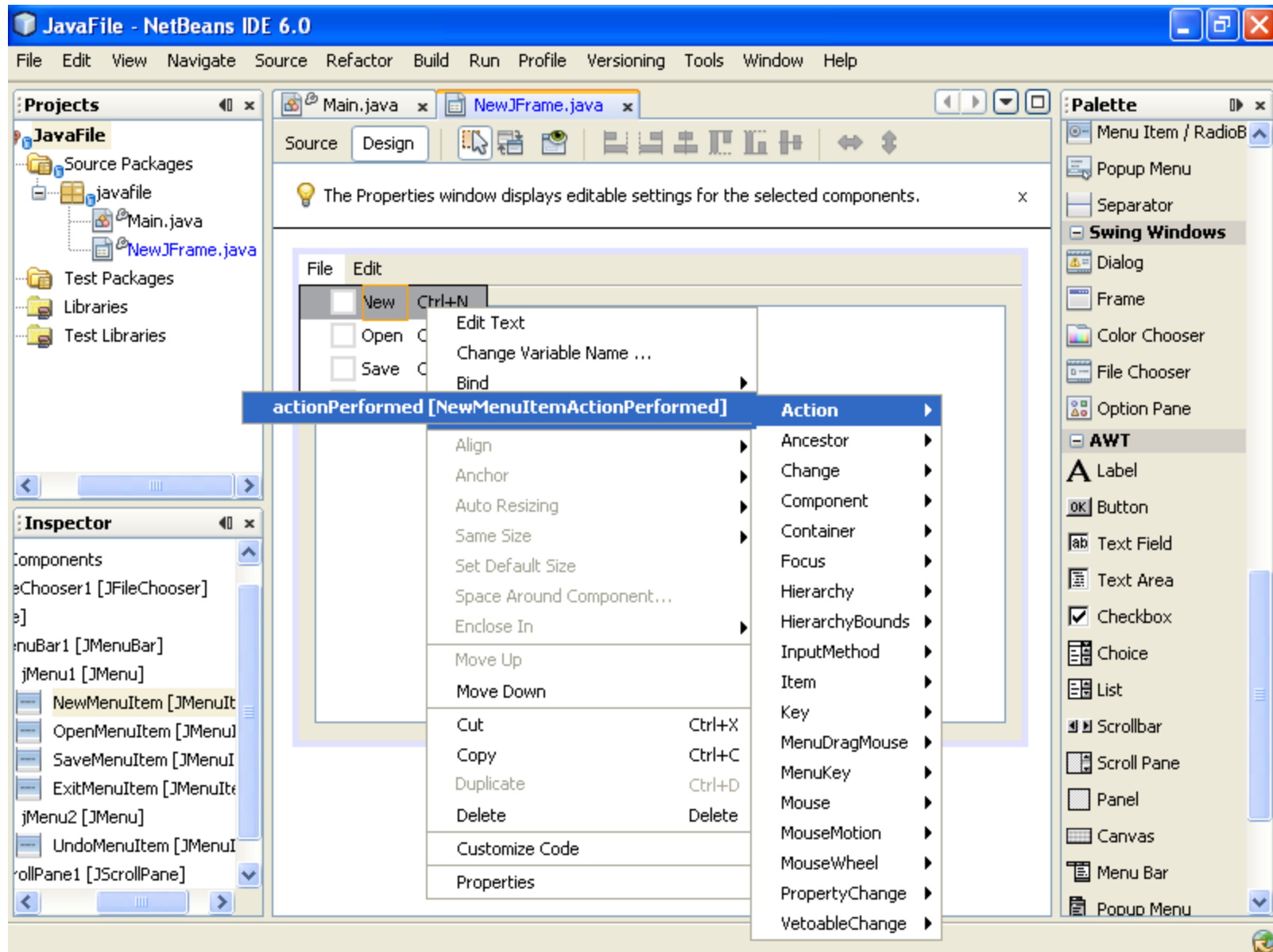












JFileChooser (Java Platform SE 6) - Mozilla Firefox

File Edit View History Bookmarks Yahoo! Tools Help

http://java.sun.com/javase/6/docs/api/javax/swing/JFileChooser.html

Overview Package **Class** Use Tree Deprecated Index Help

PREV CLASS NEXT CLASS FRAMES NO FRAMES All Classes

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Java™ Platform
Standard Ed. 6

javax.swing

Class JFileChooser

[java.lang.Object](#)

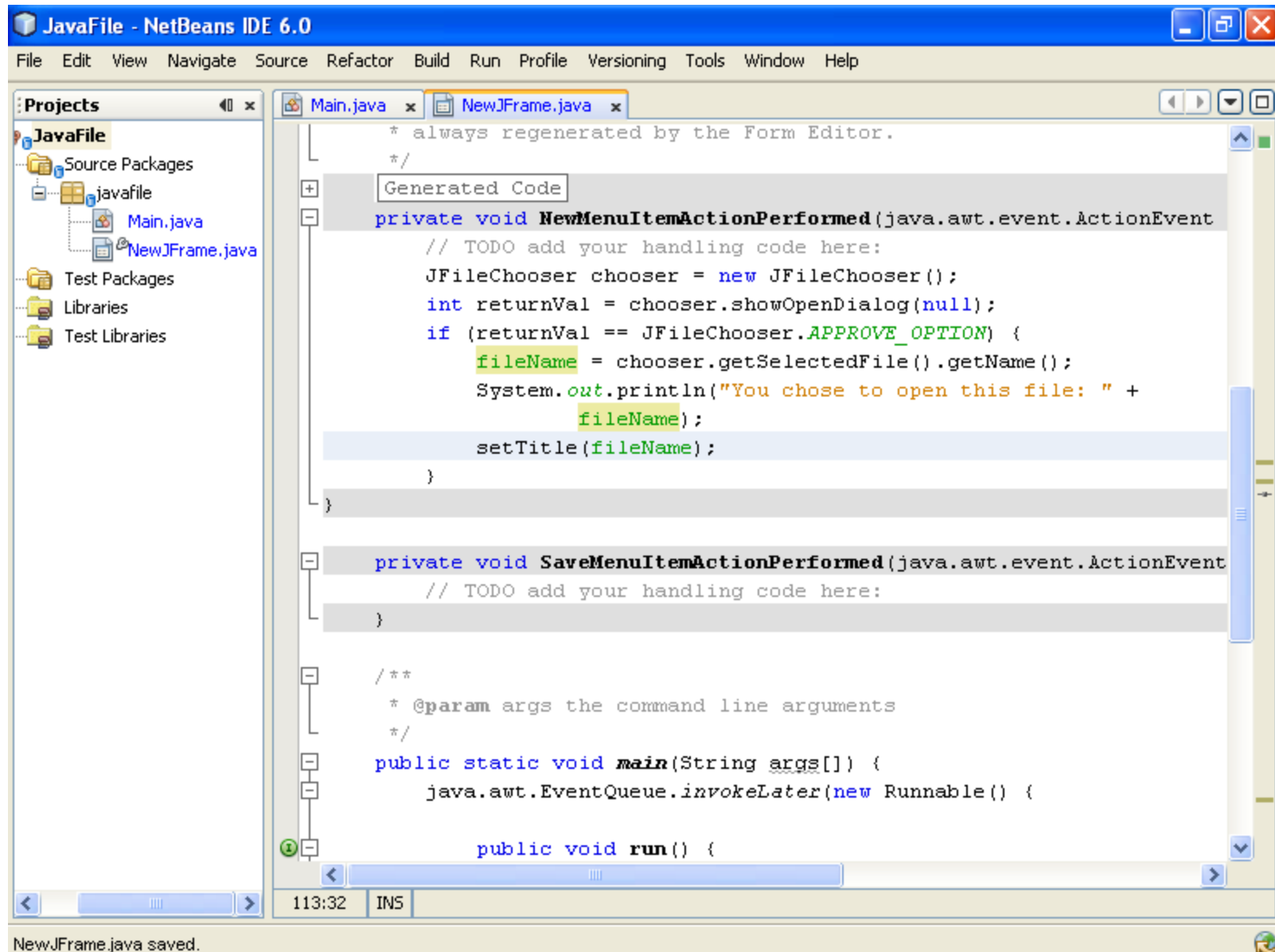
- [java.awt.Component](#)
 - [java.awt.Container](#)
 - [javax.swing.JComponent](#)
 - javax.swing.JFileChooser**

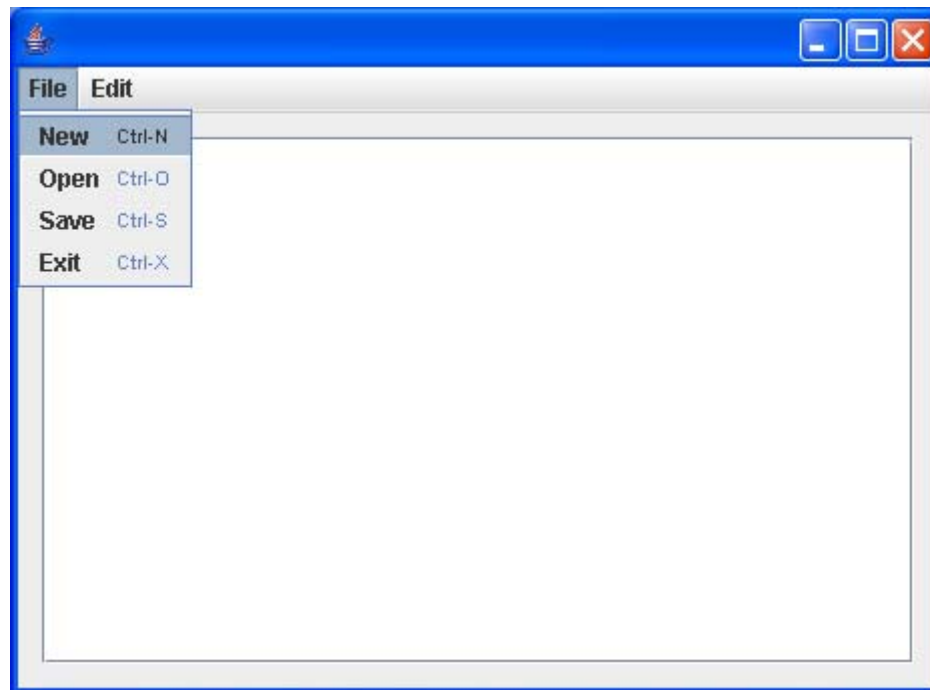
All Implemented Interfaces:
[ImageObserver](#), [MenuContainer](#), [Serializable](#), [Accessible](#)

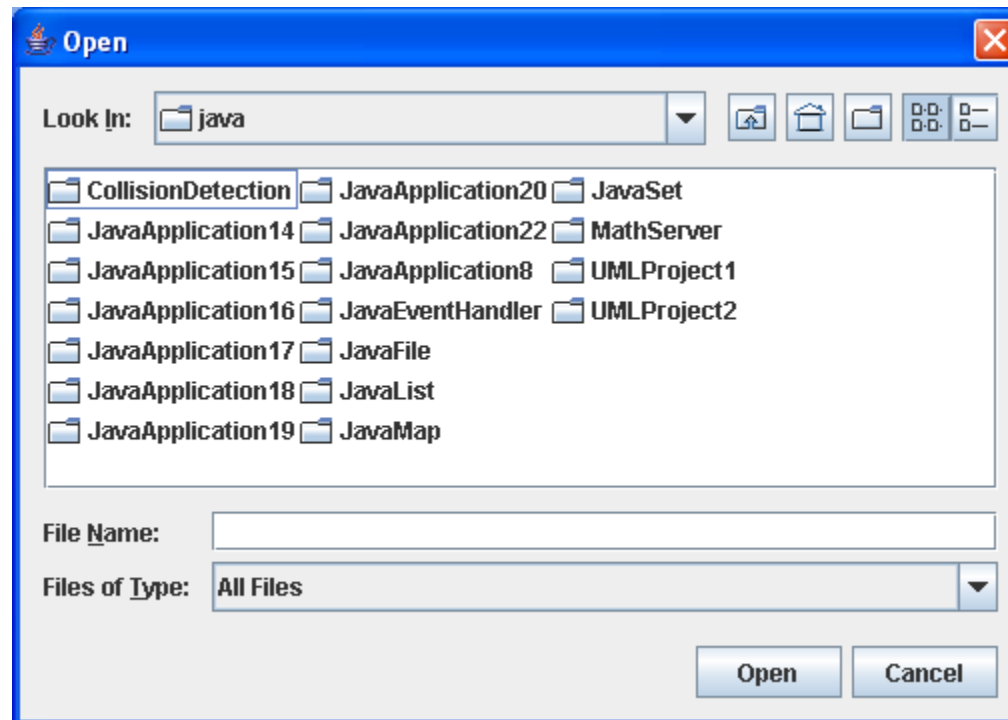
```
public class JFileChooser
extends JComponent
implements Accessible
```

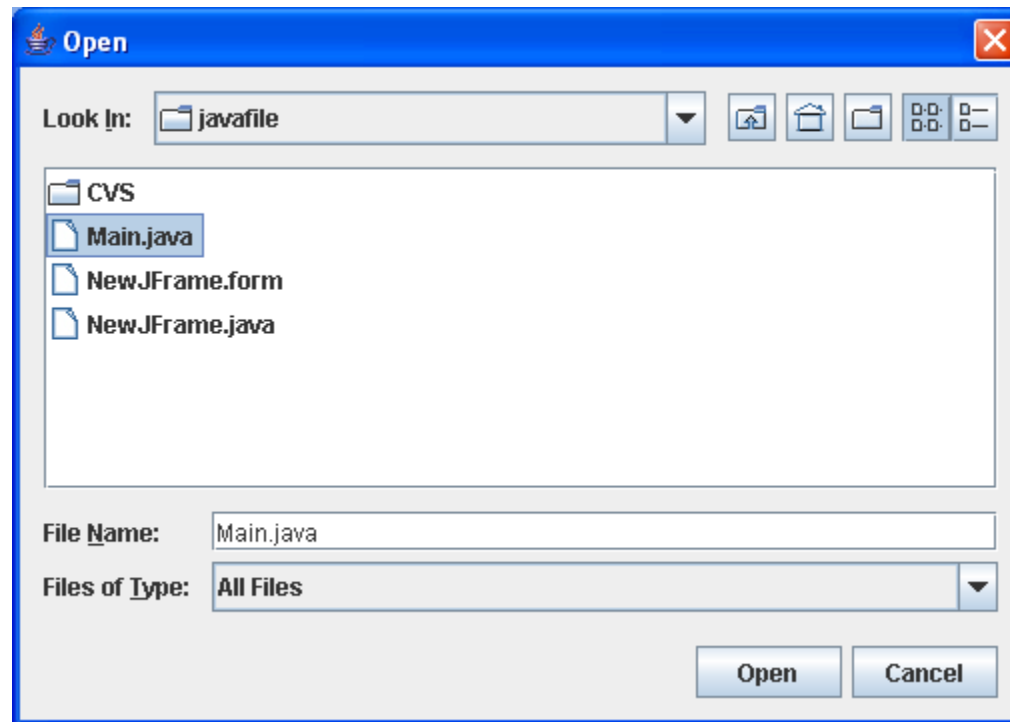
JFileChooser provides a simple mechanism for the user to choose a file. For information about using JFileChooser, see [How to Use File Choosers](#), a section in *The Java Tutorial*.

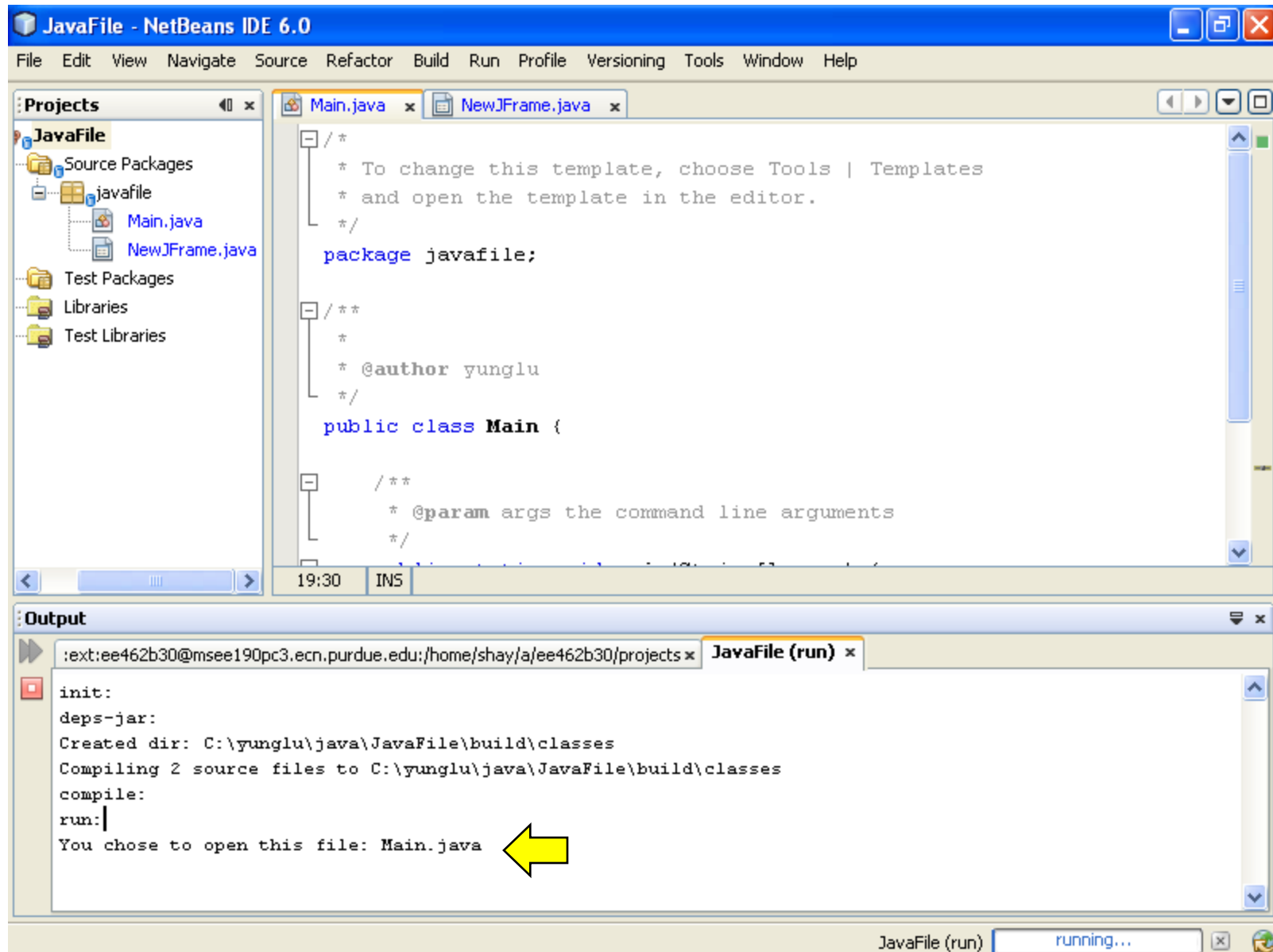
Done

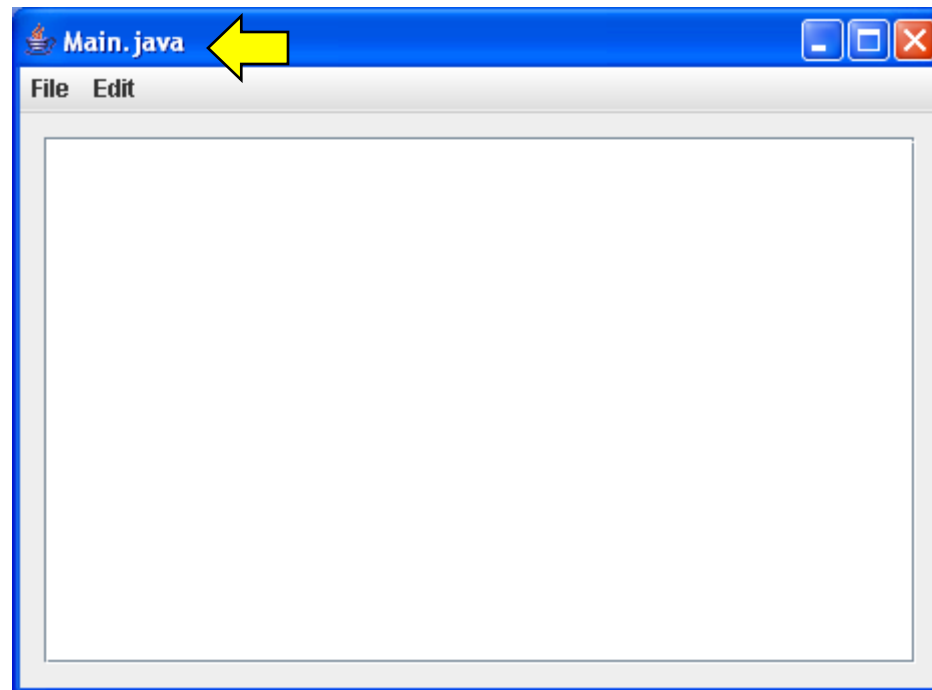


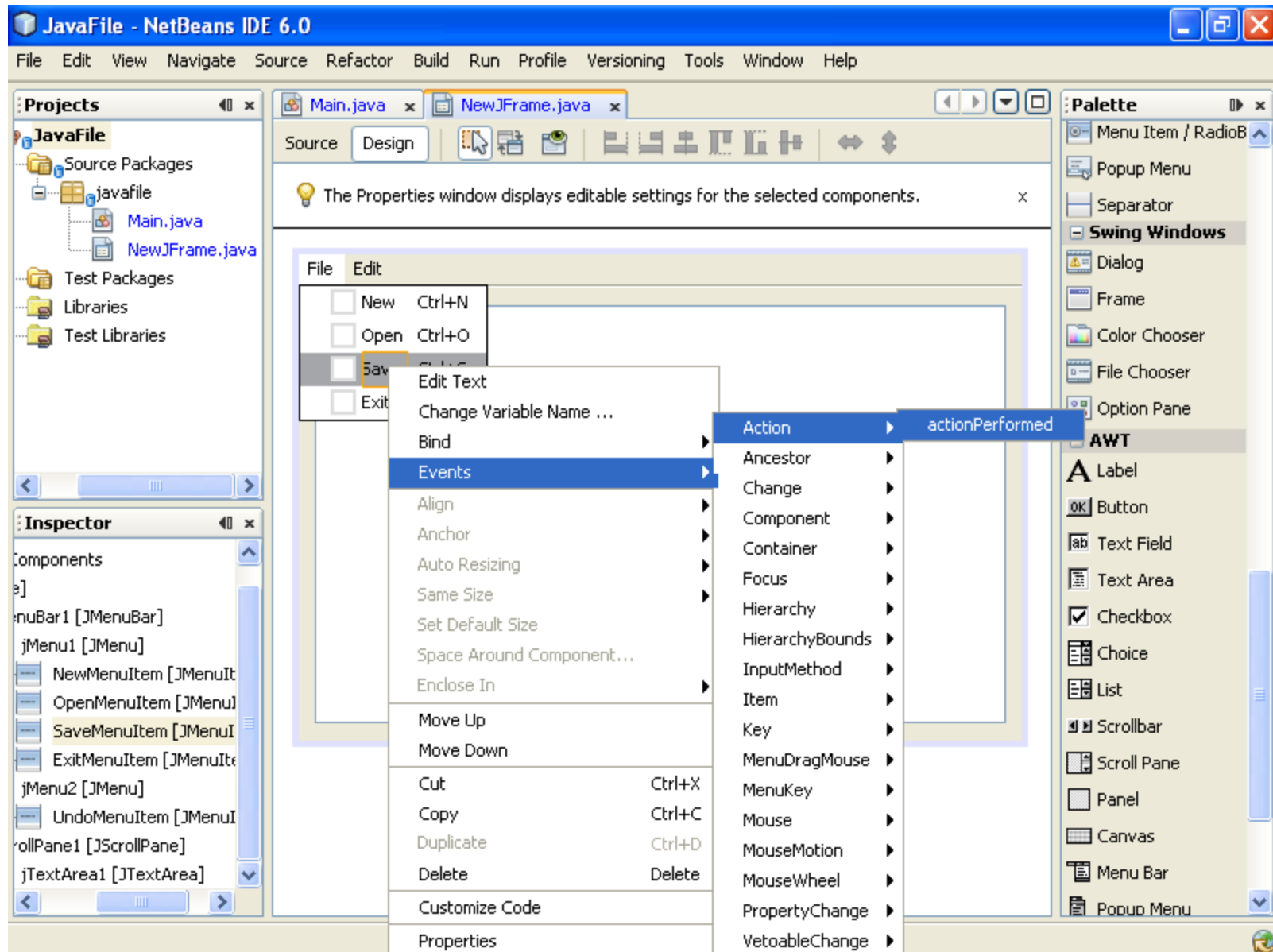


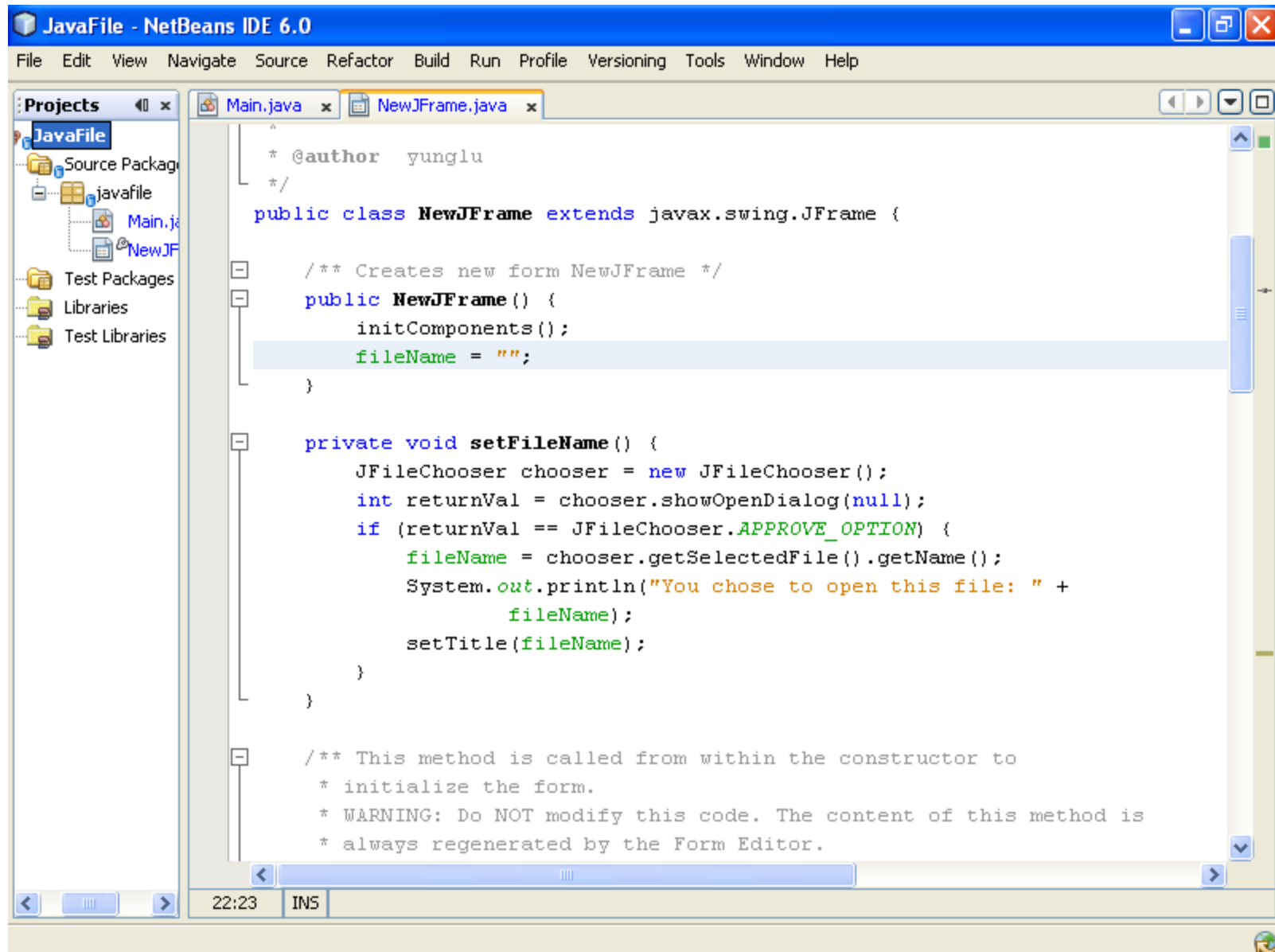


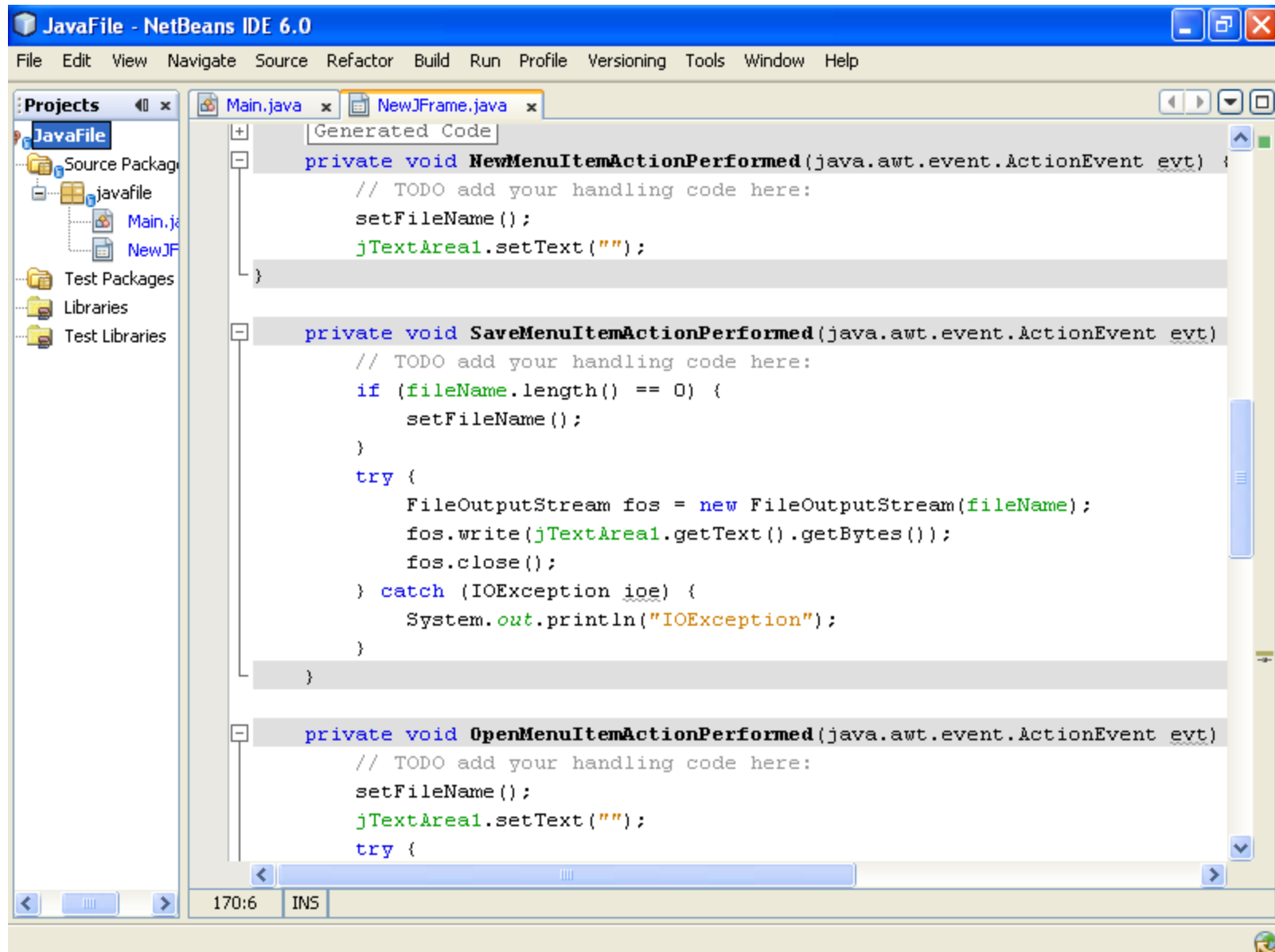


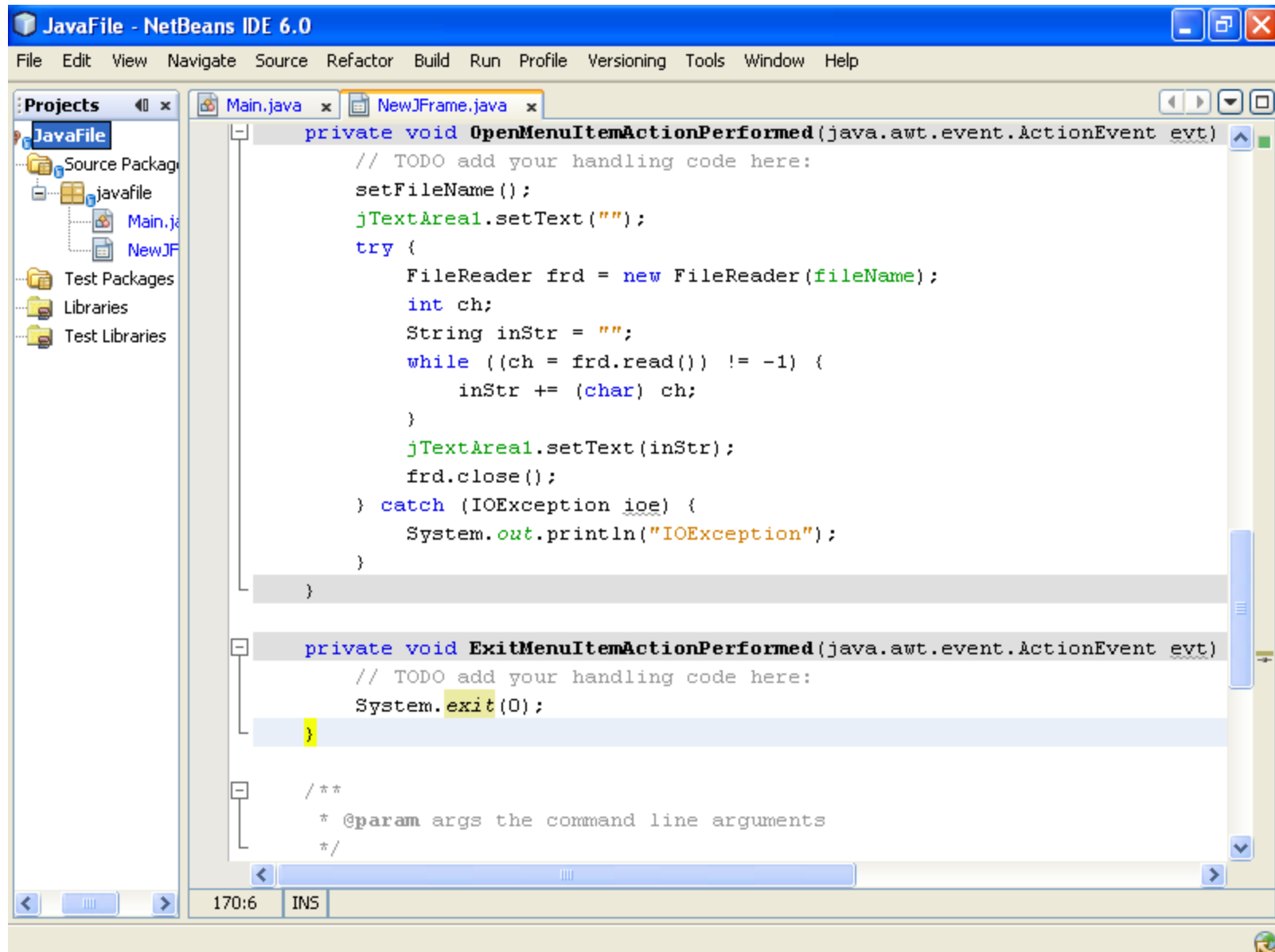












To Do

- handle Edit-Undo
 - ⇒ record the latest change for undo
- check whether the current text has been saved before File-New or File-Open
 - ⇒ keep a flag that indicates the content in the text area has changed but not saved; reset the flag after the file has been saved

Save and Open File in C++

Modified from the
Main Window - Menus
example in Qt

`/home/shay/a/sfwtools/public/qt4.3.0/examples`

Qt 4.3: Menu Example - Mozilla Firefox

File Edit View History Bookmarks Yahoo! Tools Help

http://doc.trolltech.com/4.3/mainwindows-menus.html

Google

Home · All Classes · Main Classes · Grouped Classes · Modules · Functions

TROLLTECH

Menu Example

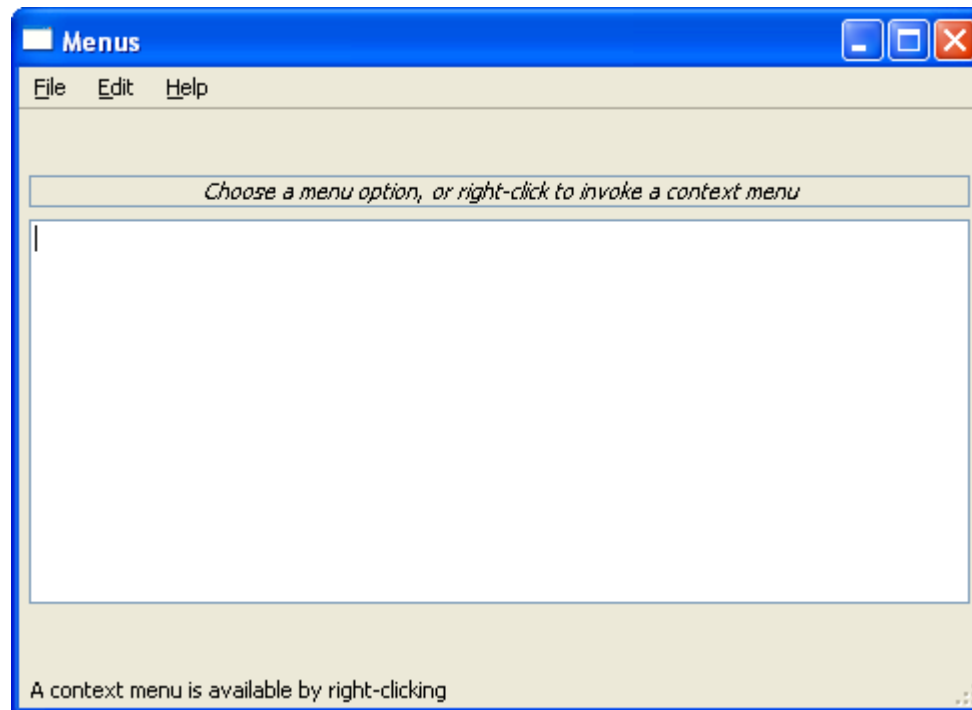
Files:

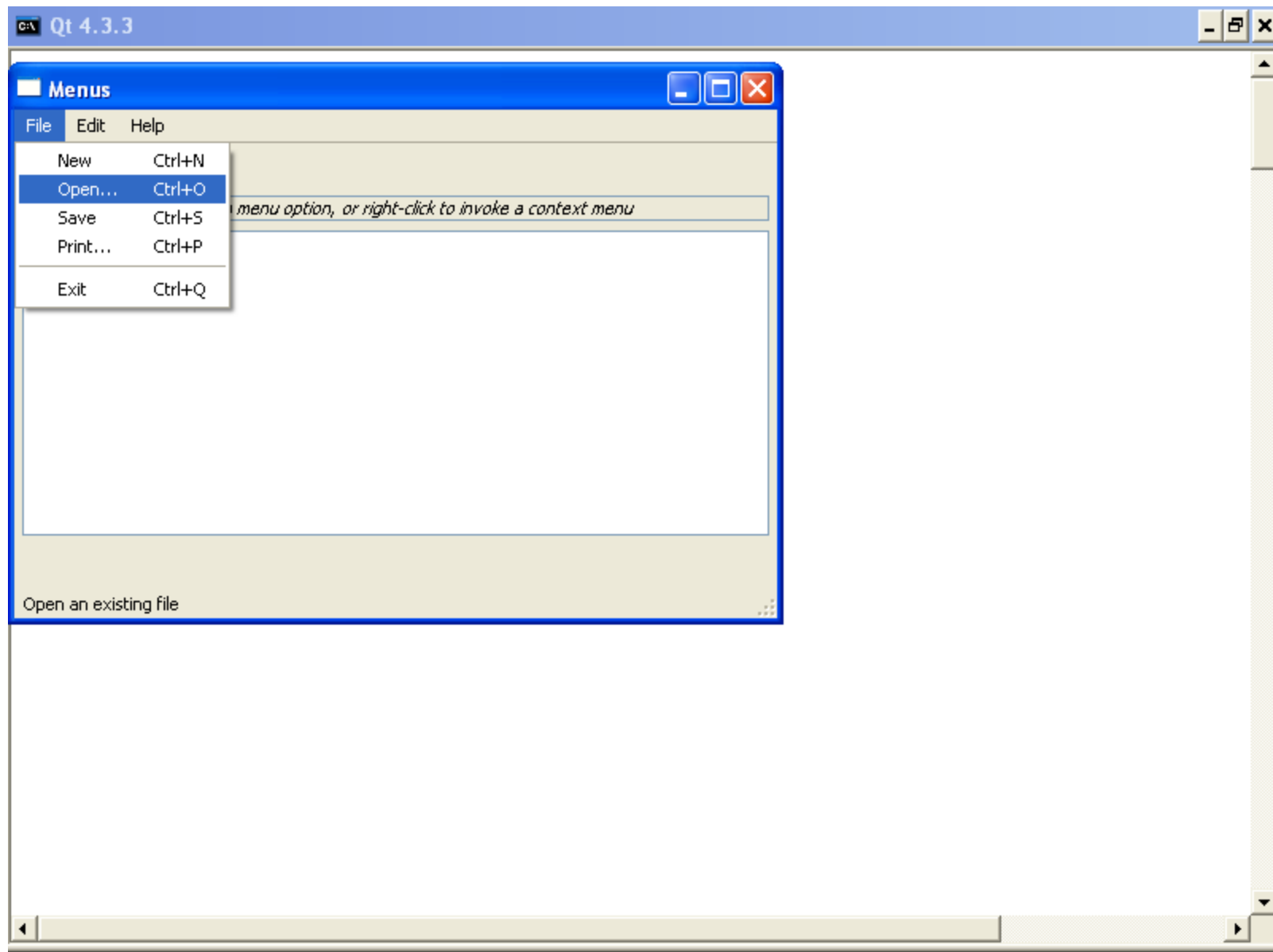
- [mainwindows/menus/mainwindow.cpp](#)
- [mainwindows/menus/mainwindow.h](#)
- [mainwindows/menus/main.cpp](#)

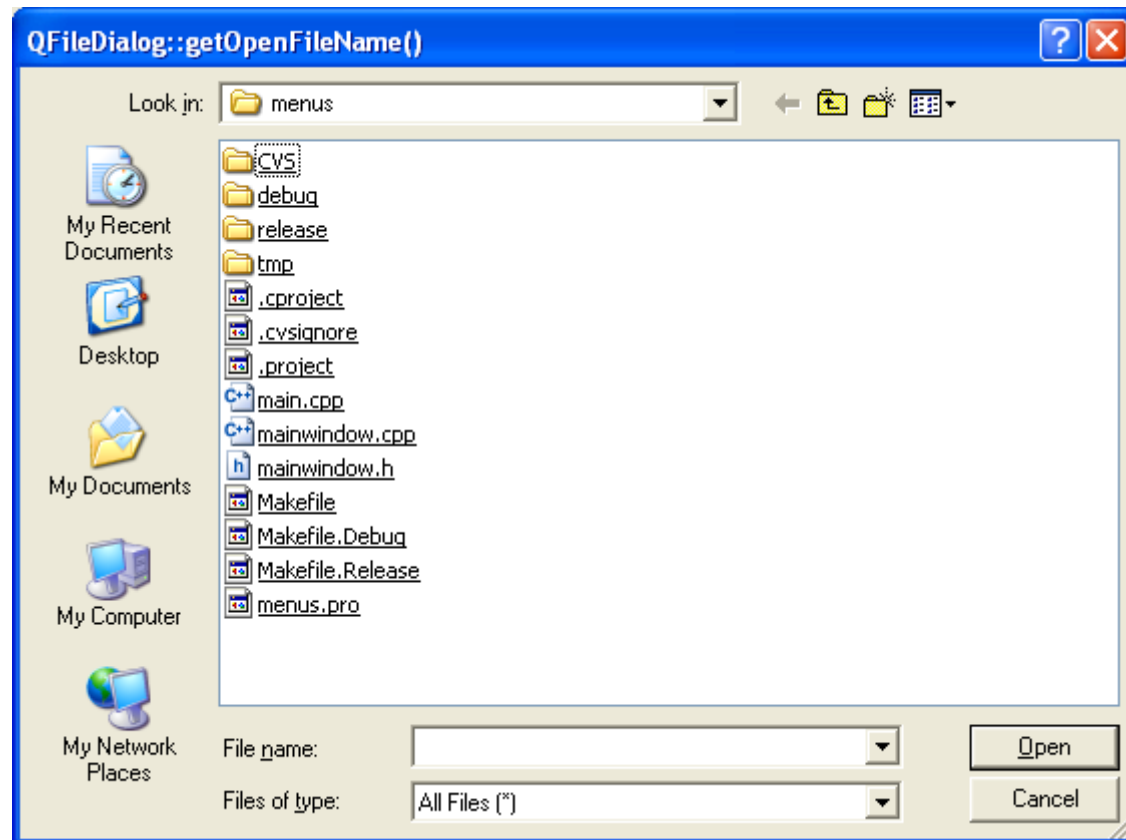
The Menu example demonstrates how menus can be used in a main window application.

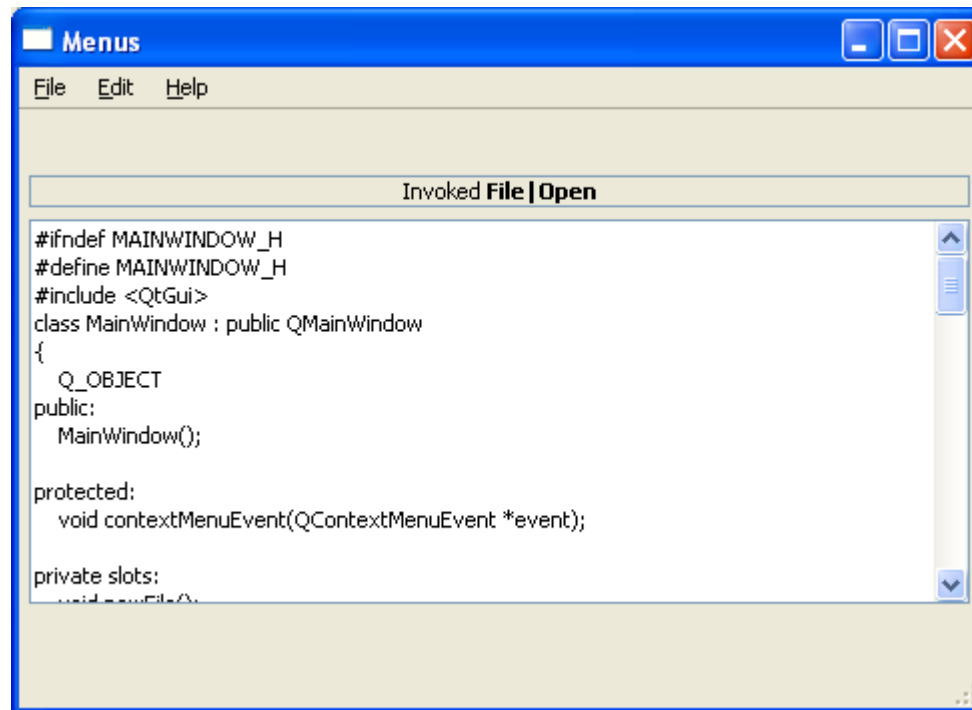
A menu widget can be either a pull-down menu in a menu bar or a standalone context menu. Pull-down menus are shown by the menu bar when the user clicks on the respective item or presses the specified shortcut key. Context menus are usually invoked by some special keyboard key or by right-clicking.

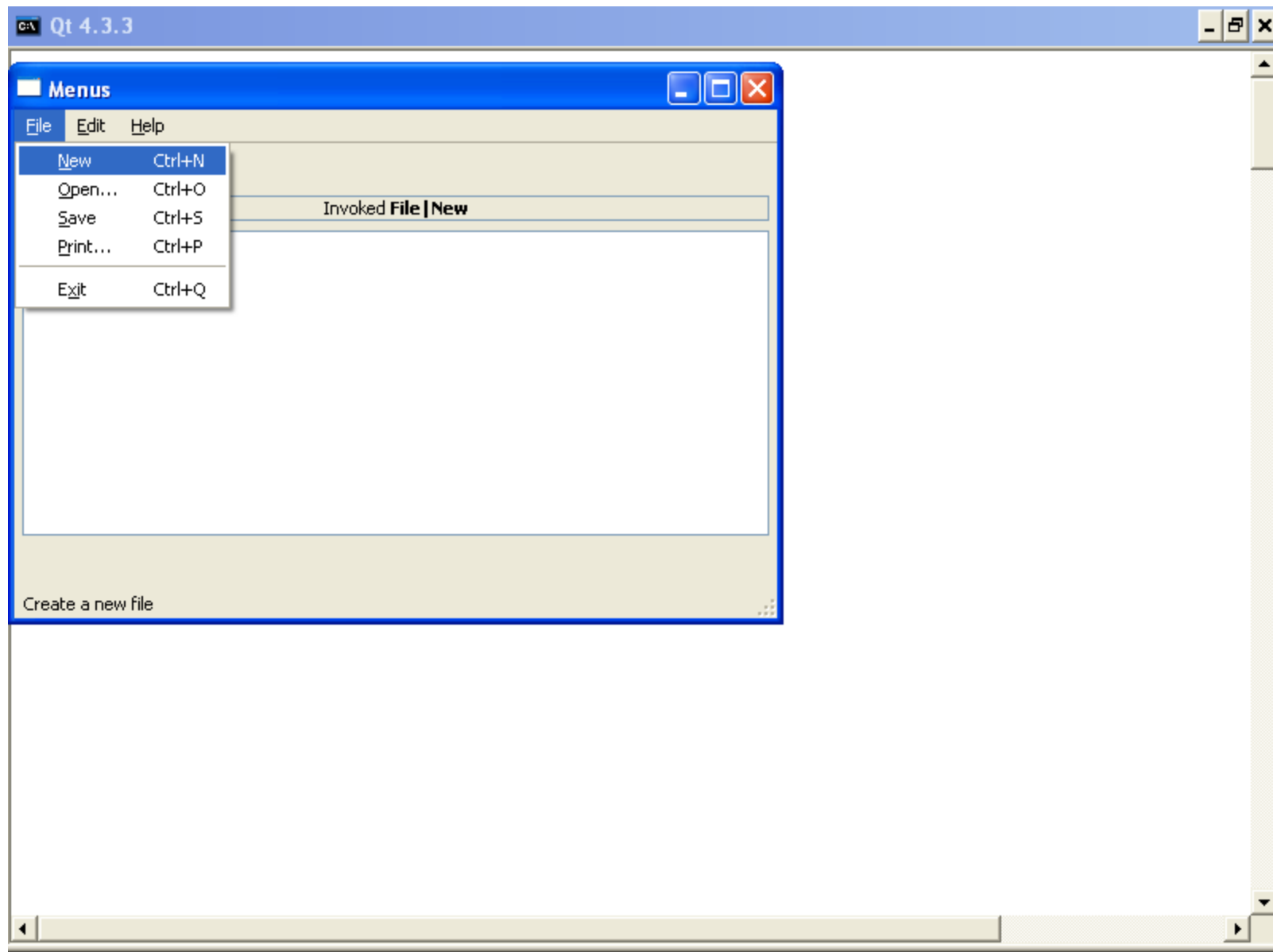
Done

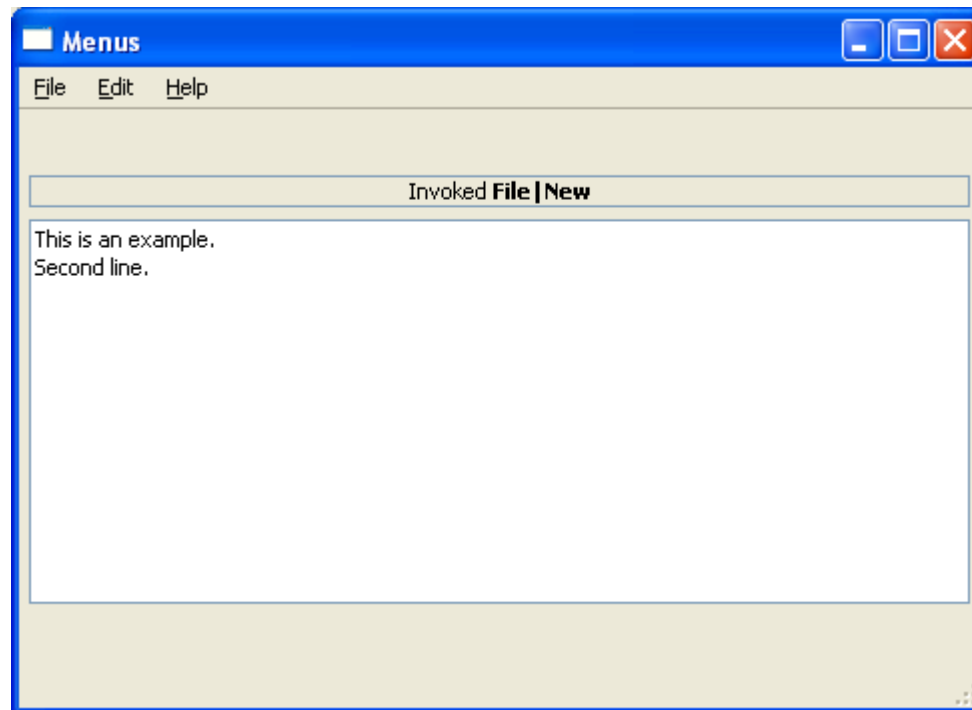


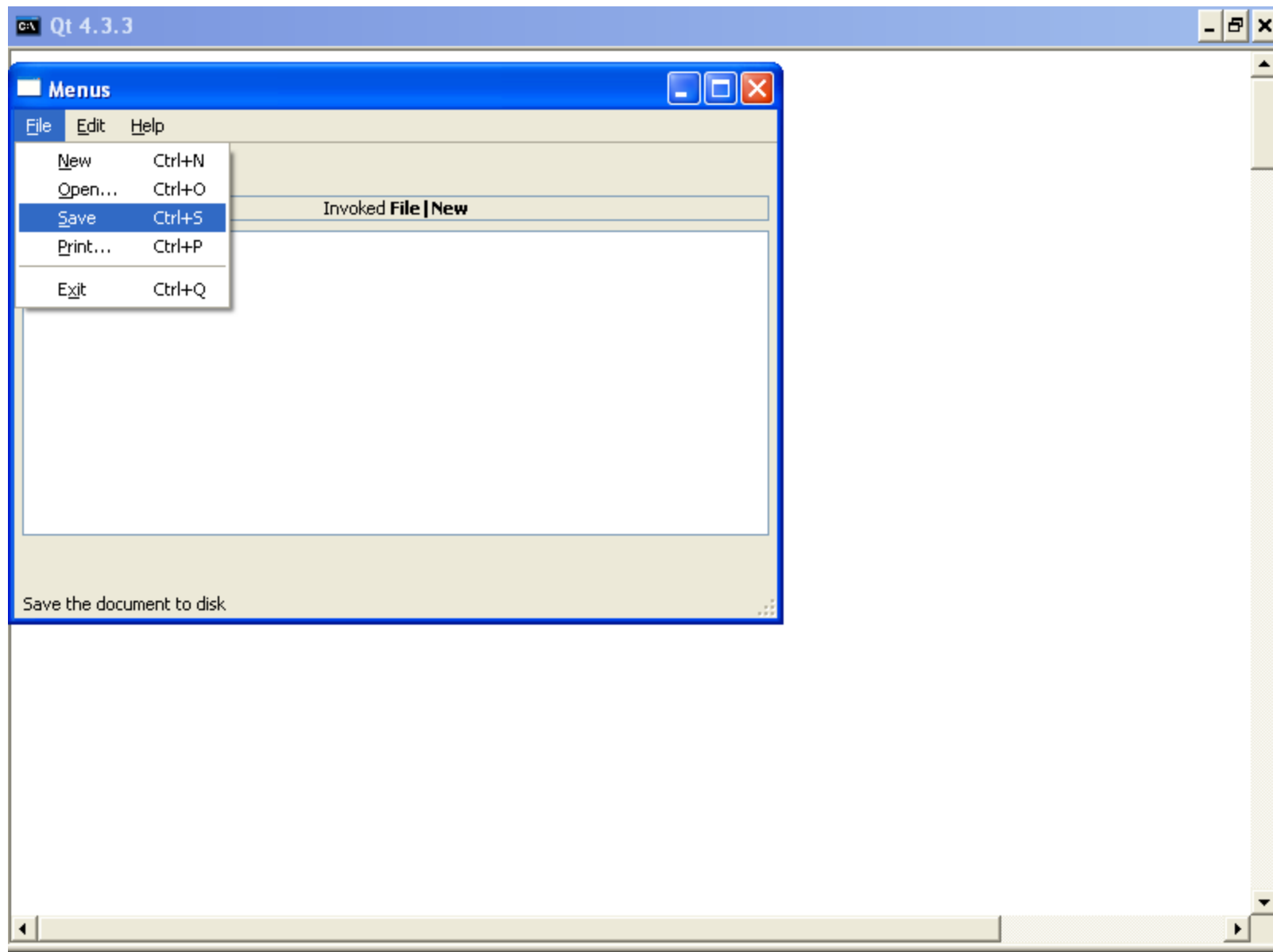


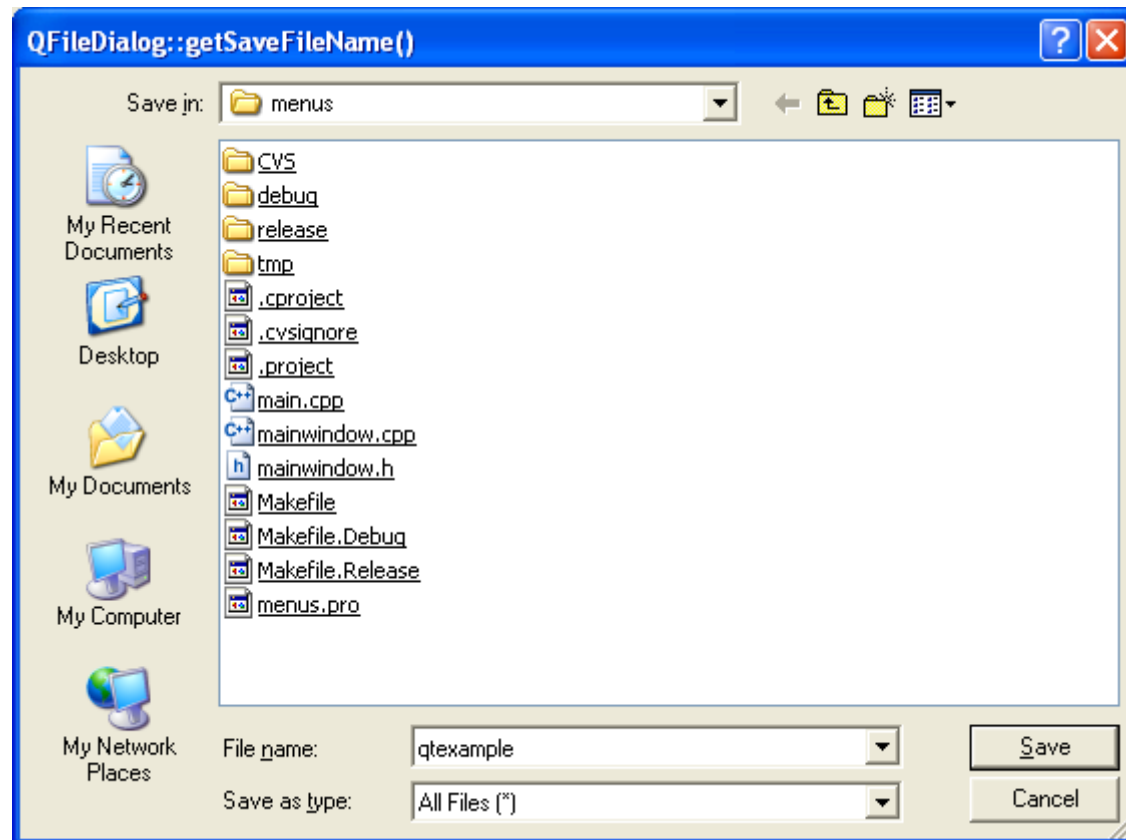


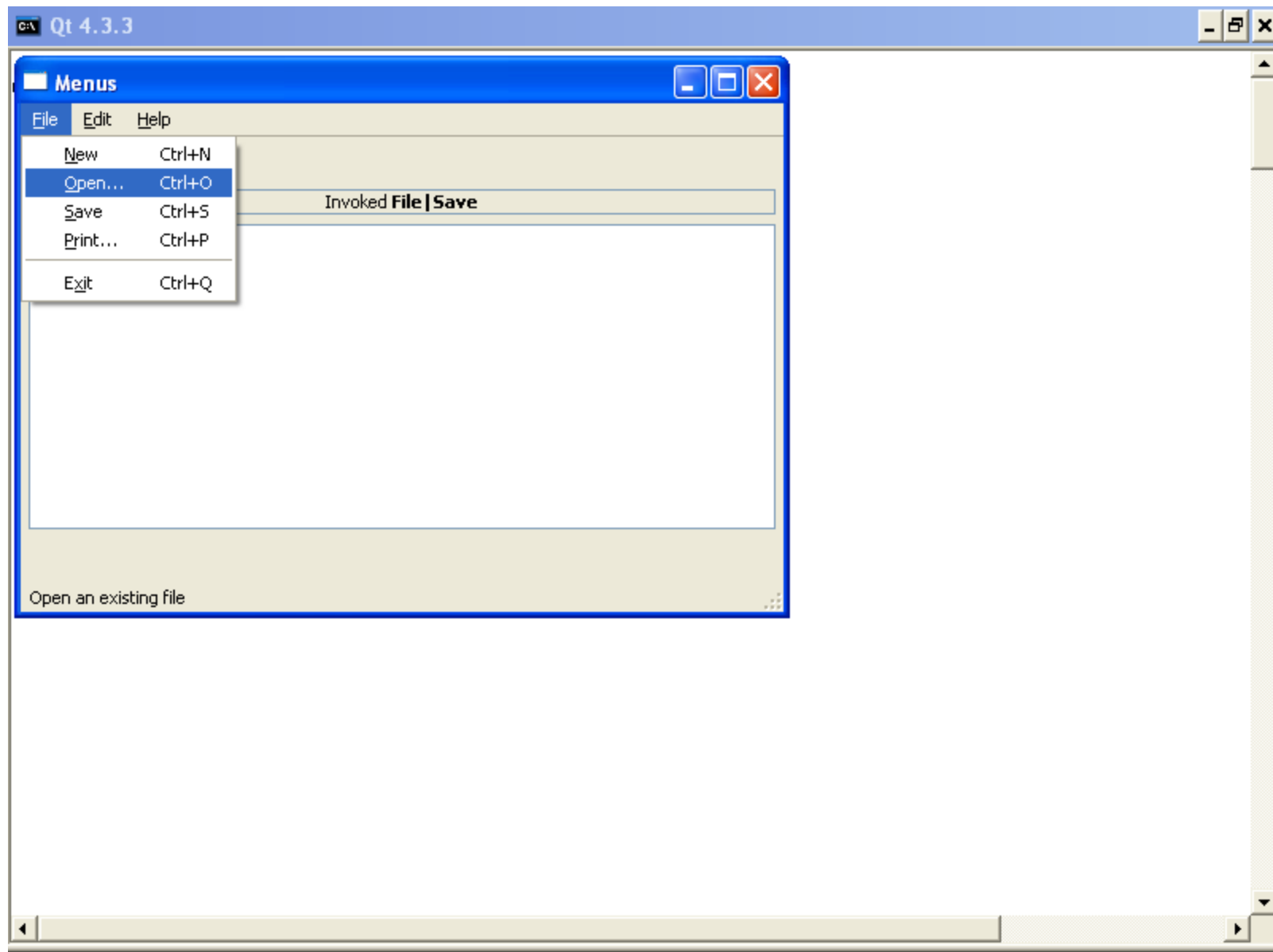


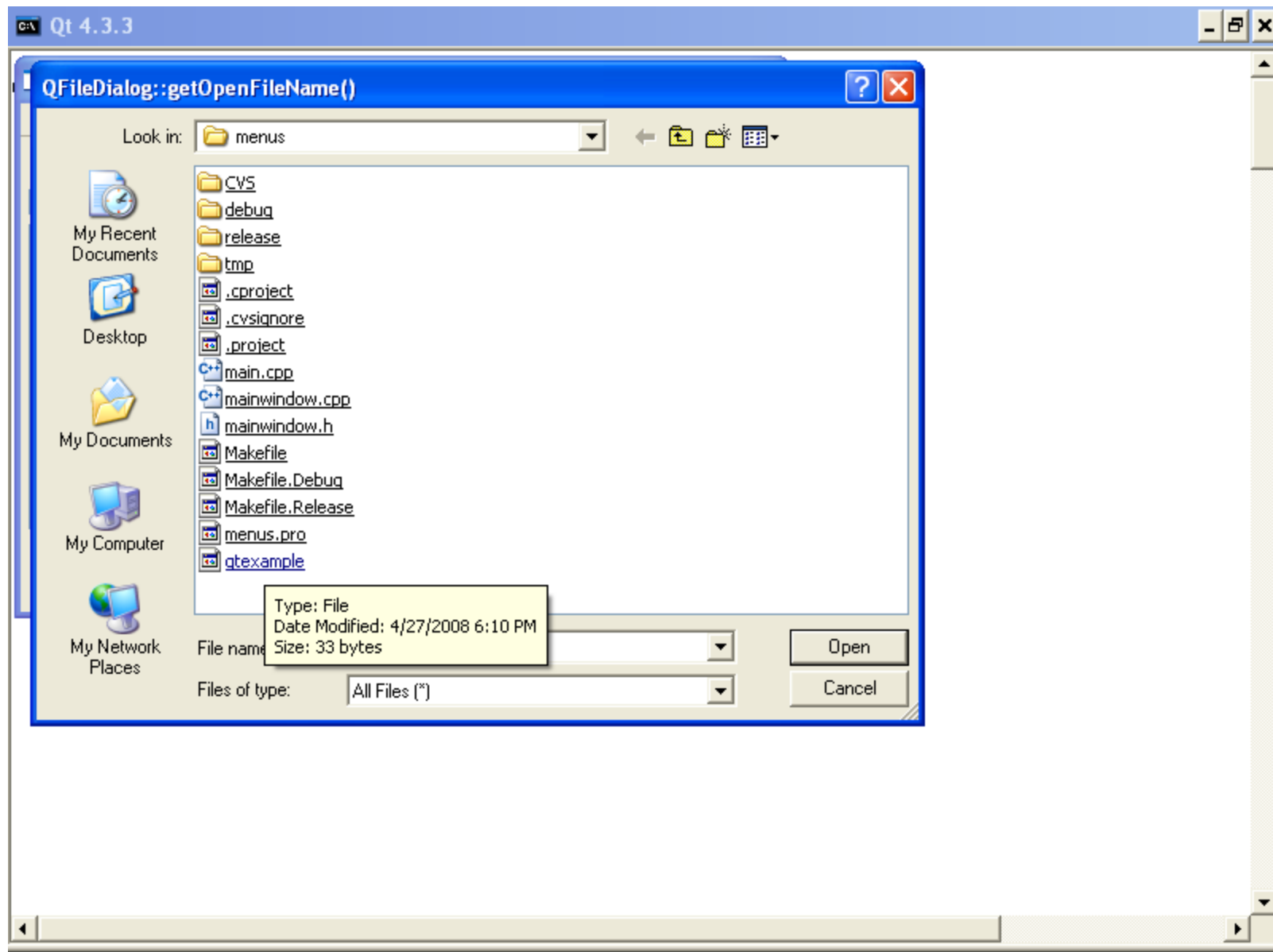


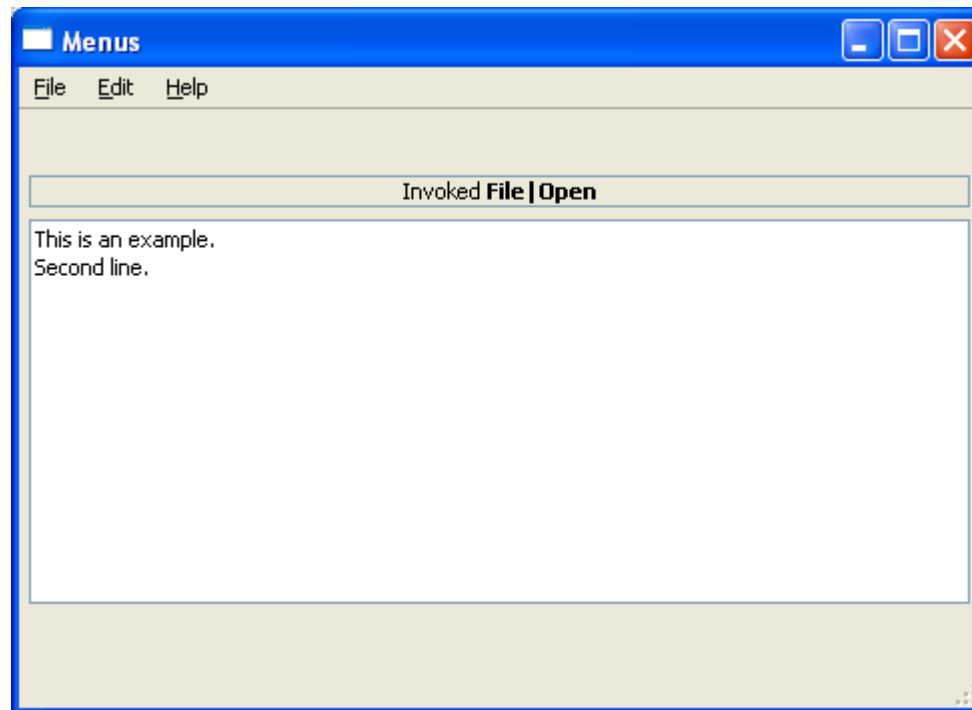


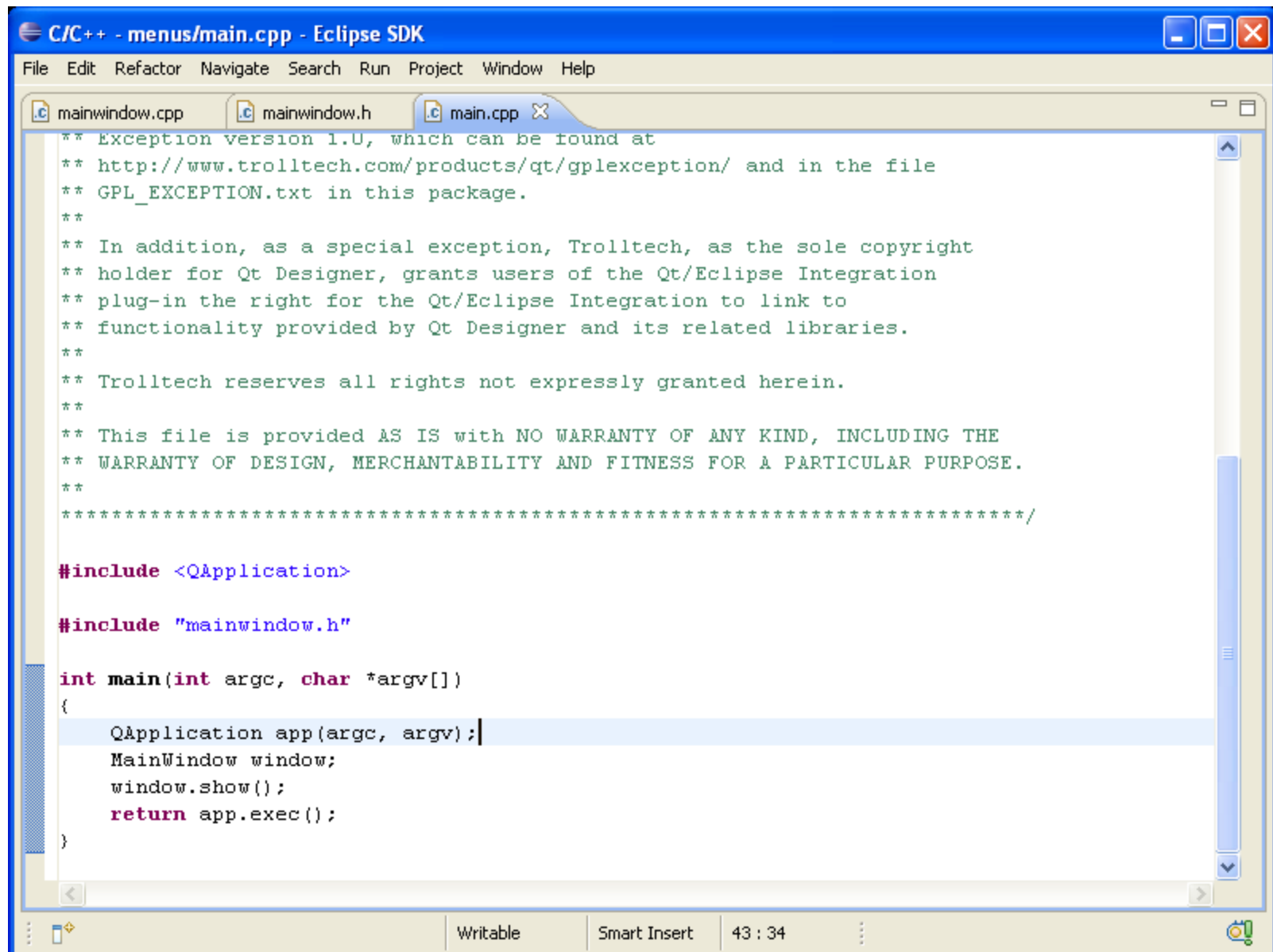












```
C/C++ - menus/main.cpp - Eclipse SDK
File Edit Refactor Navigate Search Run Project Window Help

mainwindow.cpp mainwindow.h main.cpp X

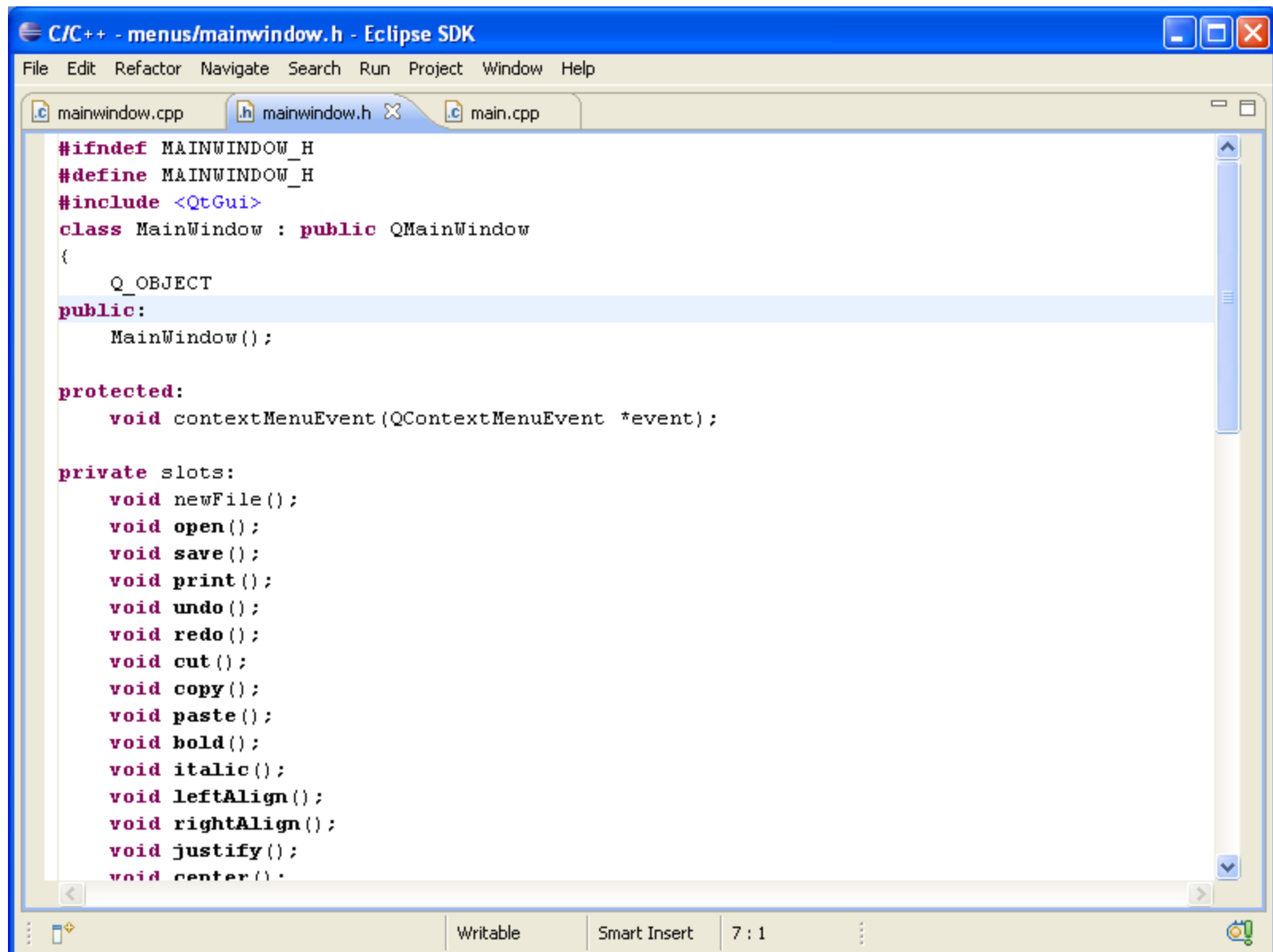
** Exception version 1.0, which can be found at
** http://www.trolltech.com/products/qt/gplexception/ and in the file
** GPL_EXCEPTION.txt in this package.
**
** In addition, as a special exception, Trolltech, as the sole copyright
** holder for Qt Designer, grants users of the Qt/Eclipse Integration
** plug-in the right for the Qt/Eclipse Integration to link to
** functionality provided by Qt Designer and its related libraries.
**
** Trolltech reserves all rights not expressly granted herein.
**
** This file is provided AS IS with NO WARRANTY OF ANY KIND, INCLUDING THE
** WARRANTY OF DESIGN, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.
**
**/

#include <QApplication>

#include "mainwindow.h"

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

Writable Smart Insert 43 : 34



The screenshot shows the Eclipse IDE interface. The title bar reads 'C/C++ - menus/mainwindow.h - Eclipse SDK'. The menu bar includes 'File', 'Edit', 'Refactor', 'Navigate', 'Search', 'Run', 'Project', 'Window', and 'Help'. The editor has three tabs: 'mainwindow.cpp', 'mainwindow.h' (which is the active tab), and 'main.cpp'. The code in 'mainwindow.h' is as follows:

```
#ifndef MAINWINDOW_H
#define MAINWINDOW_H
#include <QtGui>
class MainWindow : public QMainWindow
{
    Q_OBJECT
public:
    MainWindow();

protected:
    void contextMenuEvent(QContextMenuEvent *event);

private slots:
    void newFile();
    void open();
    void save();
    void print();
    void undo();
    void redo();
    void cut();
    void copy();
    void paste();
    void bold();
    void italic();
    void leftAlign();
    void rightAlign();
    void justify();
    void center();
};
```

The status bar at the bottom shows 'Writable', 'Smart Insert', and '7 : 1'.

```
C/C++ - menus/mainwindow.h - Eclipse SDK
File Edit Refactor Navigate Search Run Project Window Help

mainwindow.cpp mainwindow.h main.cpp

void center();
void setLineSpacing();
void setParagraphSpacing();
void about();
void aboutQt();

private:
void createActions();
void createMenus();
void setFileName();

QMenu *fileMenu;
QMenu *editMenu;
QMenu *formatMenu;
QMenu *helpMenu;
QActionGroup *alignmentGroup;
QAction *newAct;
QAction *openAct;
QAction *saveAct;
QAction *printAct;
QAction *exitAct;
QAction *undoAct;
QAction *redoAct;
QAction *cutAct;
QAction *copyAct;
QAction *pasteAct;
QAction *boldAct;
QAction *italicAct;
```

Writable Smart Insert 7 : 1

```
C/C++ - menus/mainwindow.h - Eclipse SDK
File Edit Refactor Navigate Search Run Project Window Help

mainwindow.cpp mainwindow.h x main.cpp

QActionGroup *alignmentGroup;
QAction *newAct;
QAction *openAct;
QAction *saveAct;
QAction *printAct;
QAction *exitAct;
QAction *undoAct;
QAction *redoAct;
QAction *cutAct;
QAction *copyAct;
QAction *pasteAct;
QAction *boldAct;
QAction *italicAct;
QAction *leftAlignAct;
QAction *rightAlignAct;
QAction *justifyAct;
QAction *centerAct;
QAction *setLineSpacingAct;
QAction *setParagraphSpacingAct;
QAction *aboutAct;
QAction *aboutQtAct;
QLabel *infoLabel;
QTextEdit * textEdit;
QString fileName;
};

#endif

different from the example

Writable Smart Insert 7 : 1
```

```
C/C++ - menus/mainwindow.cpp - Eclipse SDK
File Edit Refactor Navigate Search Run Project Window Help

mainwindow.cpp x mainwindow.h main.cpp

#include <QtGui>
#include "mainwindow.h"

MainWindow::MainWindow() {
    QWidget *widget = new QWidget;
    setCentralWidget(widget);

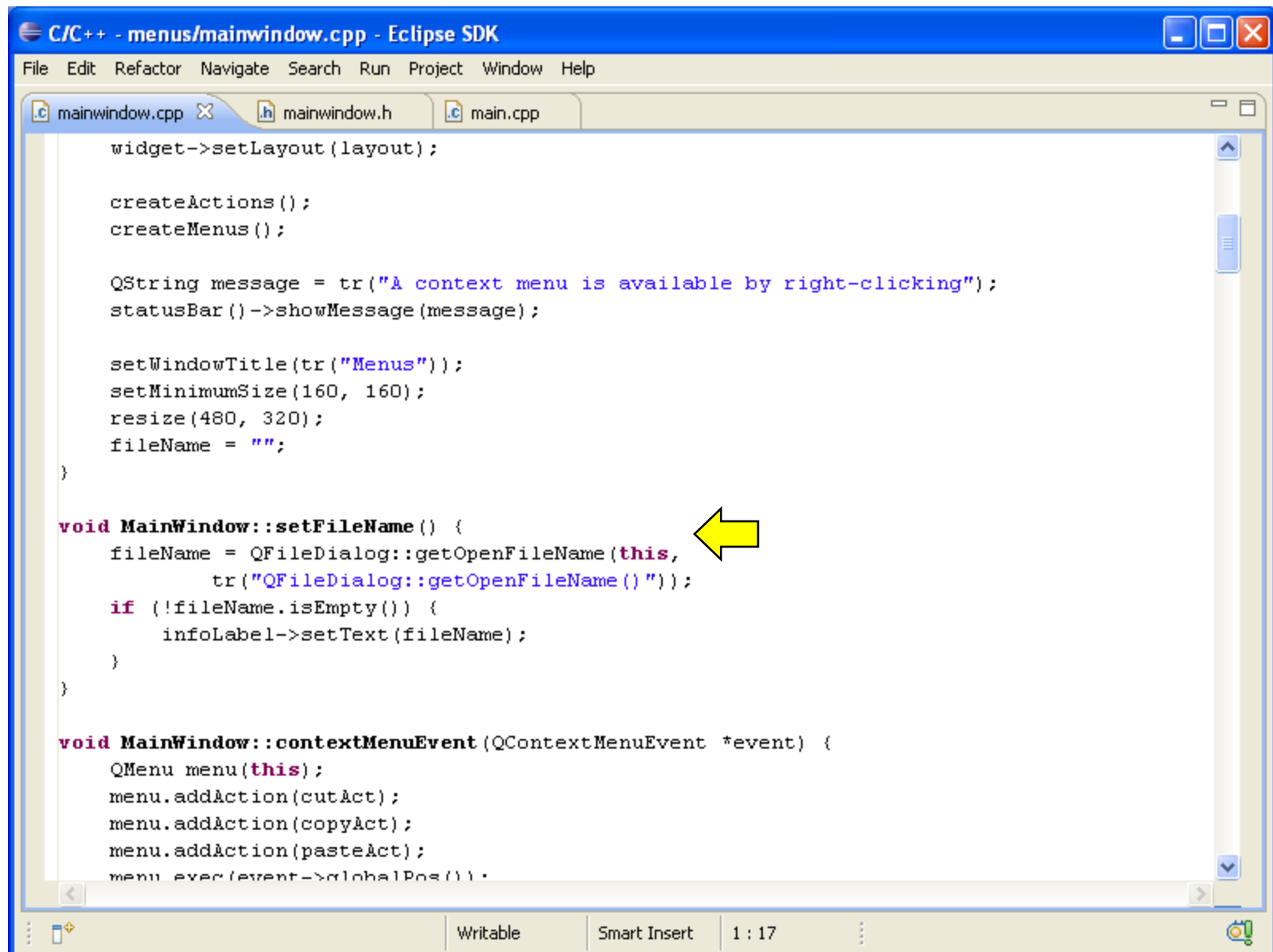
    QWidget *topFiller = new QWidget;
    topFiller->setSizePolicy(QSizePolicy::Expanding, QSizePolicy::Expanding);
    textEdit = new QTextEdit();
    infoLabel = new QLabel(tr("<i>Choose a menu option, or right-click to "
        "invoke a context menu</i>"));
    infoLabel->setFrameStyle(QFrame::StyledPanel | QFrame::Sunken);
    infoLabel->setAlignment(Qt::AlignCenter);

    QWidget *bottomFiller = new QWidget;
    bottomFiller->setSizePolicy(QSizePolicy::Expanding, QSizePolicy::Expanding);

    QVBoxLayout *layout = new QVBoxLayout;
    layout->setMargin(5);
    layout->addWidget(topFiller);
    layout->addWidget(infoLabel);
    layout->addWidget(textEdit);
    layout->addWidget(bottomFiller);
    widget->setLayout(layout);

    createActions();
    createMenus();
}
```

Writable Smart Insert 1 : 17



```
C/C++ - menus/mainwindow.cpp - Eclipse SDK
File Edit Refactor Navigate Search Run Project Window Help

mainwindow.cpp x mainwindow.h main.cpp

    widget->setLayout(layout);

    createActions();
    createMenus();

    QString message = tr("A context menu is available by right-clicking");
    statusBar()->showMessage(message);

    setWindowTitle(tr("Menus"));
    setMinimumSize(160, 160);
    resize(480, 320);
    fileName = "";
}

void MainWindow::setFileName() {
    fileName = QFileDialog::getOpenFileName(this,
        tr("QFileDialog::getOpenFileName()"));
    if (!fileName.isEmpty()) {
        infoLabel->setText(fileName);
    }
}

void MainWindow::contextMenuEvent(QContextMenuEvent *event) {
    QMenu menu(this);
    menu.addAction(cutAct);
    menu.addAction(copyAct);
    menu.addAction(pasteAct);
    menu.exec(event->globalPos());
}
```

Writable Smart Insert 1 : 17

```
C/C++ - menus/mainwindow.cpp - Eclipse SDK
File Edit Refactor Navigate Search Run Project Window Help

mainwindow.cpp x mainwindow.h main.cpp

void MainWindow::contextMenuEvent(QContextMenuEvent *event) {
    QMenu menu(this);
    menu.addAction(cutAct);
    menu.addAction(copyAct);
    menu.addAction(pasteAct);
    menu.exec(event->globalPos());
}

void MainWindow::newFile() {
    setFileName();
    textEdit->clear();
    infoLabel->setText(tr("Invoked <b>File|New</b>"));
}

void MainWindow::open() {
    setFileName();
    QFile fl(fileName);
    if (fl.open(QIODevice::ReadOnly | QIODevice::Text))
    {
        QByteArray qba = fl.readAll();
        QString txt(qba);
        textEdit -> setText(txt);
        fl.close();
    }
    infoLabel->setText(tr("Invoked <b>File|Open</b>"));
}
```

Writable Smart Insert 1 : 17

```
C/C++ - menus/mainwindow.cpp - Eclipse SDK
File Edit Refactor Navigate Search Run Project Window Help

mainwindow.cpp x mainwindow.h main.cpp

void MainWindow::save() {
    fileName = QFileDialog::getSaveFileName(this,
        tr("QFileDialog::getSaveFileName()"));
    QFile fl(fileName);
    if (fl.open(QIODevice::WriteOnly | QIODevice::Text))
    {
        QString txt = textEdit->toPlainText();
        fl.write(txt.toAscii());
        fl.close();
    }
    infoLabel->setText(tr("Invoked <b>File|Save</b>"));
}

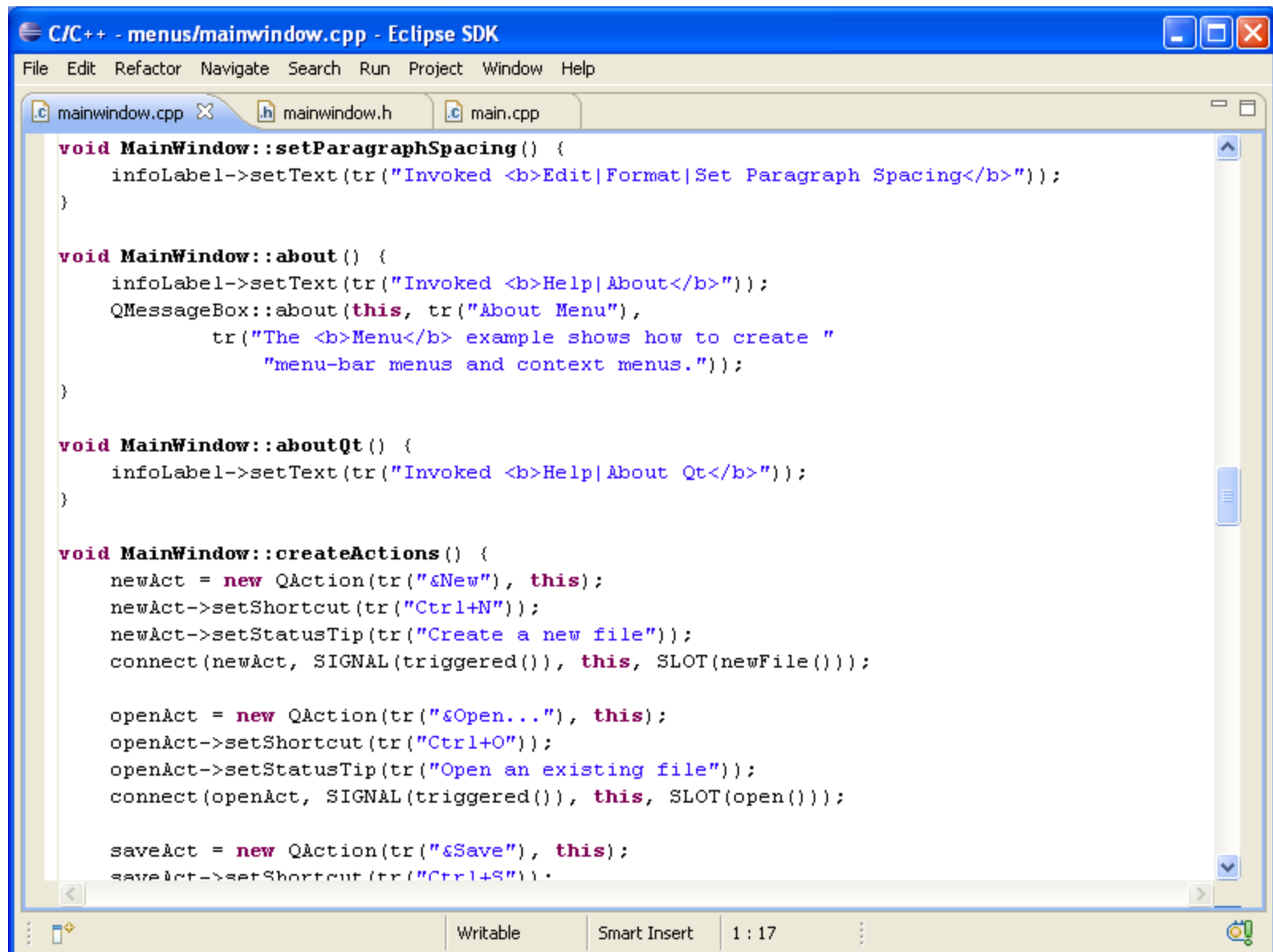
void MainWindow::print() {
    infoLabel->setText(tr("Invoked <b>File|Print</b>"));
}

void MainWindow::undo() {
    infoLabel->setText(tr("Invoked <b>Edit|Undo</b>"));
}

void MainWindow::redo() {
    infoLabel->setText(tr("Invoked <b>Edit|Redo</b>"));
}

void MainWindow::cut() {
    infoLabel->setText(tr("Invoked <b>Edit|Cut</b>"));
}

Writable Smart Insert 1 : 17
```



```
C/C++ - menus/mainwindow.cpp - Eclipse SDK
File Edit Refactor Navigate Search Run Project Window Help

mainwindow.cpp x mainwindow.h main.cpp

void MainWindow::setParagraphSpacing() {
    infoLabel->setText(tr("Invoked <b>Edit|Format|Set Paragraph Spacing</b>"));
}

void MainWindow::about() {
    infoLabel->setText(tr("Invoked <b>Help|About</b>"));
    QMessageBox::about(this, tr("About Menu"),
        tr("The <b>Menu</b> example shows how to create "
            "menu-bar menus and context menus.));
}

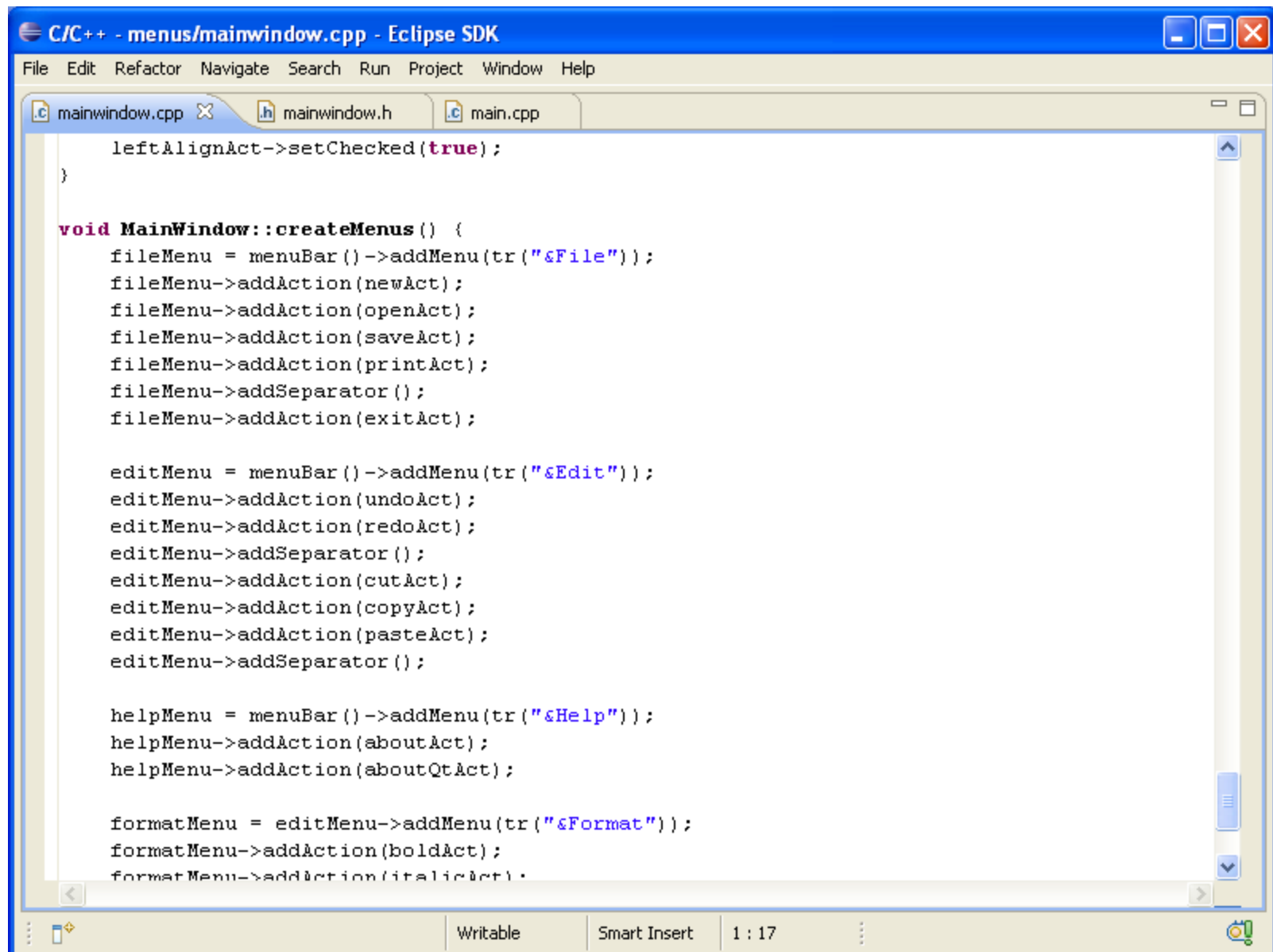
void MainWindow::aboutQt() {
    infoLabel->setText(tr("Invoked <b>Help|About Qt</b>"));
}

void MainWindow::createActions() {
    newAct = new QAction(tr("&New"), this);
    newAct->setShortcut(tr("Ctrl+N"));
    newAct->setStatusTip(tr("Create a new file"));
    connect(newAct, SIGNAL(triggered()), this, SLOT(newFile()));

    openAct = new QAction(tr("&Open..."), this);
    openAct->setShortcut(tr("Ctrl+O"));
    openAct->setStatusTip(tr("Open an existing file"));
    connect(openAct, SIGNAL(triggered()), this, SLOT(open()));

    saveAct = new QAction(tr("&Save"), this);
    saveAct->setShortcut(tr("Ctrl+S"));
}
```

Writable Smart Insert 1 : 17



C/C++ - menus/mainwindow.cpp - Eclipse SDK

File Edit Refactor Navigate Search Run Project Window Help

mainwindow.cpp x mainwindow.h main.cpp

```
        leftAlignAct->setChecked(true);
    }

    void MainWindow::createMenus() {
        fileMenu = menuBar()->addMenu(tr("&File"));
        fileMenu->addAction(newAct);
        fileMenu->addAction(openAct);
        fileMenu->addAction(saveAct);
        fileMenu->addAction(printAct);
        fileMenu->addSeparator();
        fileMenu->addAction(exitAct);

        editMenu = menuBar()->addMenu(tr("&Edit"));
        editMenu->addAction(undoAct);
        editMenu->addAction(redoAct);
        editMenu->addSeparator();
        editMenu->addAction(cutAct);
        editMenu->addAction(copyAct);
        editMenu->addAction(pasteAct);
        editMenu->addSeparator();

        helpMenu = menuBar()->addMenu(tr("&Help"));
        helpMenu->addAction(aboutAct);
        helpMenu->addAction(aboutQtAct);

        formatMenu = editMenu->addMenu(tr("&Format"));
        formatMenu->addAction(boldAct);
        formatMenu->addAction(italicAct);
    }
```

Writable Smart Insert 1 : 17

Valgrind and Qt

```
emacs@DESKTOP
Buffers  Files  Tools  Edit  Search  Mule  C  Help

#include <QtGui>
#include "mainwindow.h"
MainWindow::MainWindow() {
    // int * ptr = new int[10000];
}
MainWindow::~MainWindow() {
}

--(Unix)--  mainwindow.cpp      (C++) --L6--All-----
#ifdef MAINWINDOW_H
#define MAINWINDOW_H
#include <QtGui>
class MainWindow : public QMainWindow
{
    Q_OBJECT
public:
    MainWindow();
    ~MainWindow();
};
#endif

--(Unix)--  mainwindow.h      (C) --L12--All-----
#include <QApplication>
#include "mainwindow.h"
int main(int argc, char *argv[])
{
    QApplication app(argc, argv);
    MainWindow window;
    window.show();
    return app.exec();
}

--(Unix)--  main.cpp      (C++) --L1--All-----
Wrote e:/mainwindow.h
```

The program doesn't
do anything.



```
File Edit View Terminal Tabs Help
==6115== by 0x4CF0A68: g_slice_alloc (in /usr/lib/libglib-2.0.so.0.1400.1)
==6115== by 0x4CB6578: g_array_sized_new (in /usr/lib/libglib-2.0.so.0.1400.1)
==6115== by 0x4CB6686: g_array_new (in /usr/lib/libglib-2.0.so.0.1400.1)
==6115== by 0x4CF7F06: g_static_private_set (in /usr/lib/libglib-2.0.so.0.1400.1)
==6115== by 0x4CBE6DB: g_get_filename_charsets (in /usr/lib/libglib-2.0.so.0.1400.1)
==6115== by 0x4CBE750: (within /usr/lib/libglib-2.0.so.0.1400.1)
==6115== by 0x4CF81A9: g_thread_init_glib (in /usr/lib/libglib-2.0.so.0.1400.1)
==6115== by 0x4C98719: g_thread_init (in /usr/lib/libgthread-2.0.so.0.1400.1)
==6115== by 0x48F2528: QEventDispatcherGlibPrivate::QEventDispatcherGlibPrivate(_GMainContext*) (in /usr/lib/libQtCore.so.4.3.2)
==6115==
==6115== LEAK SUMMARY:
==6115== definitely lost: 358 bytes in 6 blocks.
==6115== indirectly lost: 572 bytes in 27 blocks.
==6115== possibly lost: 744 bytes in 3 blocks.
==6115== still reachable: 57,431 bytes in 1,982 blocks.
==6115== suppressed: 0 bytes in 0 blocks.
==6115== Reachable blocks (those to which a pointer was found)
==6115== To see them, rerun with: --leak-check=full --show-re
[Qt ] █
```

← report problems even though the program doesn't do anything.

```
emacs@DESKTOP
Buffers  Files  Tools  Edit  Search  Mule  C++  Help

#include <QtGui>
#include "mainwindow.h"
MainWindow::MainWindow() {
    int * ptr = new int[10000];
}
MainWindow::~MainWindow() {
}

--(Unix)--  mainwindow.cpp      (C++) --L4--All-----
#ifdef MAINWINDOW_H
#define MAINWINDOW_H
#include <QtGui>
class MainWindow : public QMainWindow
{
    Q_OBJECT
public:
    MainWindow();
    ~MainWindow();
};
#endif

--(Unix)--  mainwindow.h      (C) --L12--All-----
#include <QApplication>
#include "mainwindow.h"
int main(int argc, char *argv[])
{
    QApplication app(argc, argv);
    MainWindow window;
    window.show();
    return app.exec();
}

--(Unix)--  main.cpp      (C++) --L1--All-----
Wrote e:/mainwindow.cpp
```

```
File Edit View Terminal Tabs Help
==6129== by 0x4CBE6DB: g_get_filename_charsets (in /usr/lib/libglib-2.0.so.0.1400.1)
==6129== by 0x4CBE750: (within /usr/lib/libglib-2.0.so.0.1400.1)
==6129== by 0x4CF81A9: g_thread_init_glib (in /usr/lib/libglib-2.0.so.0.1400.1)
==6129== by 0x4C98719: g_thread_init (in /usr/lib/libgthread-2.0.so.0.1400.1)
==6129== by 0x48F2528: QEventDispatcherGlibPrivate::QEventDispatcherGlibPrivate(GMainContext*) (in /usr/lib/libQtCore.so.4.3.2)
==6129==
==6129==
==6129== 40,000 bytes in 1 blocks are definitely lost in loss record 76 of 76
==6129== at 0x4022B9C: operator new[](unsigned) (vg_replace_malloc.c:195)
==6129== by 0x8049EB0: MainWindow::MainWindow() (mainwindow.cpp:6)
==6129== by 0x8049D0E: main (main.cpp:6)
==6129==
==6129== LEAK SUMMARY:
==6129== definitely lost: 40,358 b
==6129== indirectly lost: 572 byte
==6129== possibly lost: 744 byte
==6129== still reachable: 57,431 b
==6129== suppressed: 0 bytes in 0 blocks.
==6129== Reachable blocks (those to which a pointer was found) are not shown.
==6129== To see them, rerun with: --leak-check=full --show-reachable=yes
[Qt ]
```

**memory leak in
MainWindow::MainWindow**

Qt and Valgrind

Valgrind - Mozilla Firefox

File Edit View History Bookmarks Yahoo! Tools Help

http://valgrind.org/docs/manual/faq.html#faq.overruns valgrind

5.2. Why doesn't Memcheck find the array overruns in this program?

```
int static[5];

int main(void)
{
    int stack[5];

    static[5] = 0;
    stack [5] = 0;

    return 0;
}
```

**valgrind does not check
index out of array bound**

Unfortunately, Memcheck doesn't do bounds checking on static or stack arrays. We'd like to, but it's just not possible to do in a reasonable way that fits with how Memcheck works. Sorry.

6. Miscellaneous

Done

Suppress False Warnings and Errors from Valgrind

If you do not want to see "errors" about the libraries.

Qt4-preview-feedback Archive - QT & valgrind - Mozilla Firefox

File Edit View History Bookmarks Yahoo! Tools Help

http://lists.trolltech.com/qt4-preview-feedback/2007-04/thread00043-1 valgrind qt

[Trolltech Home](#) | [Qt4-preview-feedback Home](#) | [Recent Threads](#) | [All Threads](#) | [Author](#) | [Date](#)

[All threads index page 1](#)

Qt4-preview-feedback Archive, April 2007

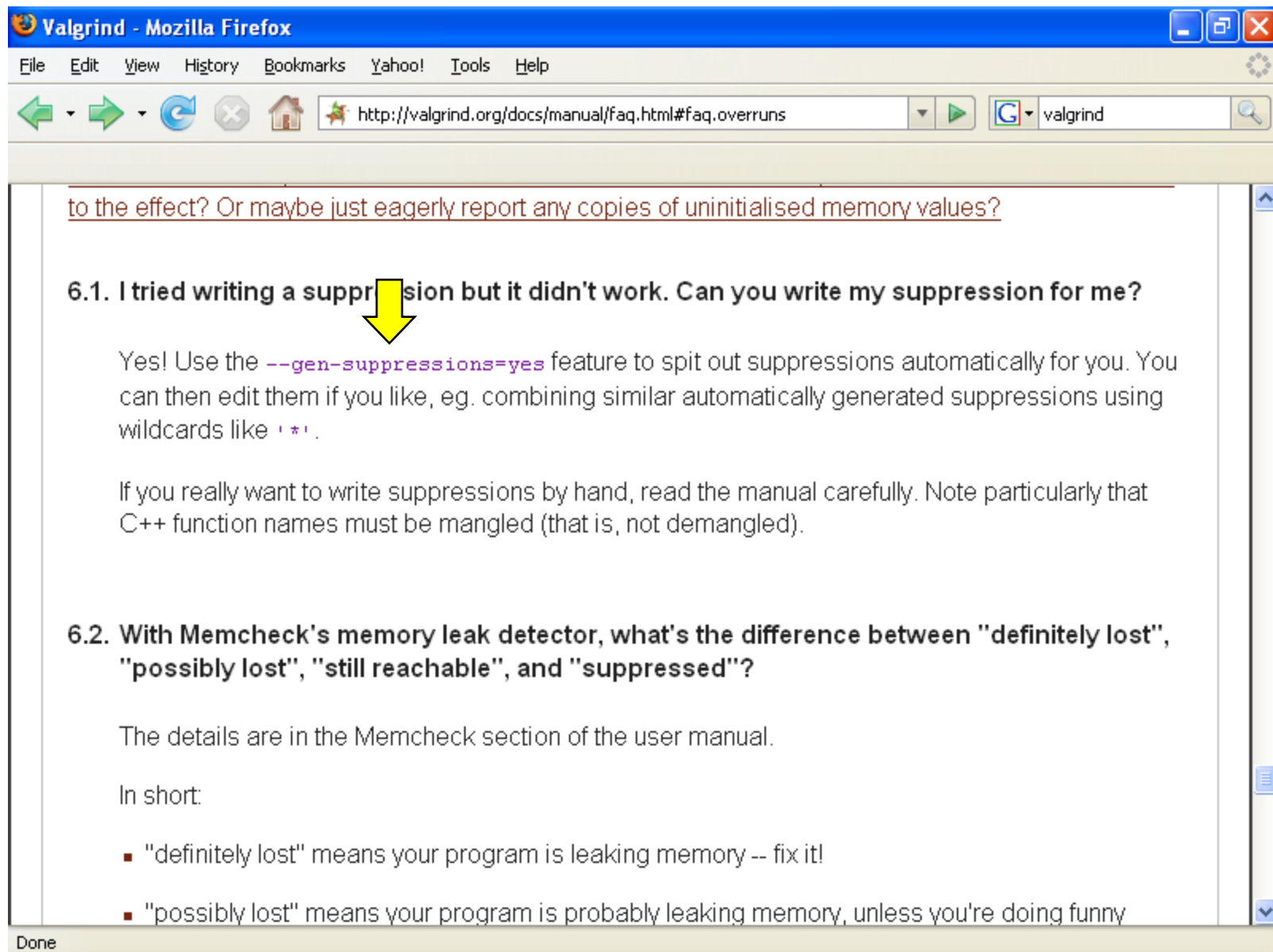
QT & valgrind

Message 1 in thread

- *Subject:* QT & valgrind
- *From:* [jeep@xxxxxxxxxx](#) (Jeff Lacki)
- *Date:* Fri, 13 Apr 2007 14:59:48 -0700 (PDT)
- *Delivered-to:* qt4-preview-feedback@trolltech.com
- *List-help:* <<mailto:qt4-preview-feedback-request@trolltech.com?subject=help>>
- *List-post:* <<mailto:qt4-preview-feedback@trolltech.com>>
- *List-subscribe:* <<mailto:qt4-preview-feedback-request@trolltech.com?subject=subscribe>>
- *List-unsubscribe:* <<mailto:qt4-preview-feedback-request@trolltech.com?subject=unsubscribe>>
- *Resent-from:* qt4-preview-feedback@xxxxxxxxxx
- *Resent-message-id:* <Qwndt.A.QKD.Y1_HGB@esparsett>
- *Resent-sender:* qt4-preview-feedback-request@xxxxxxxxxx
- *To:* qt4-preview-feedback@xxxxxxxxxx

Hi, I was just wondering (as I am running my app through

Done



Self Test

ECE 462

Object-Oriented Programming using C++ and Java

Declaration, Definition, and Initialization

Yung-Hsiang Lu
yunglu@purdue.edu

Declaration, Definition, Initialization

- declaration: inform the compiler the existence of an identifier (variable name, function name, class name ...)
- definition: for variable, memory is allocated; for function, code is written
- initialization: assign (default) value to variable
- You should **always** initialize all variables and objects. Uninitialized values are a common source for errors.
- Never leave anything uninitialized.

Declaration and Definition

`int xvar; // definition`

`int xvar = 4; // definition and initialization`

`extern int xvar; // declaration only in C++
// defined somewhere else`

`class Student; // declaration in C++`

`class Student { // definition`

`int s_id;`

`int s_year;`

`};`

`double func(int, float); // declaration`

`double func(int xval, float yval)`

`{ return (exp(yval) * xval); } // definition`

Declaration before Definition

```
class Student; // declaration in C++
class Teacher {
    list<Student*> t_stud; // list<Student> causes error since
                          // Student has not been defined
};
class Student {
    int s_id;
};
```

C ++ initialization:

- primitive types: default value
- objects: if **no-arg constructor** provided

Java and C++

Java	C++
<code>int x; // declare</code>	declare and define
<code>int x = 6; // declare, define, and // initialize to 6</code>	same
<code>AClass anobj; // declare // null</code>	declare, define, and initialize using no-arg constructor
syntax error	<code>AClass * objptr; // declared</code>
<code>AClass anobj = new AClass(...) // declare, define, initialized</code>	<code>AClass * objptr = new AClass(...); // same</code>
unnecessary	<code>class AClass; // declaration</code>

Default Initialization

C++		Java	
bool	false	boolean	false
char	'\0'	char	\u0000
int	0	int	0
float	0.0	float	0.0f
object pointer	-	not applicable	
object	no-arg constructor	object	null

no-arg constructor

```
class ClassName {  
public:          ↓  
    ClassName() {  
        // no-arg constructor  
    }  
}
```

- In C++, if no constructor is provided, the compiler will automatically provide a no-arg constructor (**probably not what you want**)
- If any programmer-provided constructor exists, the compiler does not add any constructor.

The screenshot shows the Eclipse IDE with a C++ project named 'objectinitialization'. The main.cpp file is open, showing the following code:

```
//DefaultInit.cc
#include <iostream>
#include <string>
using namespace std;
class User {
public:
    string name;
    int age;
    User() { name = "John Doe"; age = 25; } // (B)
};

int main()
{
    //object u defined and initialized :
    User u; // (C)
    cout << u.name << " " << u.age << endl; // John Doe 25
    //object *p defined and initialized:
    User* p = new User(); // (D)
    cout << p->name << " " << p->age << endl; // John Doe 25
    return 0;
}
```

A yellow arrow points to the `User()` constructor definition in the class, which is annotated with `// (B)`. The text 'no-arg constructor' is written above the arrow. The console output at the bottom shows the program's execution results:

```
<terminated> objectinitialization.exe [C/C++ Local Application] C:\yunglu\eclipse\workspace\objectinitialization\Debug\objectinitialization.exe (5/1/08 7:54 PM)
John Doe 25
John Doe 25
```

The status bar at the bottom indicates 'Writable', 'Smart Insert', '20 : 2', and 'C/C++ Indexer'.

The screenshot shows the Eclipse IDE with the following code in `DefaultInit.java`:

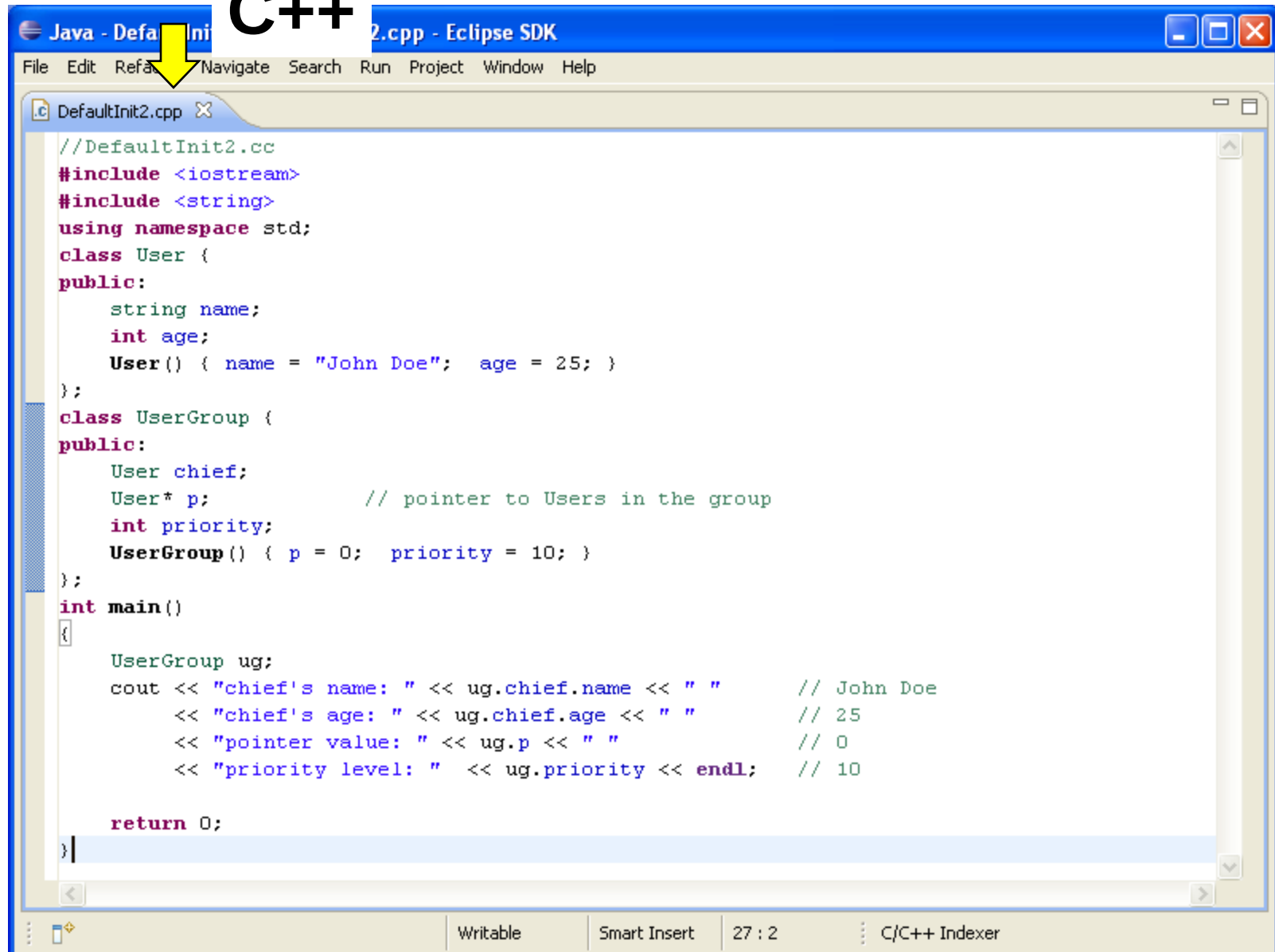
```
//DefaultInit.java
package defaultinit;
class User {
    private String name;
    private int age;
    public User() { name = "John Doe"; age = 25; }
    public String toString() { return name + " " + age; }
}
class Test {
    public static void main(String[] args) {
        // u1 declared but not defined:
        User u1; // (A)
        // System.out.println( u1 ); // ERROR //(B)
        // u2 defined and initialized:
        User u2 = new User(); // (C)
        System.out.println(u2); // John Doe 25 //(D)
    }
}
```

The console output is:

```
<terminated> Test [Java Application] C:\Program Files\java\jre1.6.0_03\bin\javaw.exe (May 2, 2008 10:25:15 PM)
John Doe 25
```

Java

C++



```
//DefaultInit2.cc
#include <iostream>
#include <string>
using namespace std;
class User {
public:
    string name;
    int age;
    User() { name = "John Doe"; age = 25; }
};
class UserGroup {
public:
    User chief;
    User* p;           // pointer to Users in the group
    int priority;
    UserGroup() { p = 0; priority = 10; }
};
int main()
{
    UserGroup ug;
    cout << "chief's name: " << ug.chief.name << " "      // John Doe
         << "chief's age: " << ug.chief.age << " "        // 25
         << "pointer value: " << ug.p << " "              // 0
         << "priority level: " << ug.priority << endl;    // 10

    return 0;
}
```



Java

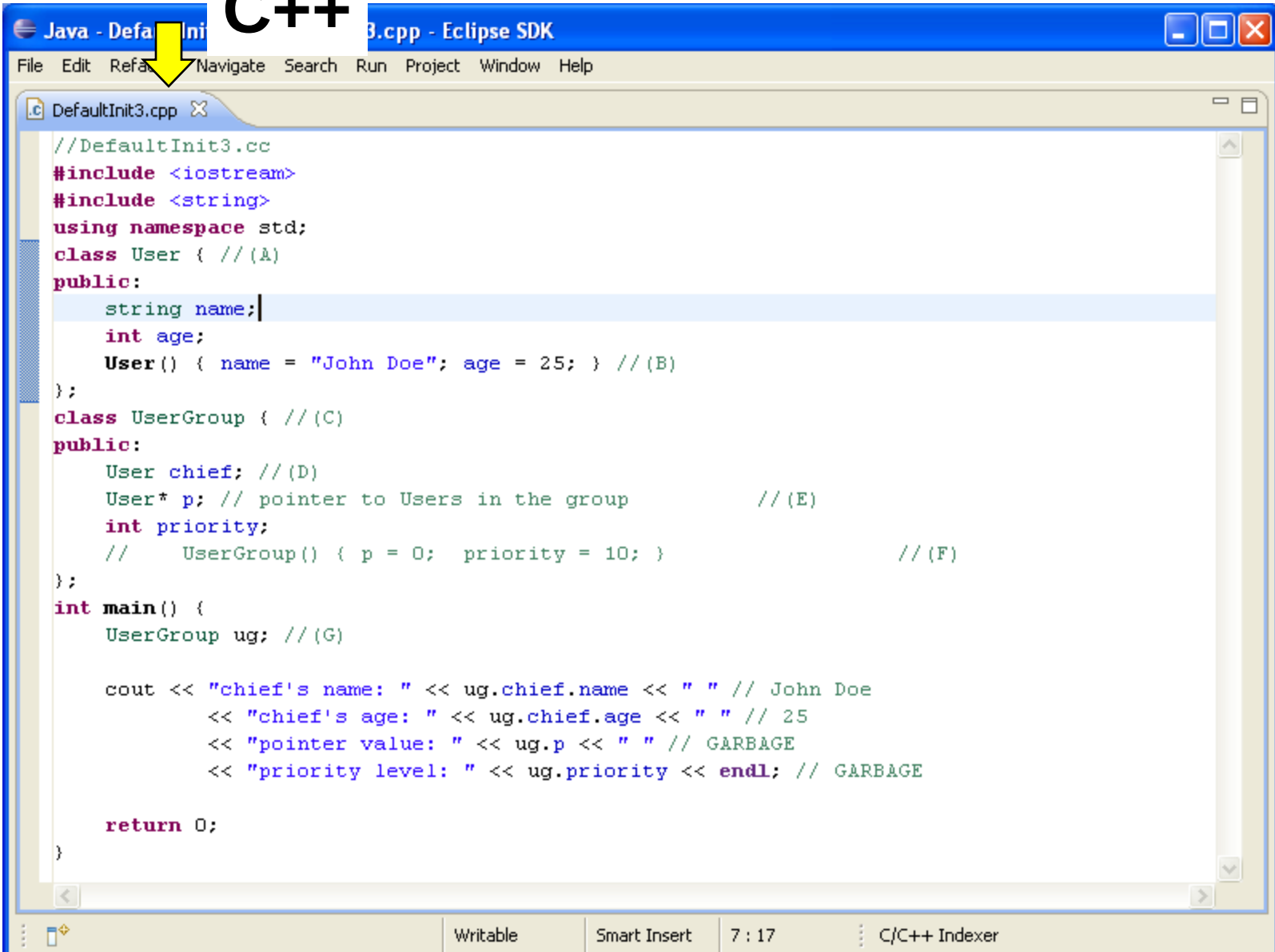
```
package defaultinit;
//DefaultInit2.java
class User {
    public String name;
    public int age;
    public User() { name = "John Doe"; age = 25; } // (A)
    public String toString() { return name + " " + age; } // (B)
}

class UserGroup {
    public String groupName;
    public User chief;
    public int priority;
    public User[] members;

    public String toString() { // (C)
        return groupName + " " + chief + " " + priority + " " + members;
    }
}

class Test {
    public static void main(String[] args) {
        UserGroup ug = new UserGroup(); // (D)
        System.out.println(ug); // null null 0 null // (E)
    }
}
```

C++



```
//DefaultInit3.cc
#include <iostream>
#include <string>
using namespace std;
class User { //(A)
public:
    string name;
    int age;
    User() { name = "John Doe"; age = 25; } //(B)
};
class UserGroup { //(C)
public:
    User chief; //(D)
    User* p; // pointer to Users in the group          //(E)
    int priority;
    //    UserGroup() { p = 0; priority = 10; }          //(F)
};
int main() {
    UserGroup ug; //(G)

    cout << "chief's name: " << ug.chief.name << " " // John Doe
         << "chief's age: " << ug.chief.age << " " // 25
         << "pointer value: " << ug.p << " " // GARBAGE
         << "priority level: " << ug.priority << endl; // GARBAGE

    return 0;
}
```

```
Java - DefaultInit3.java - Eclipse SDK
File Edit Source Run Project Window Help

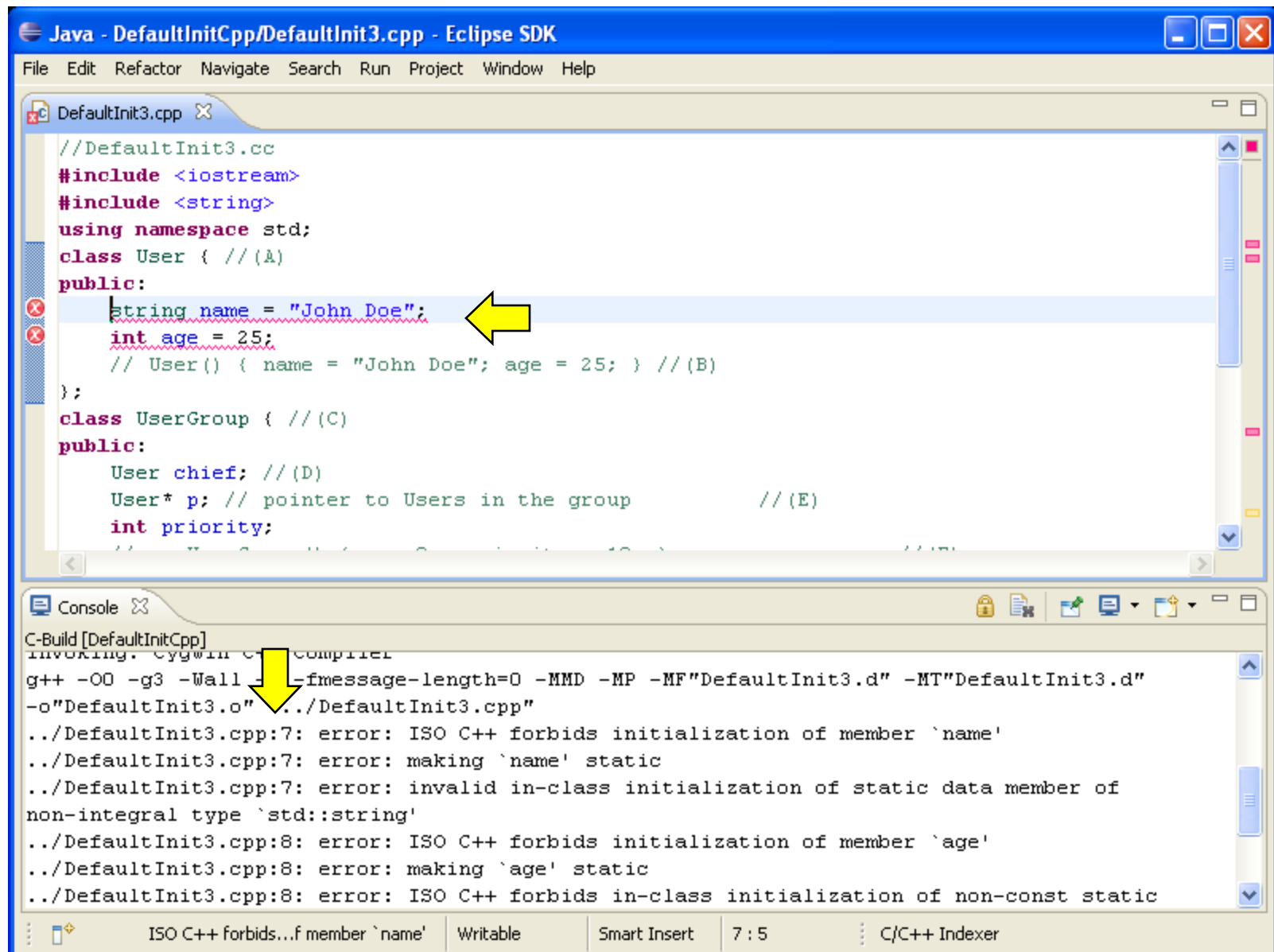
DefaultInit3.java
package defaultinit;

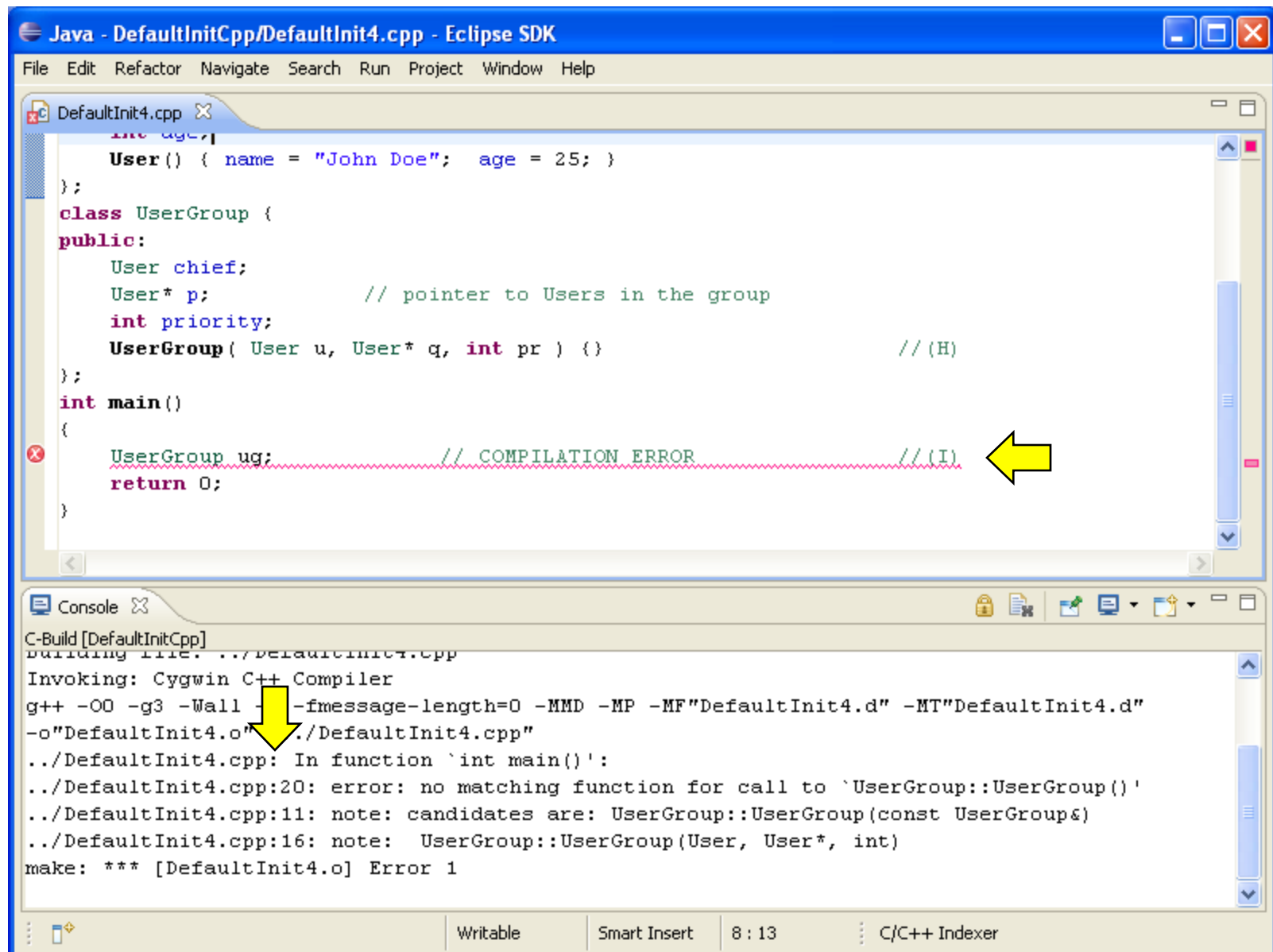
//DefaultInit3.java

class User {
    public String name = "John Doe"; // (A)
    public int age = 25; // (B)

    public String toString() {
        return name + " " + age;
    }
}

class Test {
    public static void main(String[] args) {
        User u = new User(); // (C)
        System.out.println(u); // John Doe 25 // (D)
    }
}
```





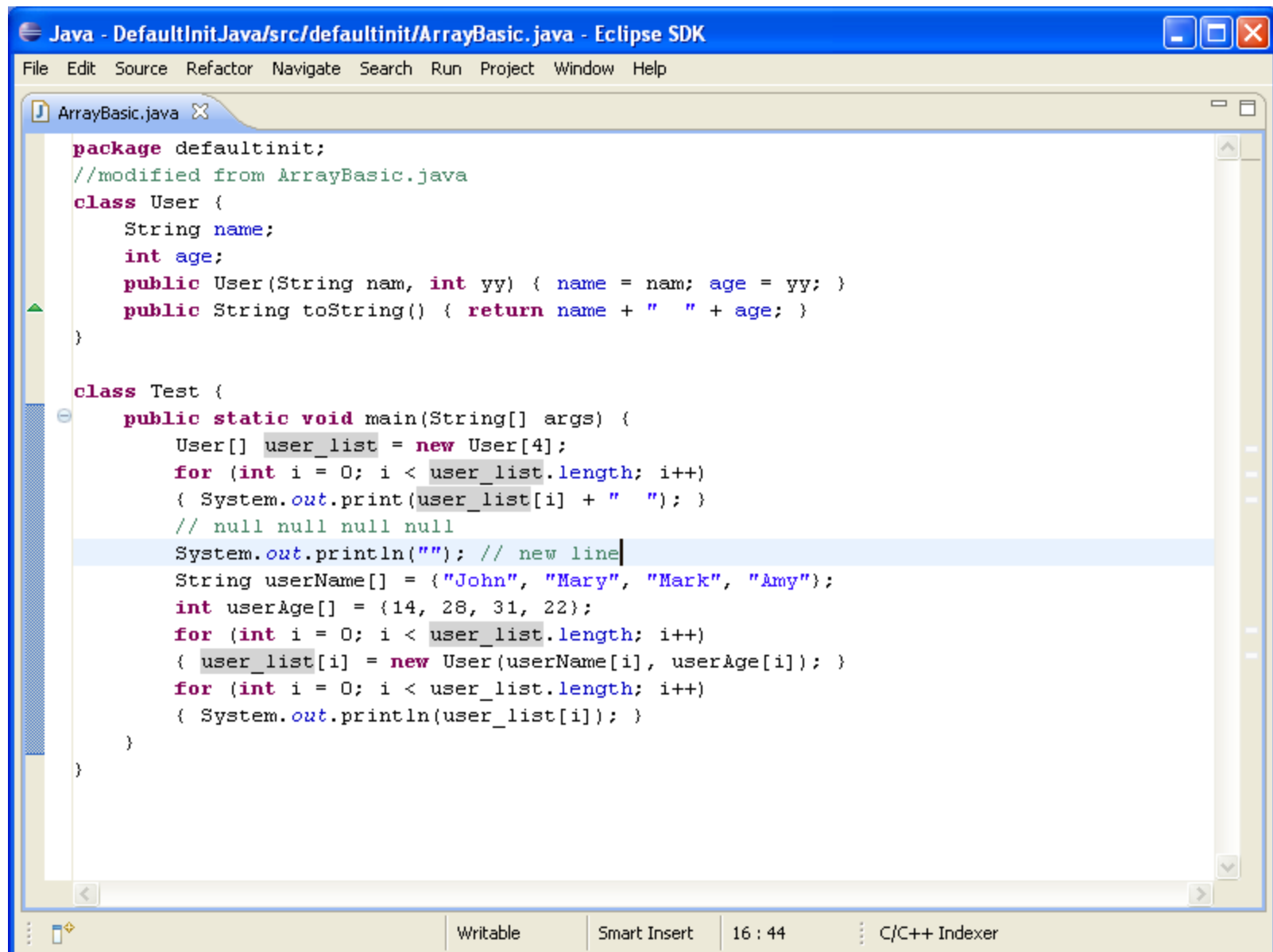
Array in C++

```
Java - DefaultInitCpp/DefaultInitClassArray2.cpp - Eclipse SDK
File Edit Refactor Navigate Search Run Project Window Help

DefaultInitClassArray2.cpp
using namespace std;
class Date {
public:
    int d, m, y;
    Date() { d = 1; m = 1; y = 1970; }
};
class User {
public:
    string name;
    int age;
    Date dateOfBirth;
    User() { name = "Zaphod"; age = 10; }
};
User uGlobal[10];           // global array
int main()
{
    User uLocal[10];        // local array
    cout << uLocal[1].name << endl;           // Zaphod
    cout << uLocal[1].age << endl;             // 10
    cout << uLocal[1].dateOfBirth.y << endl;   // 1970
    cout << uLocal[1].dateOfBirth.m << endl;   // 1

    cout << uGlobal[1].name << endl;           // Zaphod
    cout << uGlobal[1].age << endl;             // 10
    cout << uGlobal[1].dateOfBirth.y << endl;   // 1970
    cout << uGlobal[1].dateOfBirth.m << endl;   // 1
    return 0;
}
```

Array in Java

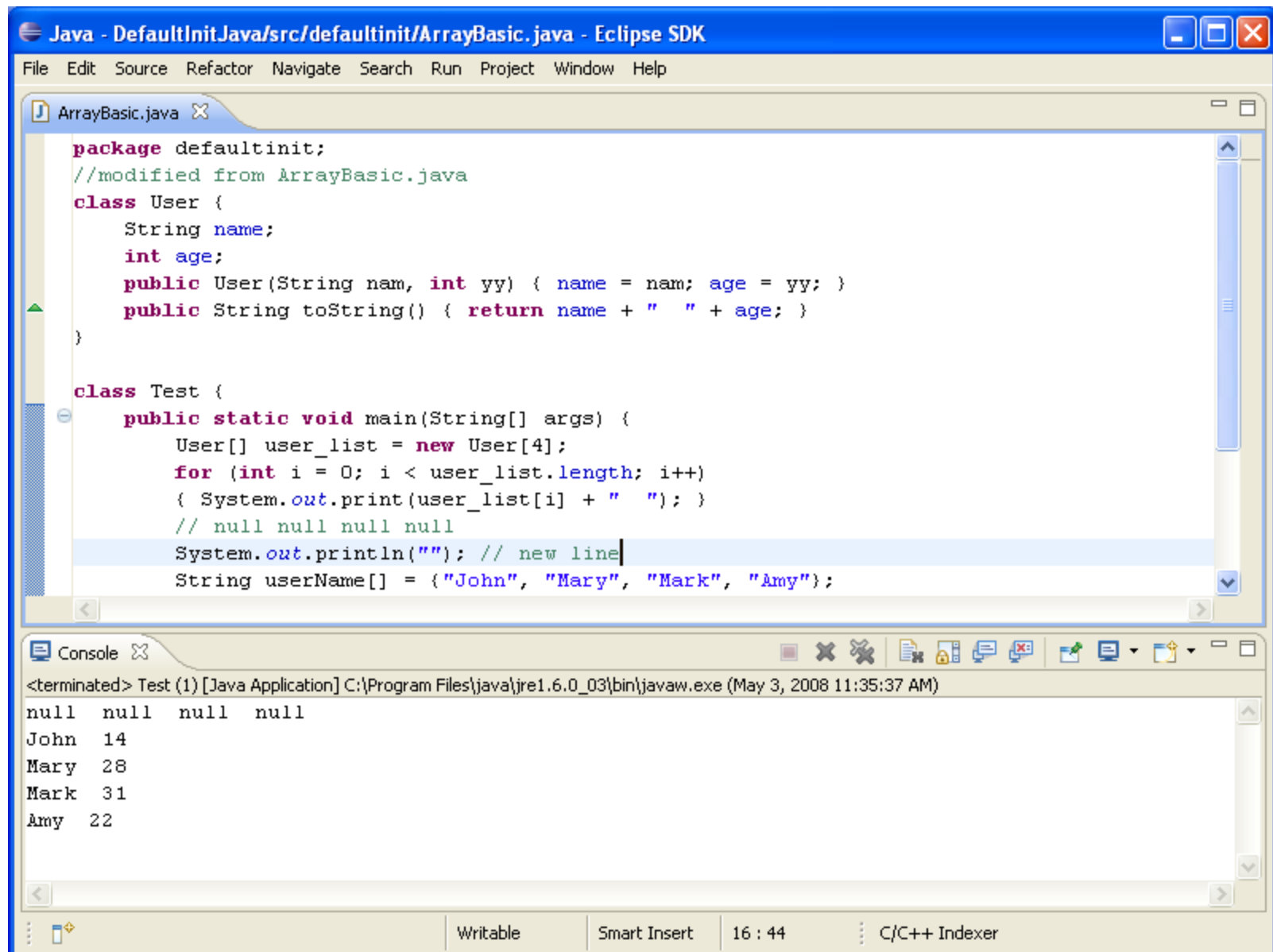


```
Java - DefaultInitJava/src/defaultinit/ArrayBasic.java - Eclipse SDK
File Edit Source Refactor Navigate Search Run Project Window Help

ArrayBasic.java
package defaultinit;
//modified from ArrayBasic.java
class User {
    String name;
    int age;
    public User(String nam, int yy) { name = nam; age = yy; }
    public String toString() { return name + " " + age; }
}

class Test {
    public static void main(String[] args) {
        User[] user_list = new User[4];
        for (int i = 0; i < user_list.length; i++)
        { System.out.print(user_list[i] + " "); }
        // null null null null
        System.out.println(""); // new line
        String userName[] = {"John", "Mary", "Mark", "Amy"};
        int userAge[] = {14, 28, 31, 22};
        for (int i = 0; i < user_list.length; i++)
        { user_list[i] = new User(userName[i], userAge[i]); }
        for (int i = 0; i < user_list.length; i++)
        { System.out.println(user_list[i]); }
    }
}
```

Writable Smart Insert 16 : 44 C/C++ Indexer



Self Test