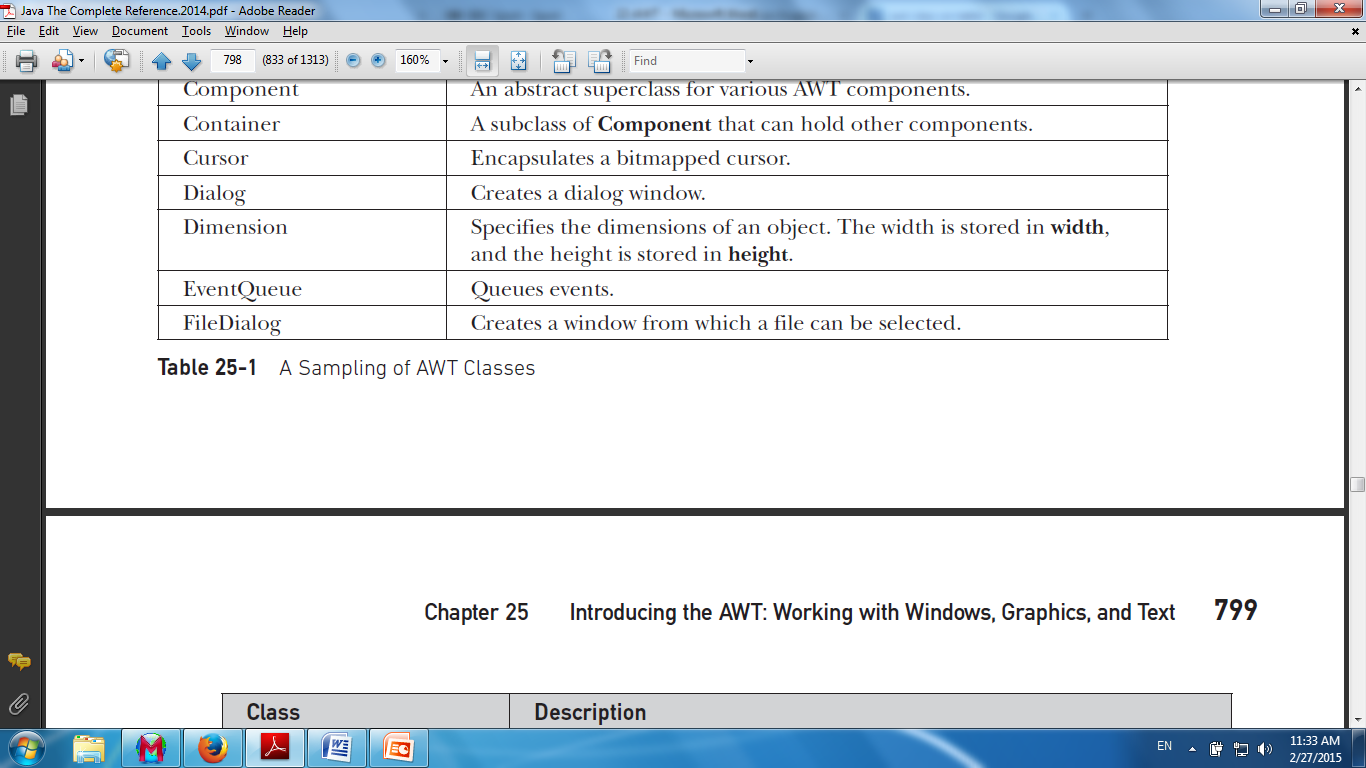
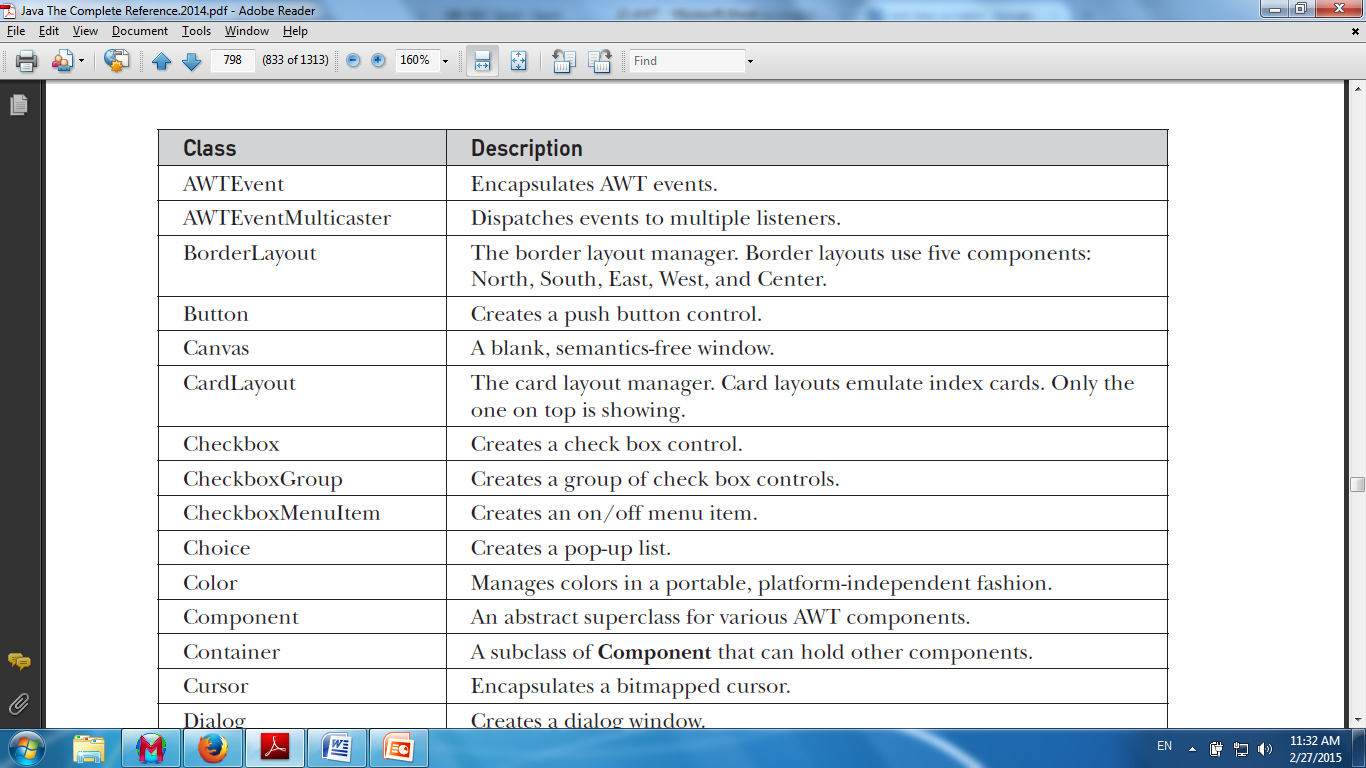
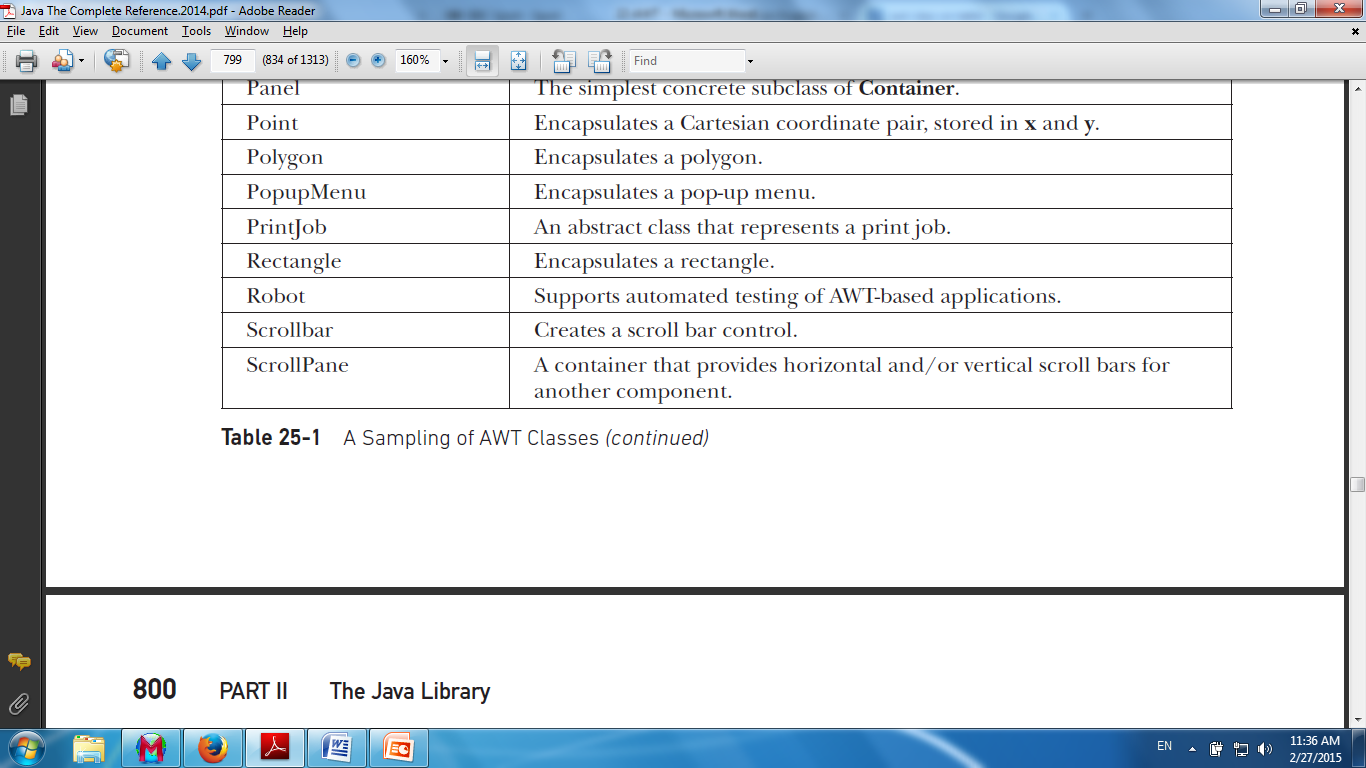
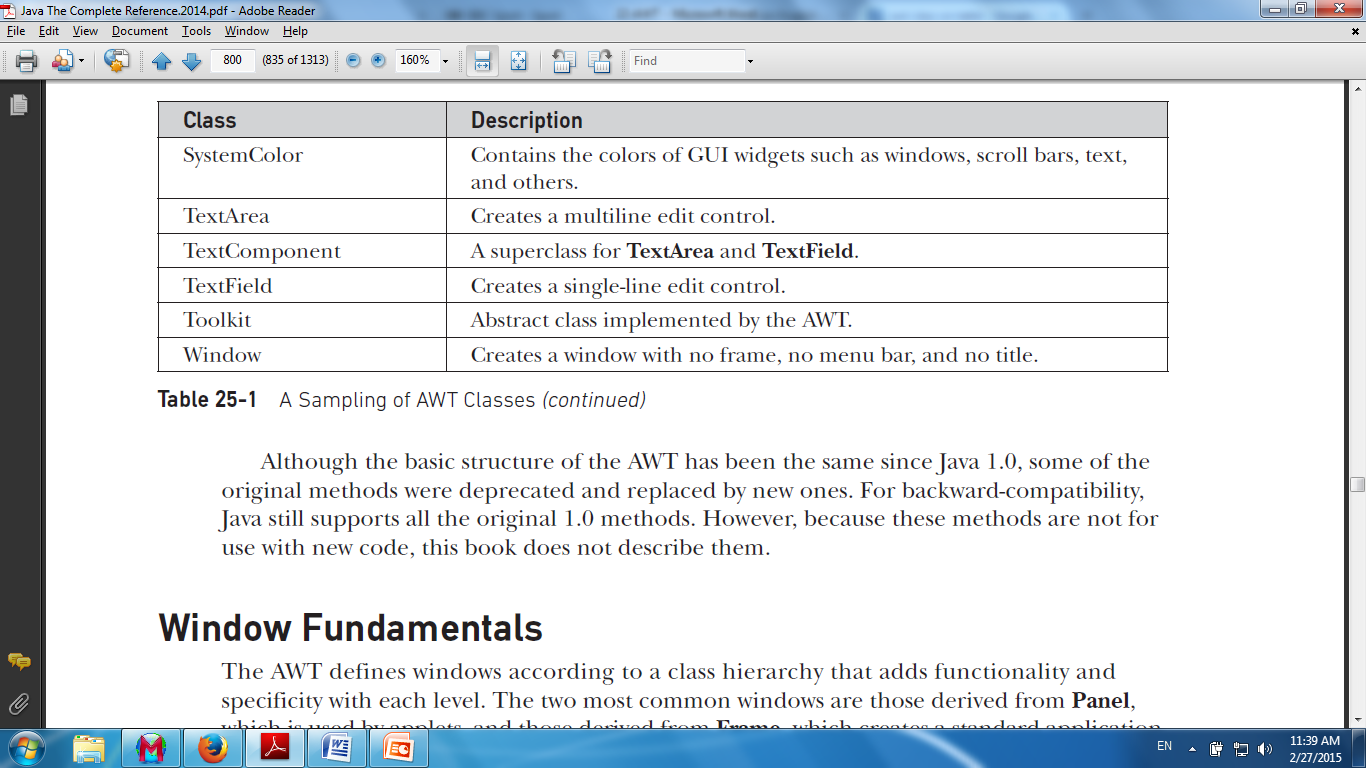
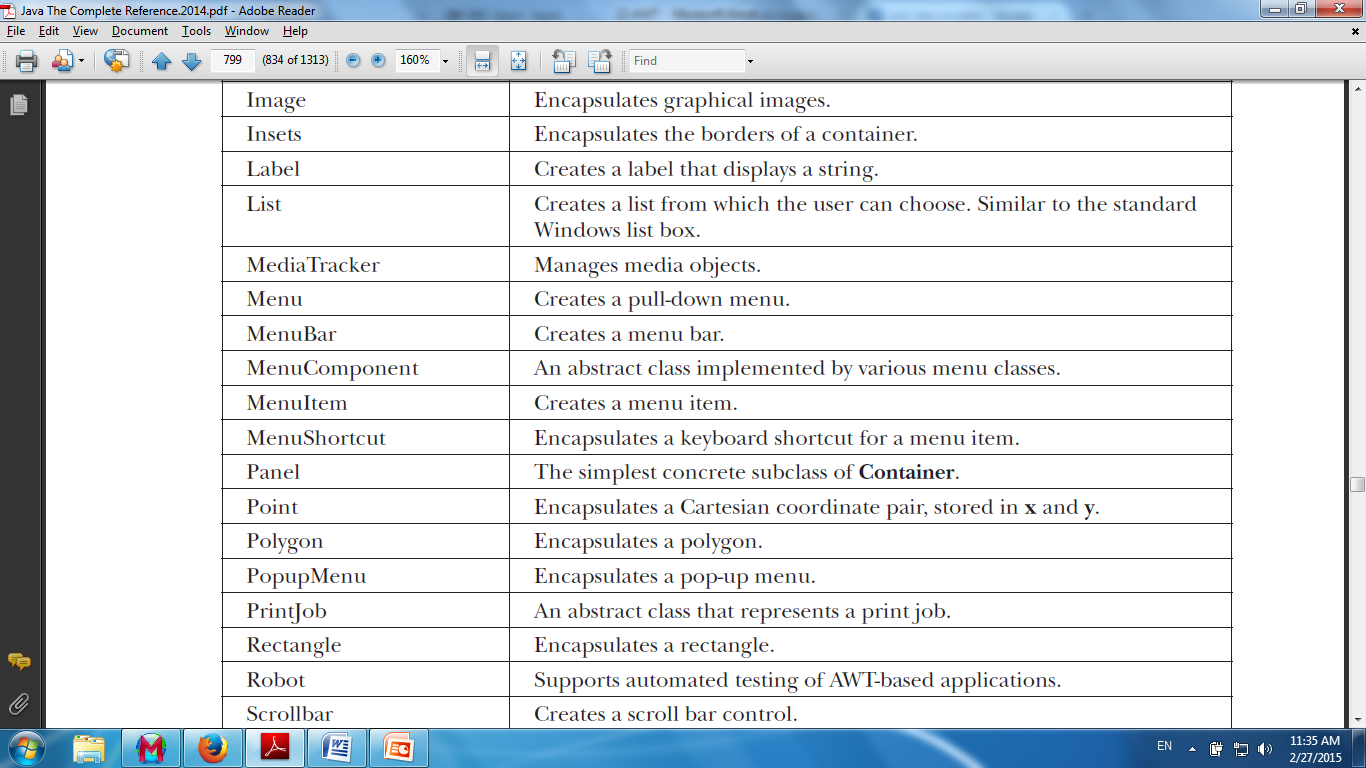
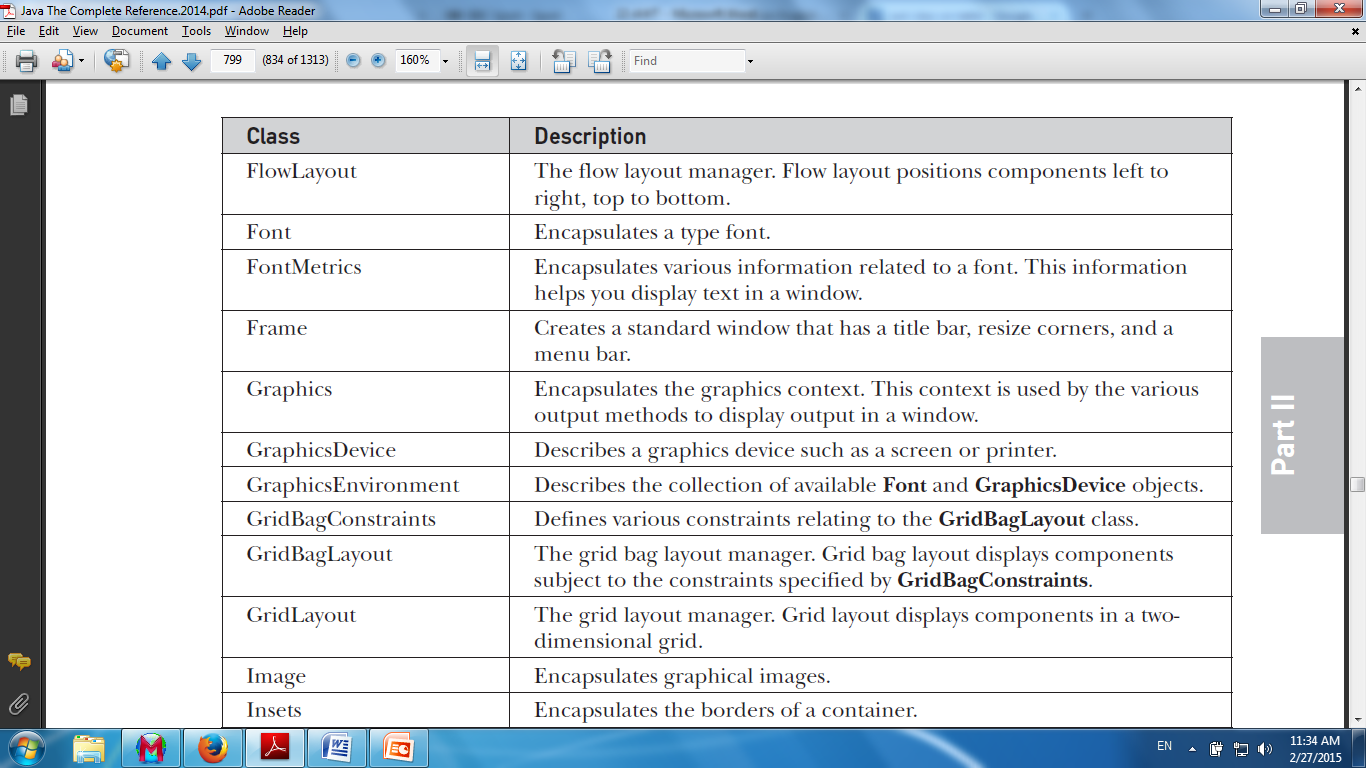
**AWT (Abstract Window Toolkit) –н тухай ба цонхтой ажиллах**

AWT нь жава хэлний анхын график дүрслэлтийг хийхэд ашигладаг байсан ба жава хэлний анхны 1.0 хувилбараас гарсан ойлголт юм. Энэ нь цонхнууд болон энгийн удирдлага үүсгэхэд зориулсан олон тооны класс, функцүүдийг агуулдаг. Бид эхний ээлжинд AWT дээр суурилан график дүрслэлийг хийх нь маш чухал юм. Яагаад гэвэл маш хүчирхэг график дүрслэлийг **Swing** болон **JavaFX** –г ашиглаж жава дээр хөгжүүлдэг. **Swing** –нь илүү баялаг нөөцтэй, уян хатан маш өргөн хэрэглэгддэг фреймворк юм. **JavaFX** –нь болохоор жава хэлний сүүлийн үеийн хамгийн шинэ буюу Swing фреймворк –г орлох фреймворк юм. Энэ хоёр фреймворкыг одоо үзэхгүй ба **AWT** –н дараа үзэх юм. Энэ хоёр фреймворк –оос үл хамааран AWT –нь жавагийн чухал хэсэгт үлддэг. Яагаад гэвэл swing –дээр график дүрслэлийг хийж хөгжүүлэхэд AWT –н олон тооны классуудыг шууд болон шууд бусаар ашигладаг юм. AWT класс нь java.awt пакежд агуулагддаг юм. Энэ пакеж н жавагийн хамгийн том пакежуудын нэг юм.

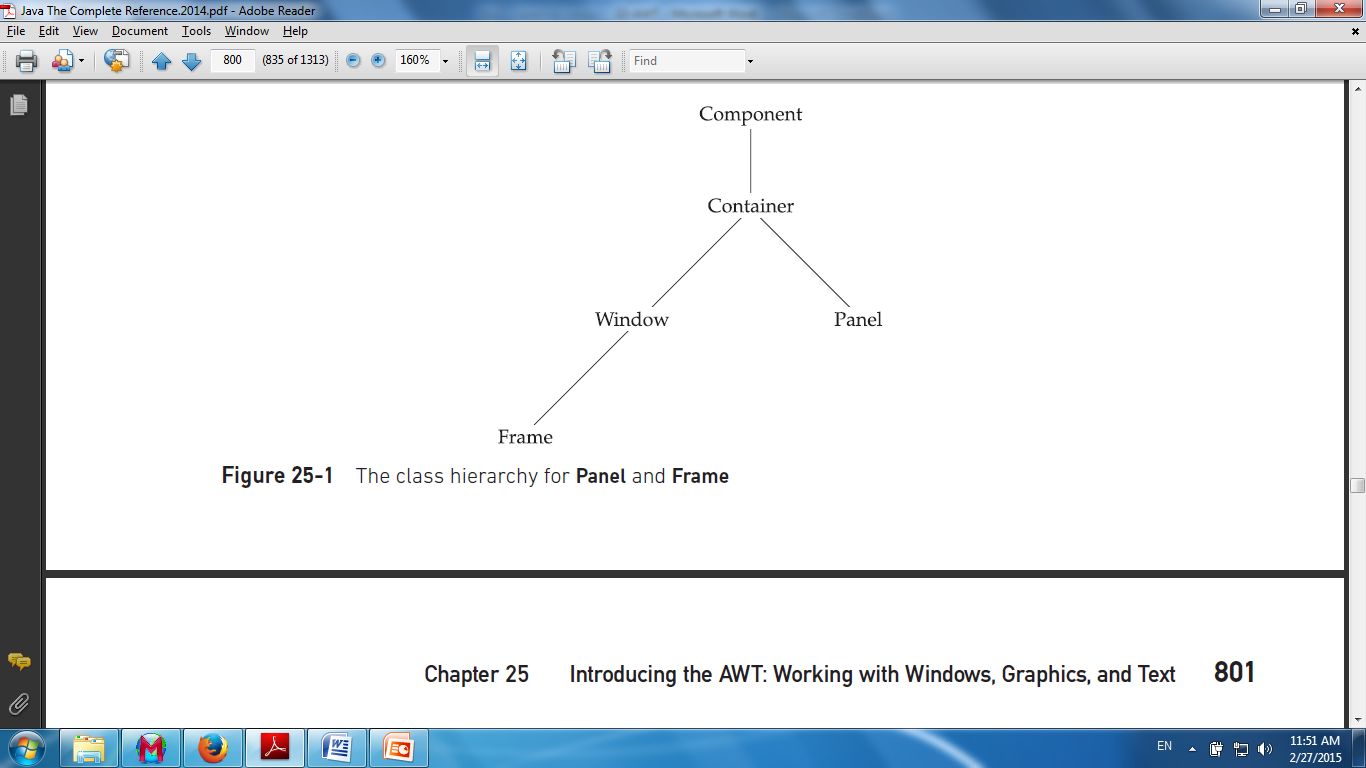
Доор AWT –н зарим классуудын жагсаалтыг үзүүллээ.



**Цонхны тухай үндсэн ойлголтууд**

AWT –нь цонхнуудыг классуудын шаталсан хэлбэрээр тухайн төвшин болгонд нэмж тодорхойлдог. Хамгийн өргөн хэрэглэгддэг хоёр цонх бол Panel болон Frame юм. Эдгээр цонхнуудын хамаарал нь цонхнуудын эцэг классаас үүсдэг. Тиймээс тухайн классуудын шаталсан харьцааг доор харуулав.



**Component:** AWT –н шаталсан харьцааны хамгийн дээд талд нь Component класс байдаг. Component бол хийсвэр (abstract) класс юм. Энэ класс нь менью –ээс бусад дэлгэцэнд дүрслэгдэх бүх объектуудыг агуулдаг юм.

**Conatainer:** Container класс бол Component –н дэд класс юм. Энэ нь бусад Component объектуудыг дотроо агуулах чадвартай ба Component классаас удамшсан функцүүд дээр нэмэлт функцүүдтэй.

**Panel:** Энэ нь component объектуудыг зохион байгуулахад хэрэглэгддэг Container –н бодит объектууд юм. Харин гарчигийн мөр, цэсний мөр, хүрээнүүдийг агуулдаггүй цонх юм. Энэ бас Applet –н эх класс юм. Бусад Component объектууд Panel классын өөрийнх нь add ( ) функцээр нэмдэг ба өөрөө тэдгээр component –туудыг байрлуулах болон хэмжээг нь өөрчлөх **setLocation( )** , **setPreferredSize( )**, **setSize( ), setBounds( )** гэсэн функцүүдтэй.

**Window:** Энэ нь дээд төвшний цонх үүсгэдэг класс. Энэ дээд төвшний цонхыг ямар нэгэн төрлийн бусад объектууд дотроо агуулдаггүй ба энэ нь шууд дэлгэцэн дээр зурагддаг. Ерөнхийдөө Window классын объектуудыг шууд үүсгэж болдоггүй үүний оронд энэ классын дэд класс болох Frame гэж дууддаг классыг ашиглан үүсгэж болно.

**Frame:** Энэ нь ерөнхийдөө Window класс шиг класс мөн Window классын дэд класс юм. Энэ нь дотроо гарчигийн мөр, цэсний мөр, хүрээтэй байдаг. Тиймээс бид график дүрслэлийг хийхэд энэ классыг өргөн хэрэглэх болно.

**Frame цонхуудтай ажиллах:**

Энэ класс нь хоёр байгуулагч функцтэй:

**Frame( ) throws HeadlessException**

**Frame(String *title*) throws HeadlessException**

Энхний хэлбэрийн байгуулагч функц нь гарчиггүй стандарт цонх үүсгэдэг. Хоёр дахь хэлбэрийн байгуулагч функц нь гарчигтай стандарт цонх үүсгэдэг. Анхааруулахад та цонхны хэмжигдэхүүнийг байгуулагч функцээр нь тодорхой тогтоож өгөх боломжгүй харин энэ классын өөрийнх нь функц болох **setSize( )** функцээр тавьж өгөх ёстой. Энэ класс нь дараах функцүүдтэй.

Цонхны хэмжээг тогтоох: **setSize( )** функцийг цонхын хэмжээг тогтооход хэрэглэгддэг ба доор бичигдэх хэлбэрийг нь харууллаа.

**void setSize(int *newWidth*, int *newHeight*)**

**void setSize(Dimension *newSize*)**

**getSize( )** функцийг цонхныхоо хэмжээг авхад хэрэглэдэг ба доор бичигдэх хэлбэрийг нь харууллаа.

**Dimension getSize( )**

**Цонхыг гаргах, нууцлах:** Цонхыг үүсгэсний дараа **setVisible( )** функцийг дуудах хүртэл тухайн цонх гарч ирэхгүй юм. Доор бичигдэх хэлбэрийг харууллаа.

**void setVisible(boolean *visibleFlag*)**

**Гарчигийг тодорхойлох:** Цонхны гарчигийг өөрчлөн өгхөд **setTitle( )** функцийг хэрэглэдэг.

**void setTitle(String *newTitle*)**

**Цонхыг хаах:** Frame цонхыг ашиглаж байхдаа та **setVisible( false )** гэж дуудна. Харин Window цонхыг хаахдаа та **WindowListener** интерфейсийн **windowClosing( )** функцийг имплемент хийх хэрэгтэй.

Жишээ: Applet –дээр суурилсан Frame цонх үүсгэх.

**import** java.awt.Frame;

**import** java.awt.Graphics;

**class** SampleFrame **extends** Frame{

SampleFrame(String title) {

**super**(title);

// create an object to handle window events

MyWindowAdapter adapter = **new** MyWindowAdapter(**this**);

// register it to receive those events

addWindowListener(adapter);

}

**public** **void** paint(Graphics g) {

g.drawString("This is in frame window", 10, 40);

}

}

**import** java.awt.event.WindowAdapter;

**import** java.awt.event.WindowEvent;

**class** MyWindowAdapter **extends** WindowAdapter{

SampleFrame sampleFrame;

**public** MyWindowAdapter(SampleFrame sampleFrame) {

**this**.sampleFrame = sampleFrame;

}

**public** **void** windowClosing(WindowEvent we) {

sampleFrame.setVisible(**false**);

}

}

**import** java.applet.Applet;

**import** java.awt.Frame;

**import** java.awt.Graphics;

**public** **class** AppletFrame **extends** Applet{

Frame f;

**public** **void** init() {

f = **new** SampleFrame("A Frame Window");

f.setSize(250, 250);

f.setVisible(**true**);

}

**public** **void** start() {

f.setVisible(**true**);

}

**public** **void** stop() {

f.setVisible(**false**);

}

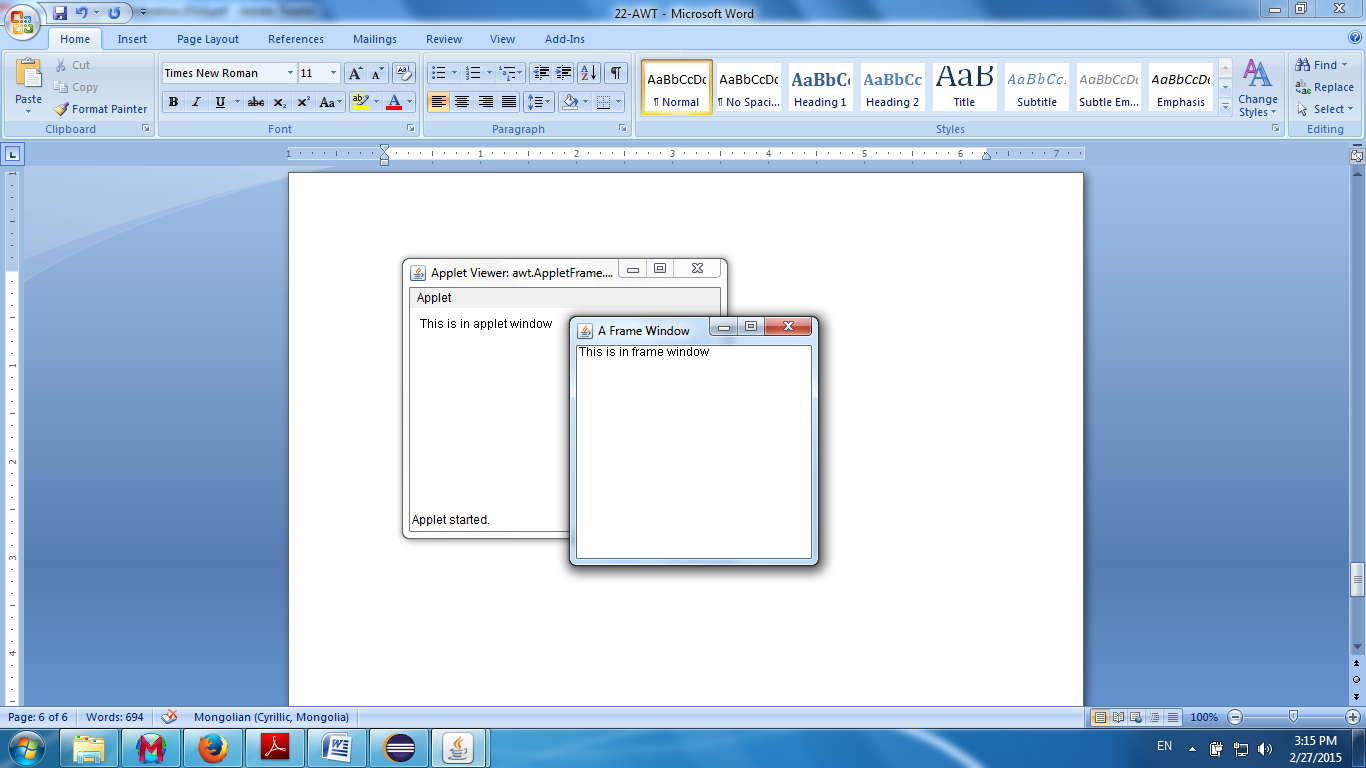
**public** **void** paint(Graphics g) {

g.drawString("This is in applet window", 10, 20);

}

}

**Үр дүн**



Жишээ: Frame цонхны үзэгдлүүдийг барьж авах

**package** awt1;

**import** java.awt.Frame;

**import** java.awt.Graphics;

**import** java.awt.event.MouseEvent;

**import** java.awt.event.MouseListener;

**import** java.awt.event.MouseMotionListener;

**class** SampleFrame **extends** Frame **implements** MouseListener, MouseMotionListener{

String msg = "";

**int** mouseX=10, mouseY=40;

**int** movX=0, movY=0;

SampleFrame(String title) {

**super**(title);

// register this object to receive its own mouse events

addMouseListener(**this**);

addMouseMotionListener(**this**);

// create an object to handle window events

MyWindowAdapter adapter = **new** MyWindowAdapter(**this**);

// register it to receive those events

addWindowListener(adapter);

}

// Handle mouse clicked.

**public** **void** mouseClicked(MouseEvent me) {

}

// Handle mouse entered.

**public** **void** mouseEntered(MouseEvent evtObj) {

// save coordinates

mouseX = 10;

mouseY = 54;

msg = "Mouse just entered child.";

repaint();

}

// Handle mouse exited.

**public** **void** mouseExited(MouseEvent evtObj) {

// save coordinates

mouseX = 10;

mouseY = 54;

msg = "Mouse just left child window.";

repaint();

}

// Handle mouse pressed.

**public** **void** mousePressed(MouseEvent me) {

// save coordinates

mouseX = me.getX();

mouseY = me.getY();

msg = "Down";

repaint();

}

// Handle mouse released.

**public** **void** mouseReleased(MouseEvent me) {

// save coordinates

mouseX = me.getX();

mouseY = me.getY();

msg = "Up";

repaint();

}

// Handle mouse dragged.

**public** **void** mouseDragged(MouseEvent me) {

// save coordinates

mouseX = me.getX();

mouseY = me.getY();

movX = me.getX();

movY = me.getY();

msg = "\*";

repaint();

}

// Handle mouse moved.

**public** **void** mouseMoved(MouseEvent me) {

// save coordinates

movX = me.getX();

movY = me.getY();

repaint(0, 0, 100, 60);

}

**public** **void** paint(Graphics g) {

g.drawString(msg, mouseX, mouseY);

g.drawString("Mouse at " + movX + ", " + movY, 10, 40);

}

}

**package** awt1;

**import** java.awt.event.WindowAdapter;

**import** java.awt.event.WindowEvent;

**class** MyWindowAdapter **extends** WindowAdapter{

SampleFrame sampleFrame;

**public** MyWindowAdapter(SampleFrame sampleFrame) {

**this**.sampleFrame = sampleFrame;

}

**public** **void** windowClosing(WindowEvent we) {

sampleFrame.setVisible(**false**);

}

}

**package** awt1;

**import** java.applet.Applet;

**import** java.awt.Graphics;

**import** java.awt.event.MouseEvent;

**import** java.awt.event.MouseListener;

**import** java.awt.event.MouseMotionListener;

**public** **class** WindowEvents **extends** Applet **implements** MouseListener, MouseMotionListener{

SampleFrame f;

String msg = "";

**int** mouseX=0, mouseY=10;

**int** movX=0, movY=0;

// Create a frame window.

**public** **void** init() {

f = **new** SampleFrame("Handle Mouse Events");

f.setSize(300, 200);

f.setVisible(**true**);

// register this object to receive its own mouse events

addMouseListener(**this**);

addMouseMotionListener(**this**);

}

// Remove frame window when stopping applet.

**public** **void** stop() {

f.setVisible(**false**);

}

// Show frame window when starting applet.

**public** **void** start() {

f.setVisible(**true**);

}

// Handle mouse clicked.

**public** **void** mouseClicked(MouseEvent me) {

}

// Handle mouse entered.

**public** **void** mouseEntered(MouseEvent me) {

// save coordinates

mouseX = 0;

mouseY = 24;

msg = "Mouse just entered applet window.";

repaint();

}

// Handle mouse exited.

**public** **void** mouseExited(MouseEvent me) {

// save coordinates

mouseX = 0;

mouseY = 24;

msg = "Mouse just left applet window.";

repaint();

}

// Handle button pressed.

**public** **void** mousePressed(MouseEvent me) {

// save coordinates

mouseX = me.getX();

mouseY = me.getY();

msg = "Down";

repaint();

}

// Handle button released.

**public** **void** mouseReleased(MouseEvent me) {

// save coordinates

mouseX = me.getX();

mouseY = me.getY();

msg = "Up";

repaint();

}

// Handle mouse dragged.

**public** **void** mouseDragged(MouseEvent me) {

// save coordinates

mouseX = me.getX();

mouseY = me.getY();

movX = me.getX();

movY = me.getY();

msg = "\*";

repaint();

}

// Handle mouse moved.

**public** **void** mouseMoved(MouseEvent me) {

// save coordinates

movX = me.getX();

movY = me.getY();

repaint(0, 0, 100, 20);

}

// Display msg in applet window.

**public** **void** paint(Graphics g) {

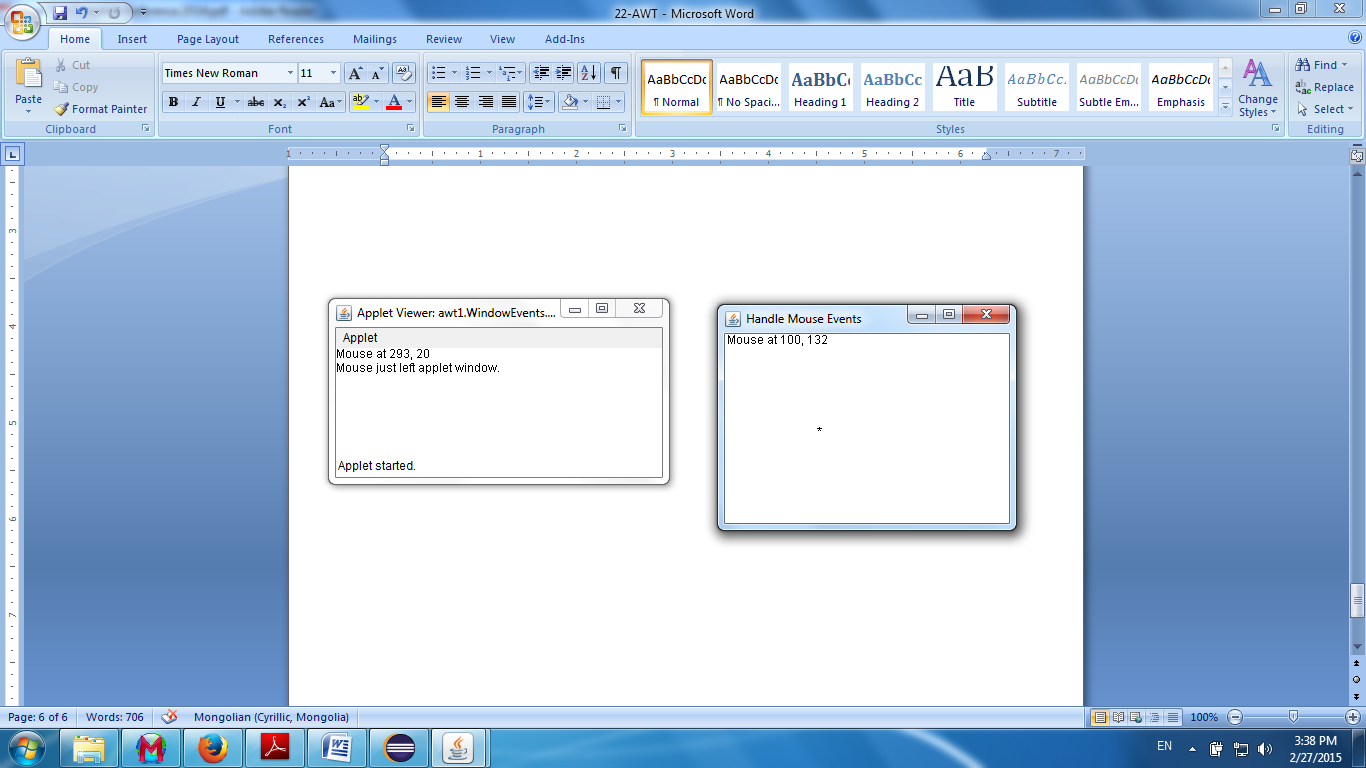
g.drawString(msg, mouseX, mouseY);

g.drawString("Mouse at " + movX + ", " + movY, 0, 10);

}

}

**Үр дүн**



Жишээ: Цонхтой програм үүсгэх

**package** awt1;

**import** java.awt.Dimension;

**import** java.awt.Frame;

**import** java.awt.Graphics;

**public** **class** AppWindow **extends** Frame {

String keymsg = "This is a test.";

String mousemsg = "";

**int** mouseX=30, mouseY=30;

**public** AppWindow() {

addKeyListener(**new** MyKeyAdapter(**this**));

addMouseListener(**new** MyMouseAdapter(**this**));

addWindowListener(**new** MyWindowAdapter());

}

**public** **void** paint(Graphics g) {

g.drawString(keymsg, 10, 40);

g.drawString(mousemsg, mouseX, mouseY);

}

// Create the window.

**public** **static** **void** main(String args[]) {

AppWindow appwin = **new** AppWindow();

appwin.setSize(**new** Dimension(300, 200));

appwin.setTitle("An AWT-Based Application");

appwin.setVisible(**true**);

}

}

**package** awt1;

**import** java.awt.event.KeyAdapter;

**import** java.awt.event.KeyEvent;

**class** MyKeyAdapter **extends** KeyAdapter{

AppWindow appWindow;

**public** MyKeyAdapter(AppWindow appWindow) {

**this**.appWindow = appWindow;

}

**public** **void** keyTyped(KeyEvent ke) {

appWindow.keymsg += ke.getKeyChar();

appWindow.repaint();

};

}

**package** awt1;

**import** java.awt.event.MouseAdapter;

**import** java.awt.event.MouseEvent;

**class** MyMouseAdapter **extends** MouseAdapter{

AppWindow appWindow;

**public** MyMouseAdapter(AppWindow appWindow) {

**this**.appWindow = appWindow;

}

**public** **void** mousePressed(MouseEvent me) {

appWindow.mouseX = me.getX();

appWindow.mouseY = me.getY();

appWindow.mousemsg = "Mouse Down at " + appWindow.mouseX +", " + appWindow.mouseY;

appWindow.repaint();

}

}

**package** awt1;

**import** java.awt.event.WindowAdapter;

**import** java.awt.event.WindowEvent;

**class** MyWindowAdapter **extends** WindowAdapter{

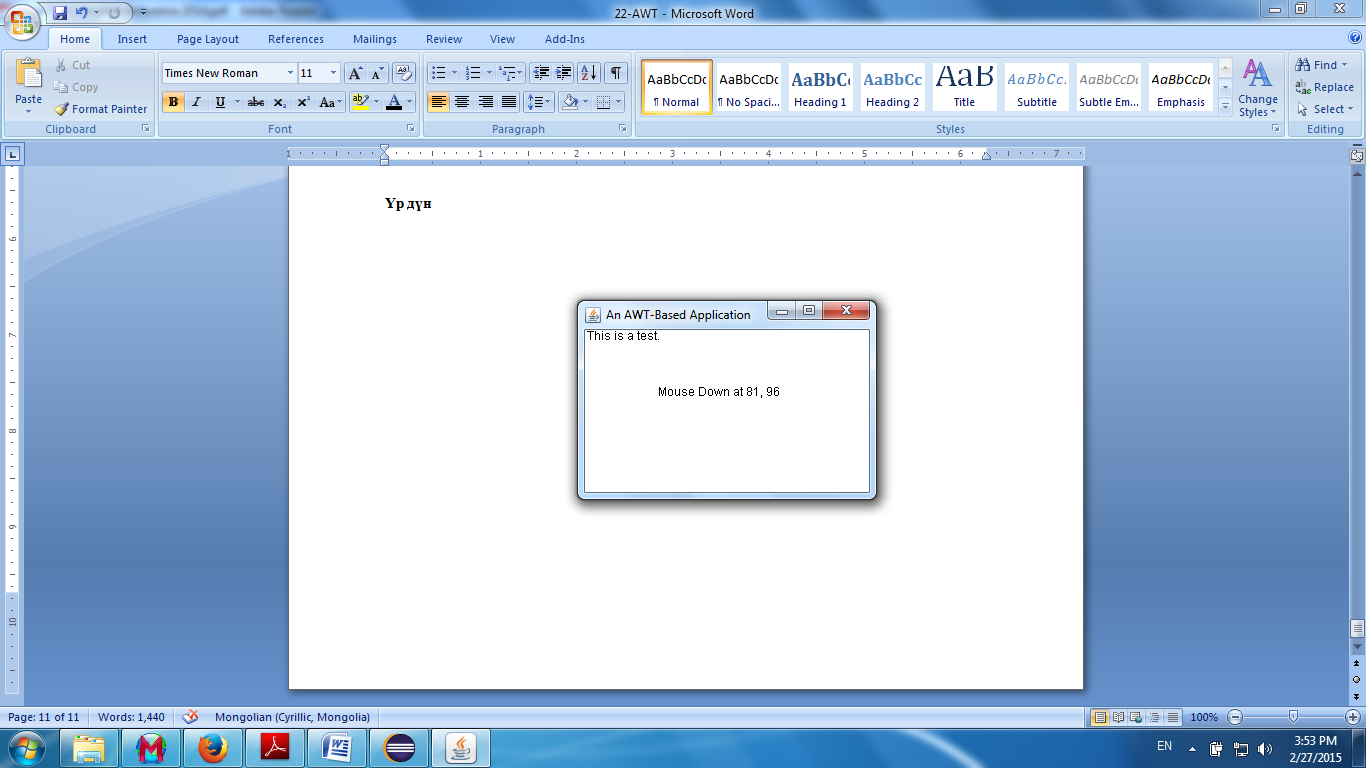
**public** **void** windowClosing(WindowEvent we) {

System.*exit*(0);

}

}

**Үр дүн**

****