**Name : Ni Gusti Putu Pamela Putri**

**NIM : 210040155**

**University : ITB STIKOM BALI**

1. **Version 1**

**import** java.awt.\*;

**public class** TankClient **extends Frame** {

**public void** launchFrame() {

**this**.setLocation(300, 100);

**this**.setSize(800, 600); setVisible(**true**);

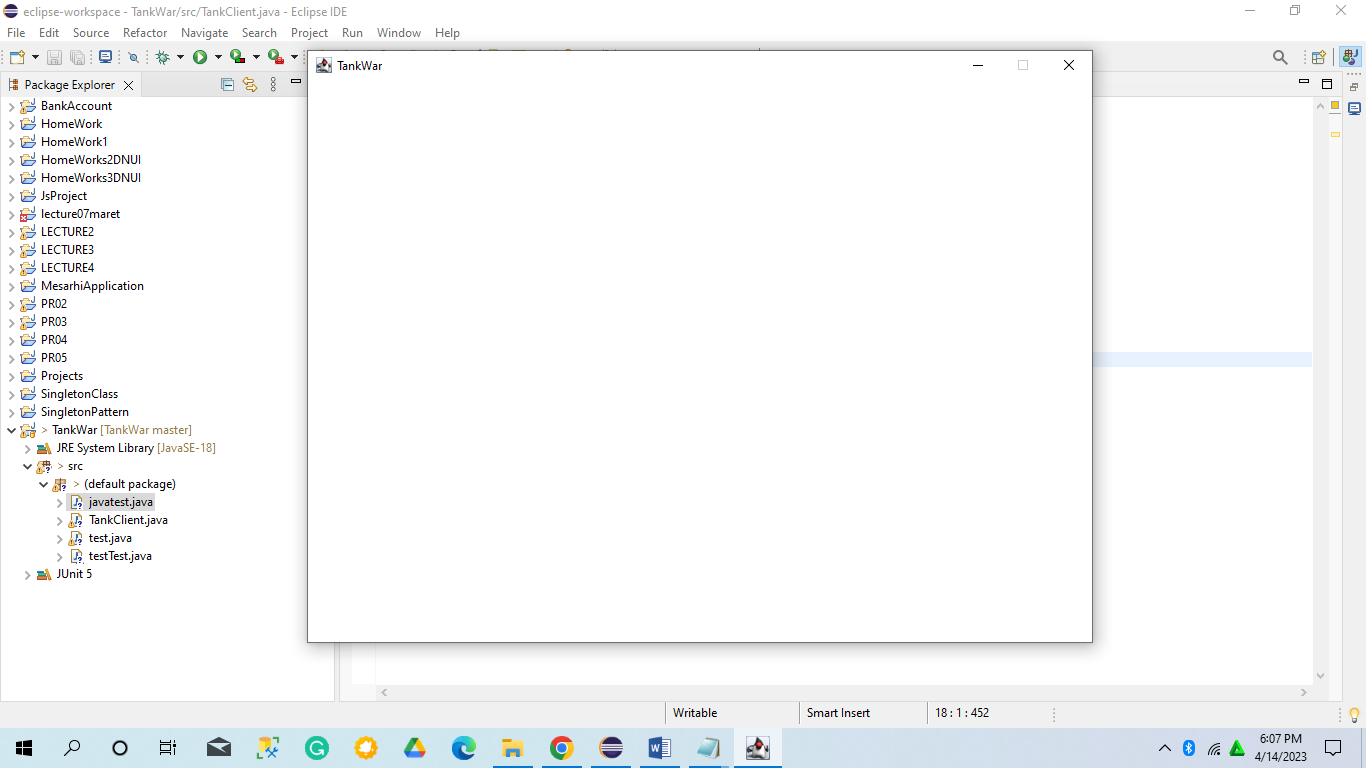
}

**public static void** main(String[] args) { TankClient tc = **new** TankClient(); tc.launchFrame();

}

}

Result::



1. **Version 2**

**import** java.awt.\*; import java.awt.event.\*;

**public class** TankClient **extends** Frame {

**public void** launchFrame() {

**this**.setLocation(300, 50);

**this**.setSize(800, 600); **this**.setTitle("TankWar"); **addWindowListener(new** **WindowAdapter()** **{**

**public void windowClosing(WindowEvent e) {** **System.*exit*(0);**

**}**

**});**

**setResizable(false);**

setVisible(**true**);

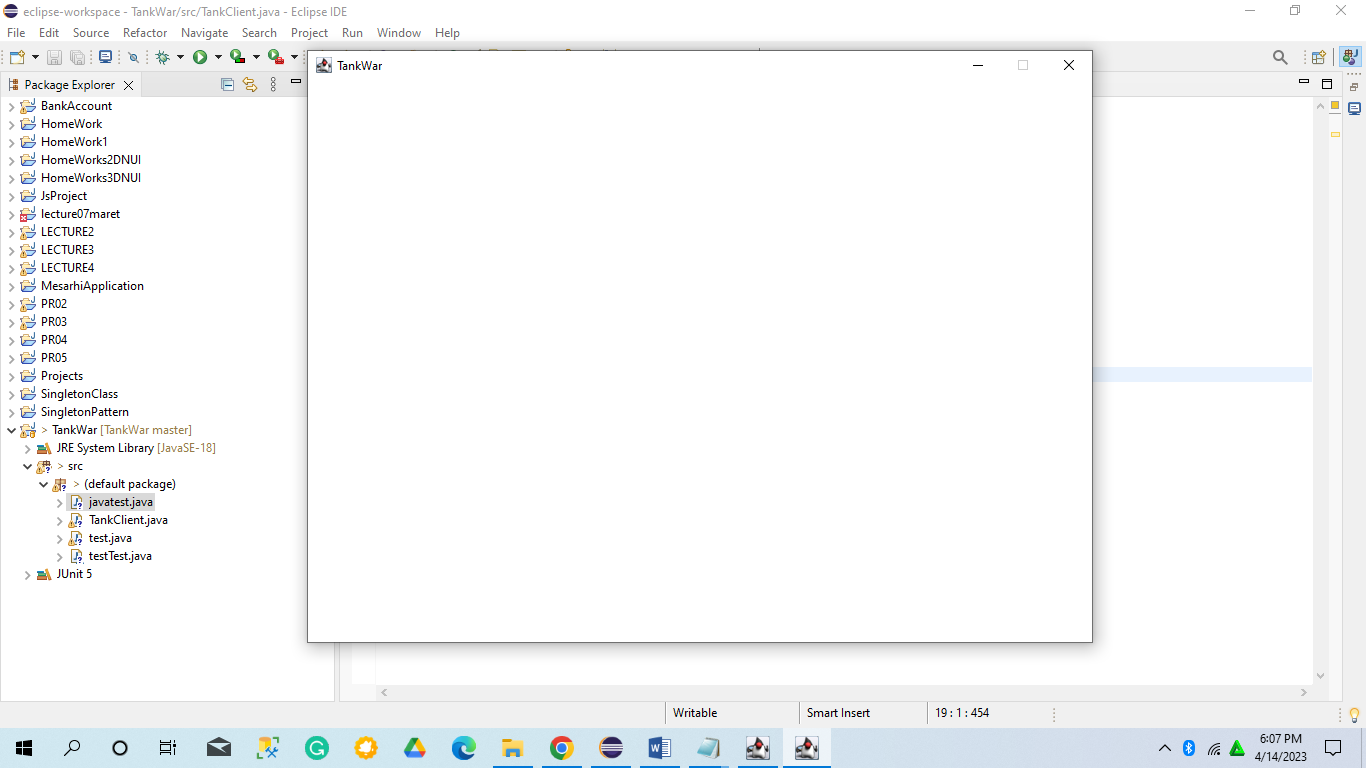
}

**public static void** main(String[] args) { TankClient tc = **new** TankClient(); tc.launchFrame();

}

}

Result:



1. **Version 3**

**import** java.awt.\*;

**import** java.awt.event.\*;

**public class** TankClient **extends** Frame {

//The paint method does not need to be called and will be automatically called once it is to be redrawn

**public void** paint(Graphics g) { Color c = g.getColor(); g.setColor(Color.*RED*); g.fillOval(50, 50, 30, 30); g.setColor(c);

}

**public void** launchFrame() {

**this**.setLocation(300, 50);

**this**.setSize(800, 600); **this**.setTitle("TankWar"); addWindowListener(**new** WindowAdapter() {

**public void** windowClosing(WindowEvent e) { System.*exit*(0);

}

});

**this**.setResizable(**false**); **this**.setBackground(Color.*GREEN*); setVisible(**true**);

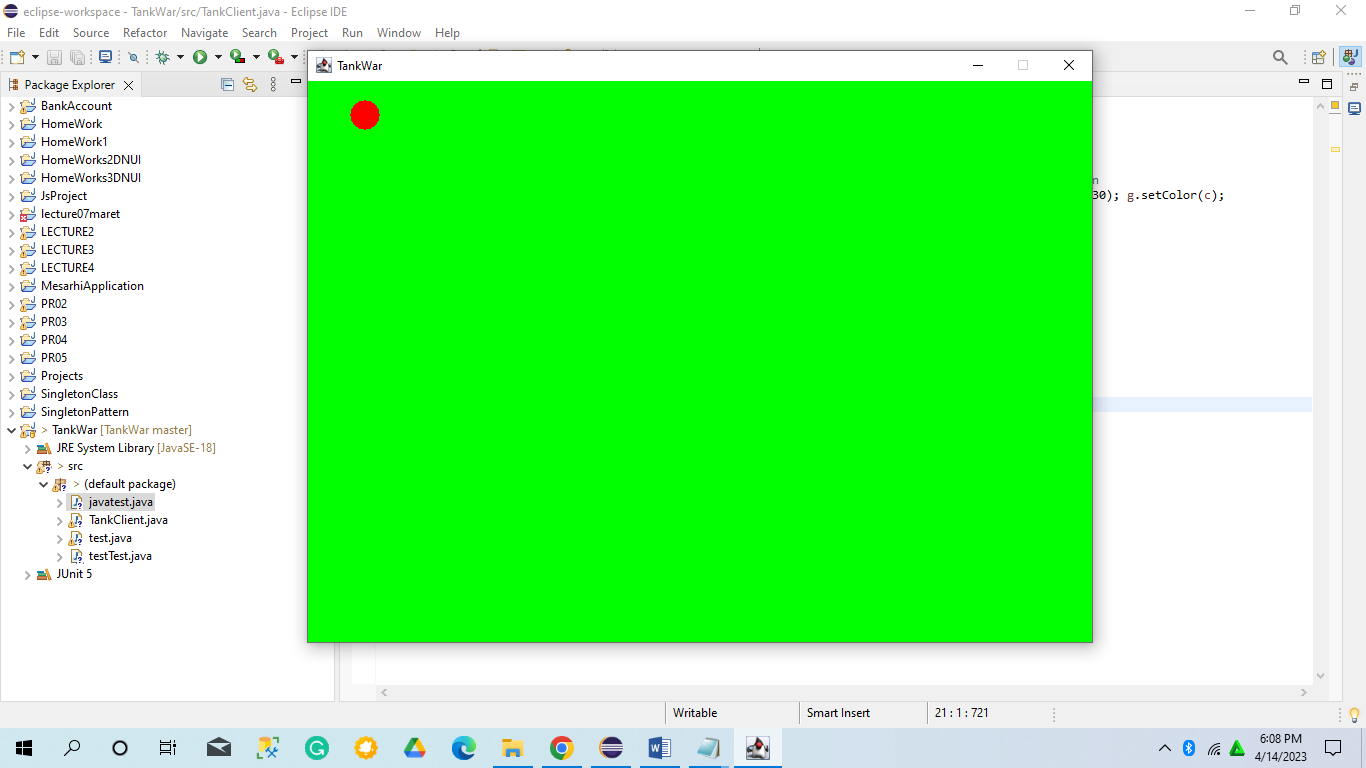
}

**public static void** main(String[] args) { TankClient tc = **new** TankClient(); tc.launchFrame();

}

}

Result



1. **Version 4**

**import** java.awt.\*;

**import** java.awt.event.\*;

**public class** TankClient **extends** Frame {

**int** x = 50, y = 50;

//The paint method does not need to be called and will be automatically called once it is to be redrawn

**public void** paint(Graphics g) { Color c = g.getColor(); g.setColor(Color.*RED*); g.fillOval(x, y, 30, 30); g.setColor(c);

y += 5;

}

**public void** launchFrame() {

**this**.setLocation(300, 50);

**this**.setSize(800, 600); **this**.setTitle("TankWar"); addWindowListener(**new** WindowAdapter() {

**public void** windowClosing(WindowEvent e) { System.*exit*(0);

}

});

**this**.setResizable(**false**); **this**.setBackground(Color.*GREEN*); setVisible(**true**);

**new** Thread(**new** PaintThread()).start();

}

**public static void** main(String[] args) { TankClient tc = **new** TankClient(); tc.launchFrame();

}

**private class** PaintThread **implements** Runnable {

**public void** run() {

**while**(**true**) { repaint(); **try** {

Thread.*sleep*(100);

} **catch** (InterruptedException e) { e.printStackTrace();

}

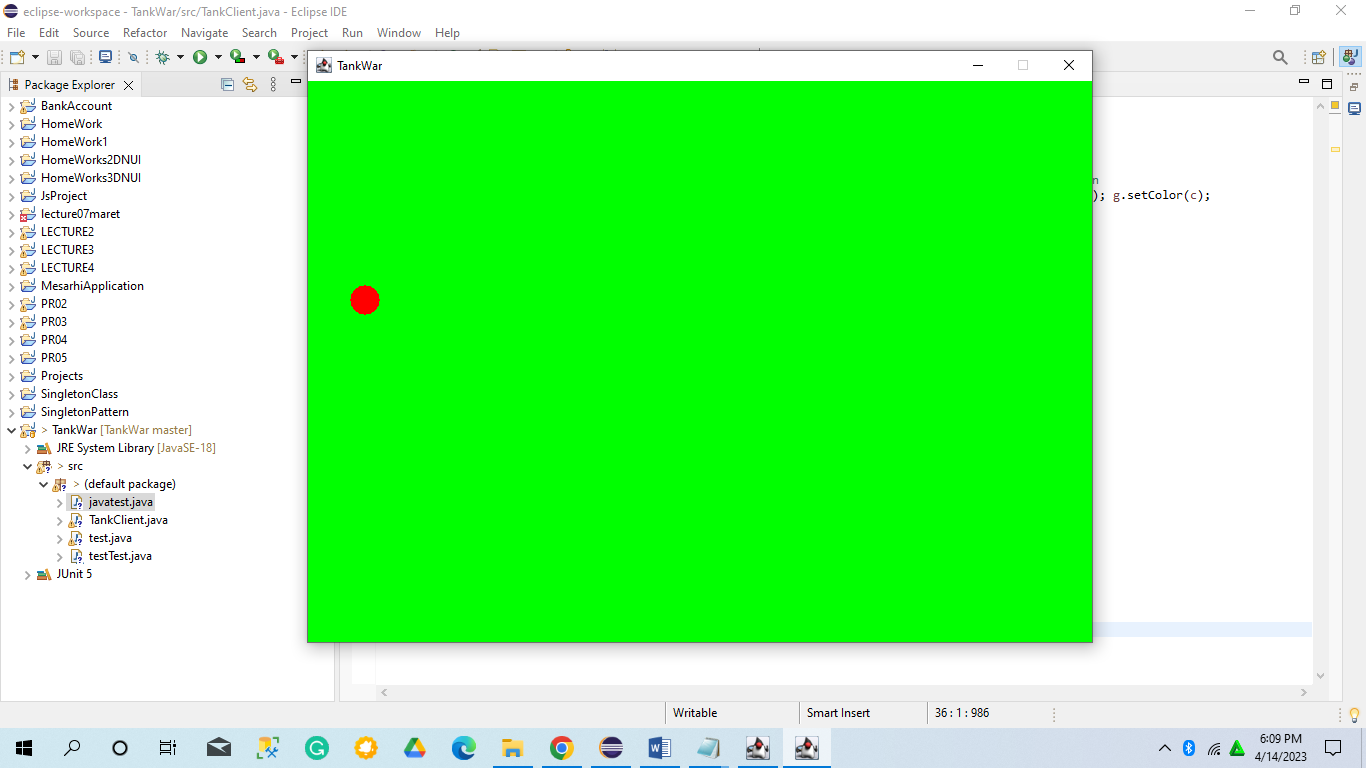
}

}

}

}

Result



1. **Version 41**

**import** java.awt.\*;

**import** java.awt.event.\*;

**public class** TankClient **extends** Frame {

**int** x = 50, y = 50;

//这是一张虚拟图片

Image offScreenImage = **null**;

//The paint method does not need to be called and will be automatically called once it is to be redrawn

**public void** paint(Graphics g) { Color c = g.getColor(); g.setColor(Color.*RED*); g.fillOval(x, y, 30, 30); g.setColor(c);

y += 5;

}

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

**if**(offScreenImage == **null**) {

offScreenImage = **this**.createImage(800, 600);

}

//拿到这个图片的画笔

Graphics gOffScreen = offScreenImage.getGraphics();

Color c = gOffScreen.getColor(); gOffScreen.setColor(Color.*GREEN*); gOffScreen.fillRect(0, 0, 800, 600); gOffScreen.setColor(c); print(gOffScreen); g.drawImage(offScreenImage, 0, 0, **null**);

}

**public void** launchFrame() {

**this**.setLocation(300, 50);

**this**.setSize(800, 600); **this**.setTitle("TankWar"); addWindowListener(**new** WindowAdapter() {

**public void** windowClosing(WindowEvent e) { System.*exit*(0);

}

});

**this**.setResizable(**false**); **this**.setBackground(Color.*GREEN*); setVisible(**true**);

**new Thread(new PaintThread()).start();**

}

**public static void** main(String[] args) { TankClient tc = **new** TankClient(); tc.launchFrame();

}

**private class** PaintThread **implements** Runnable {

**public void** run() {

**while**(**true**) { repaint(); **try** {

Thread.*sleep*(50);

} **catch** (InterruptedException e) { e.printStackTrace();

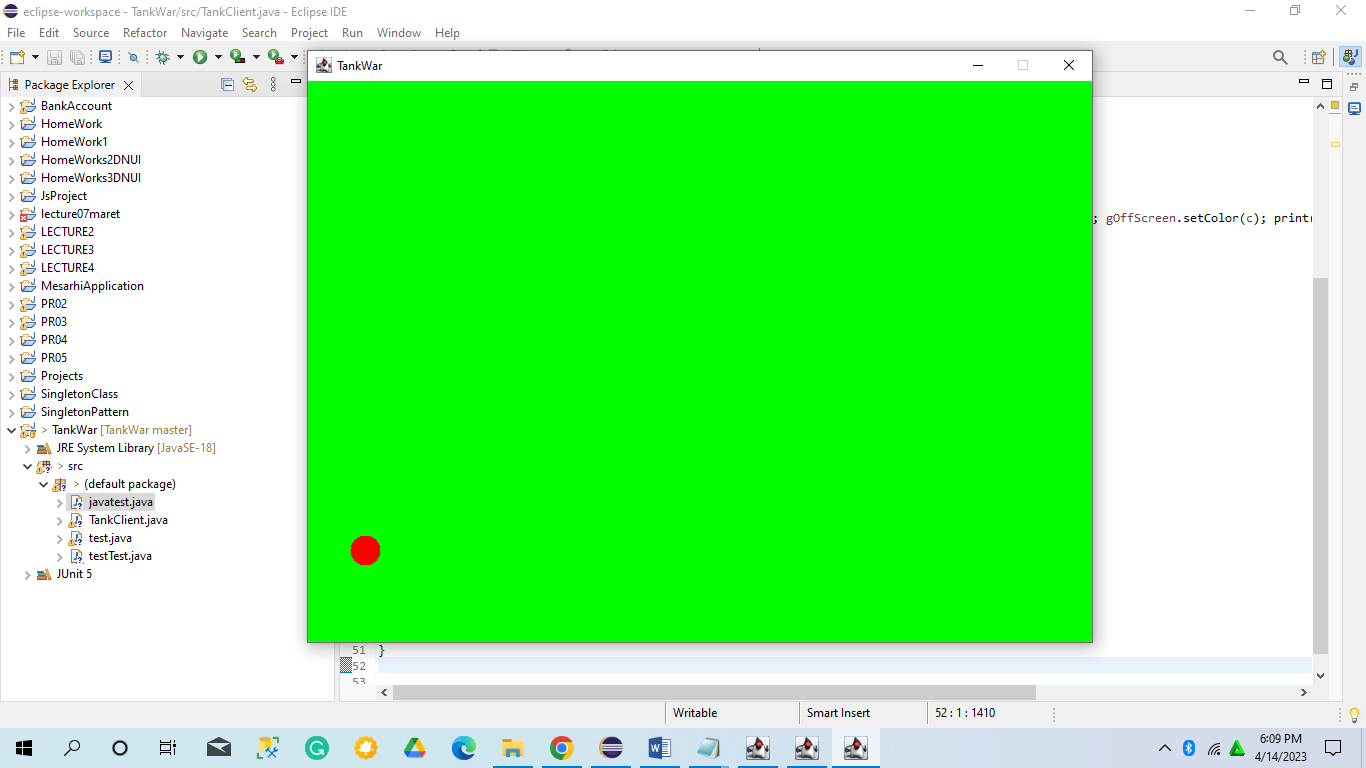
}

}

}

}

}



1. **Version 5**

**import** java.awt.\*;

**import** java.awt.event.\*;

**public** **class** TankClient **extends** Frame { **public static final int *GAME\_WIDTH* = 800;** **public** **static** **final** **int** ***GAME\_HEIGHT*** **=** **600;**

**int** x = 50, y = 50;

//这是一张虚拟图片

Image offScreenImage = **null**;

//The paint method does not need to be called and will be automatically called once it is to be redrawn

**public void** paint(Graphics g) { Color c = g.getColor(); g.setColor(Color.*RED*); g.fillOval(x, y, 30, 30); g.setColor(c);

y += 5;

}

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

**if**(offScreenImage == **null**) {

offScreenImage = **this**.createImage(*GAME\_WIDTH*,

*GAME\_ HEIGHT*);

}

//拿到这个图片的画笔

Graphics gOffScreen = offScreenImage.getGraphics();

Color c = gOffScreen.getColor(); gOffScreen.setColor(Color.*GREEN*); gOffScreen.fillRect(0, 0, *GAME\_WIDTH*, *GAME\_HEIGHT*); gOffScreen.setColor(c);

print(gOffScreen); g.drawImage(offScreenImage, 0, 0, **null**);

}

**public void** launchFrame() { **this**.setLocation(300, 50); **this**.setSize(*GAME\_WIDTH*, *GAME\_HEIGHT*); **this**.setTitle("TankWar"); addWindowListener(**new** WindowAdapter() {

**public void** windowClosing(WindowEvent e) { System.*exit*(0);

}

});

**this**.setResizable(**false**); **this**.setBackground(Color.*GREEN*); setVisible(**true**);

**new** Thread(**new** PaintThread()).start();

}

**public static void** main(String[] args) { TankClient tc = **new** TankClient(); tc.launchFrame();

}

**private class** PaintThread **implements** Runnable {

**public void** run() {

**while**(**true**) { repaint(); **try** {

Thread.*sleep*(50);

} **catch** (InterruptedException e) { e.printStackTrace();

}

}

}

}

}

Result

