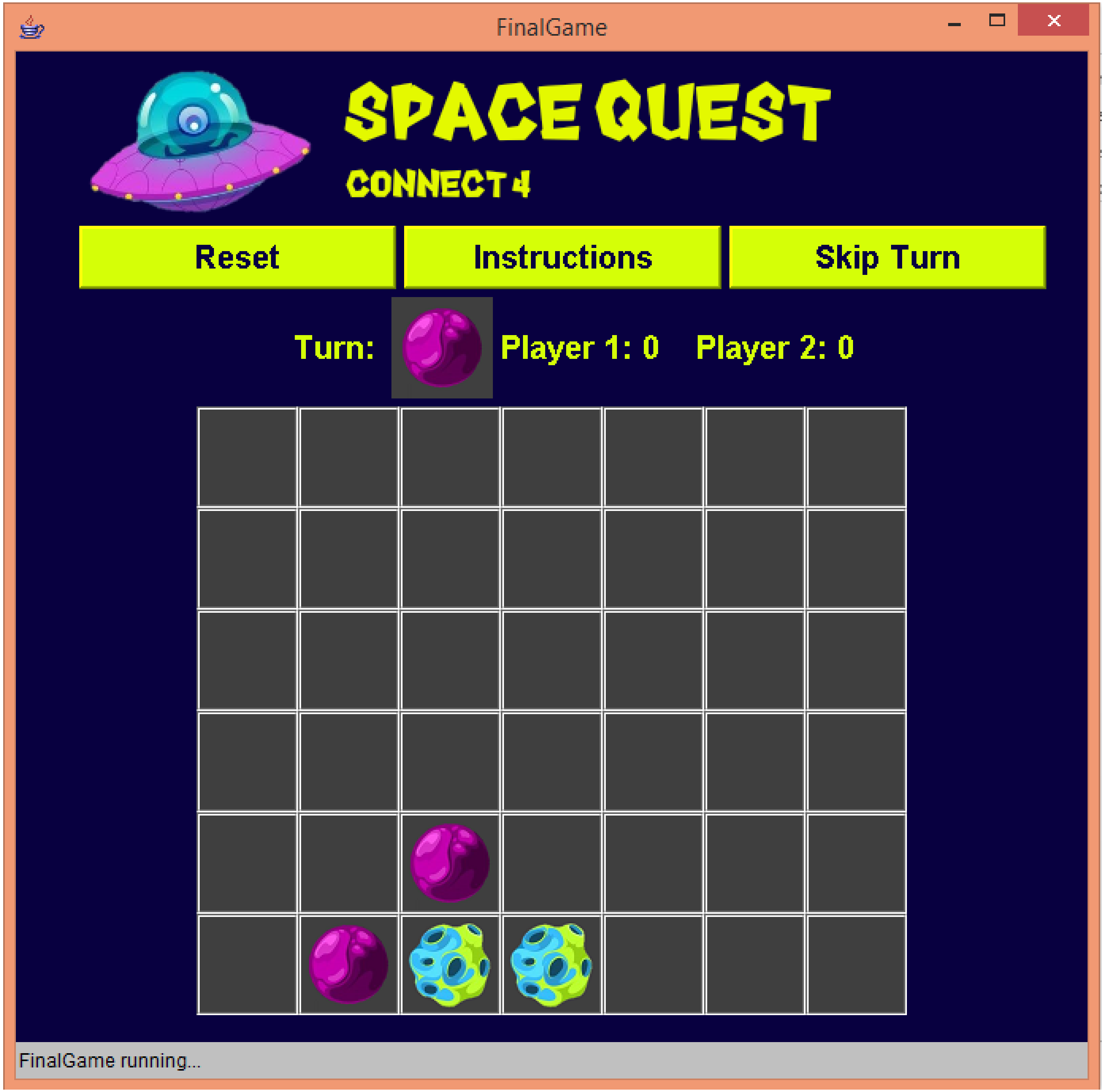
Applets

An Introduction



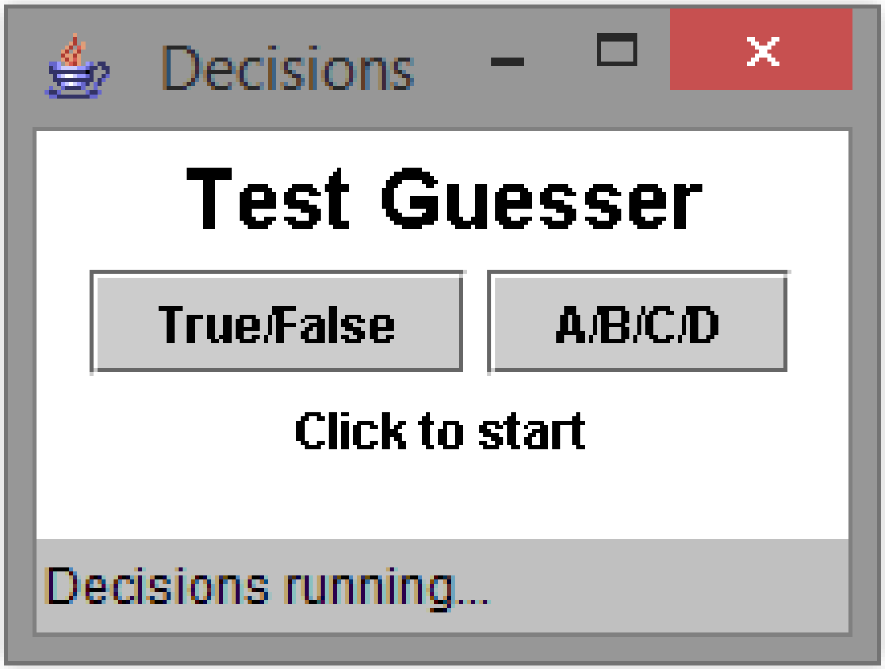
FinalGame running...

Graphical User Interfaces

Colour
Pictures
Widgets

Can click on the screen; user-driven.

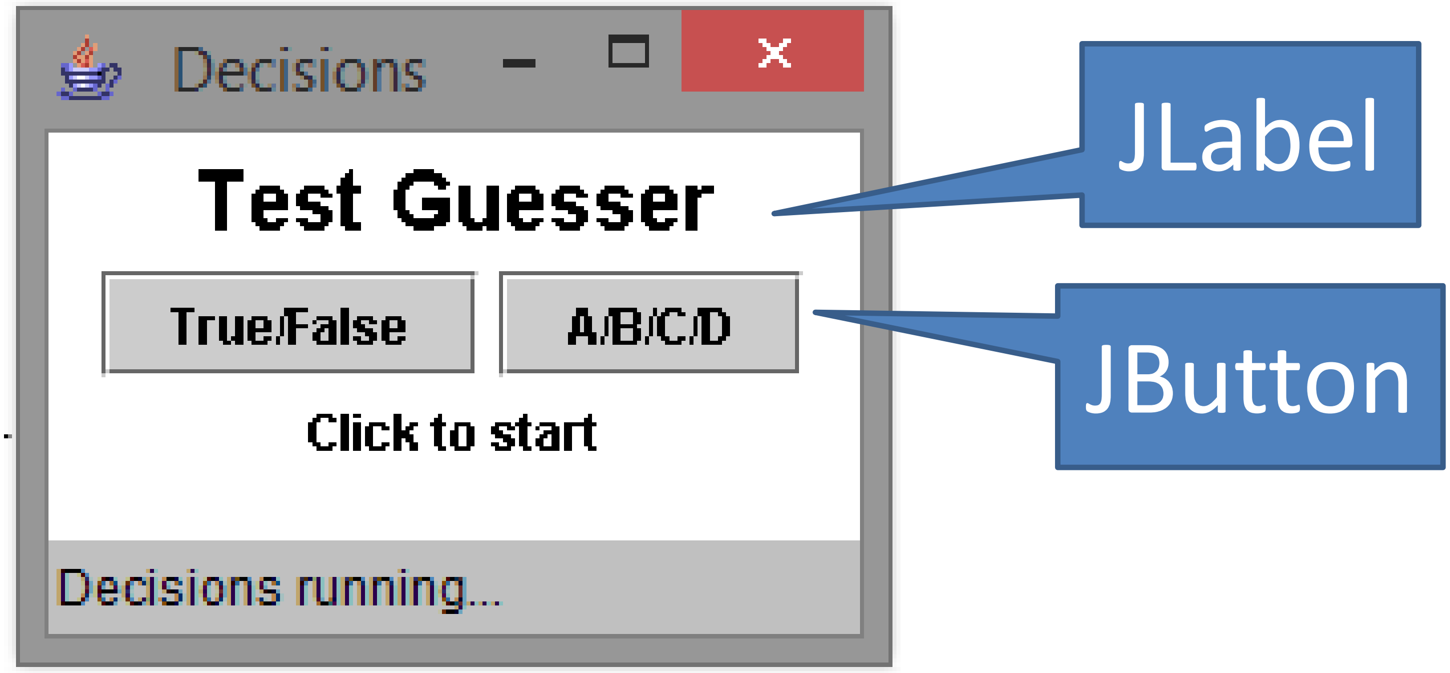
Graphical User Interfaces (GUIs) are made up of widgets



Click to start

Decisions running...

Graphical User Interfaces (GUIs) are made up of widgets



JLabel

JButton

```
public class basic
{
    public static void main (String args[])
    {
        new basic ();
    }

    public basic ()
    {
        System.out.println ("Hi");
        String name = IO.inputString ("Name? ");
    }
}
```

1. Class Line Changes:

Makes code into an Applet

Allows buttons to work (eventually)

```
public class basic extends Applet implements ActionListener
{
    public static void main (String args[])
    {
        new basic ();
    }

    public basic ()
    {
        System.out.println ("Hi");
        String name = IO.inputString ("Name? ");
    }
}
```

2. Libraries

Libraries are code written by experts.
We will be reusing the methods inside them.

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;

public class basic extends Applet implements ActionListener
{
    public static void main (String args[])
    {
        new basic ();
    }

    public basic ()
    {
        System.out.println ("Hi");
        String name = IO.inputString ("Name? ");
    }
}
```

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;

public class basic extends Applet implements ActionListener
{
    public static void main (String args[])
    {
        new basic ();
    }

    public basic ()
    {
        System.out.println ("Hi");
        String name = IO.inputString ("Name? ");
    }


}
```

3. New Default Methods

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;

public class basic extends Applet implements ActionListener
{
   public void init ()
   {      System.out.println ("Hi");
      String name = IO.inputString ("Name? ");
   }

   public void actionPerformed (ActionEvent e)
   {
   }

}
```

Init sets up the screen.

ActionPerformed runs buttons

4. Widgets

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;

public class basic extends Applet implements ActionListener
{
    public void init ()
  {    JLabel hi = new JLabel ("Hi");
        JLabel prompt = new JLabel("Name?");
        JTextField name = new JTextField (10);
        JButton click = new JButton ("Click to enter");
        add (hi);
        add (prompt);
        add (name);
        add (click);
    }
    public void actionPerformed (ActionEvent e)
    {
    }
}
```

A card you should write down.

CLI

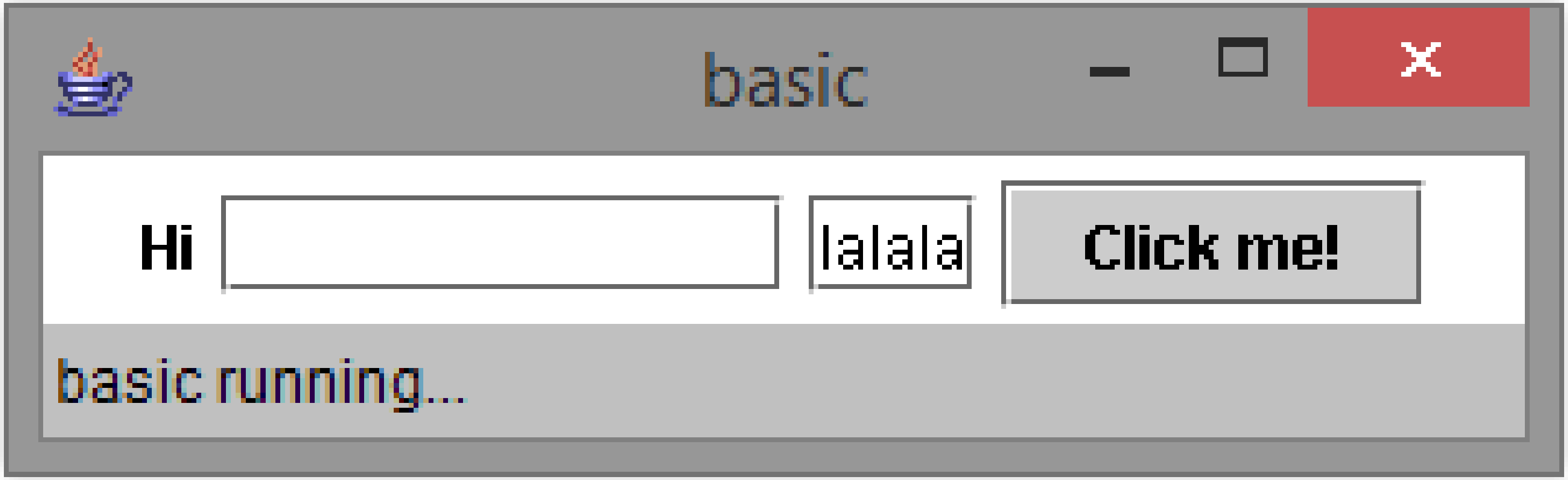
- A command line interface
- They are simple, so they don’t need libraries
- They are computer-driven; the code tells the user what they must do next.
- They input using IO (Keyboards).
- Their output is uses System.out.println and it can’t have colour or real pictures.
- The two main methods are the main method and the constructor.

A card you should write down.

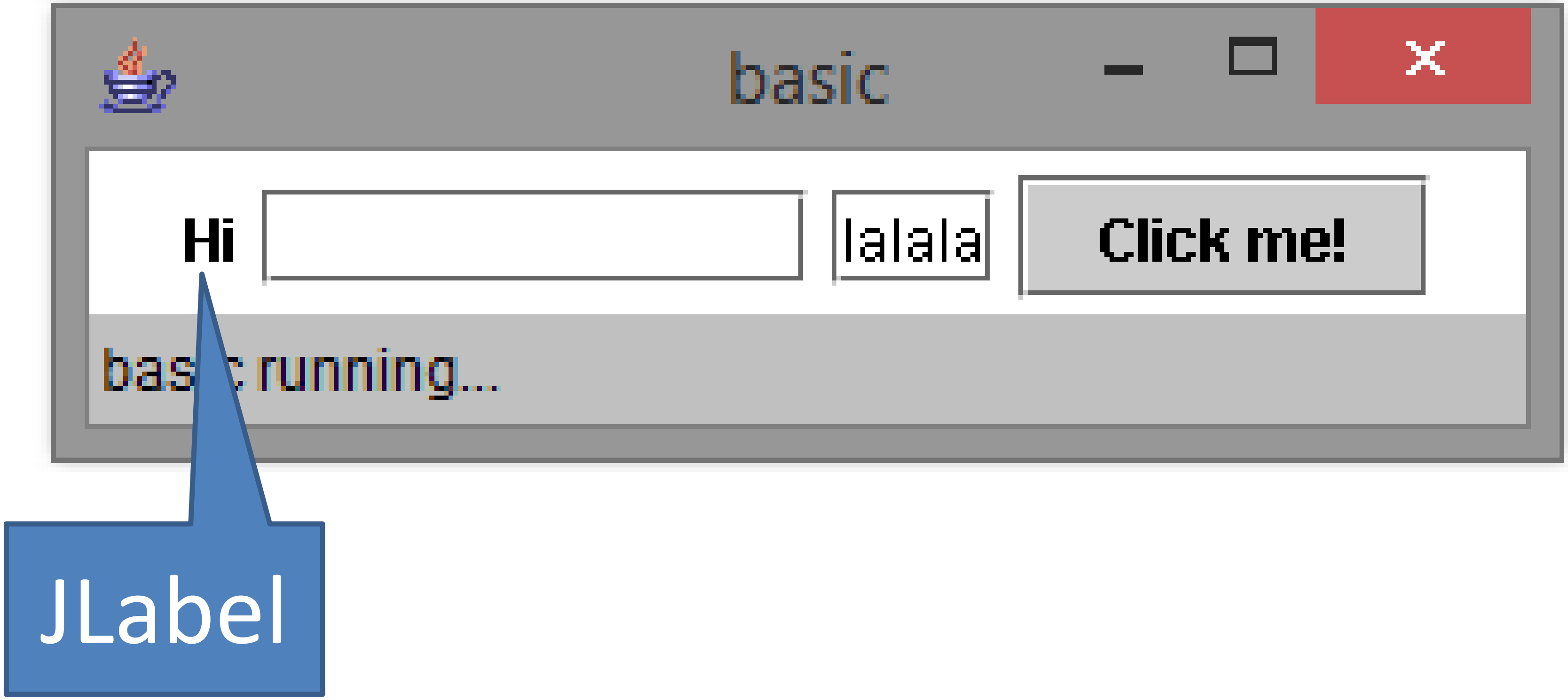
GUI

- A graphic user interface
- They are complex, so we draw on libraries for additional code
- They are user-driven; the user can decide what they want to do next.
- They input using JTextFields (Keyboards) and JButtons (mouse)
- Their output is mostly on JLabels and it can have colour and pictures
- The two main methods are init and actionPerformed.

How many widgets are on the screen?
What type are they?

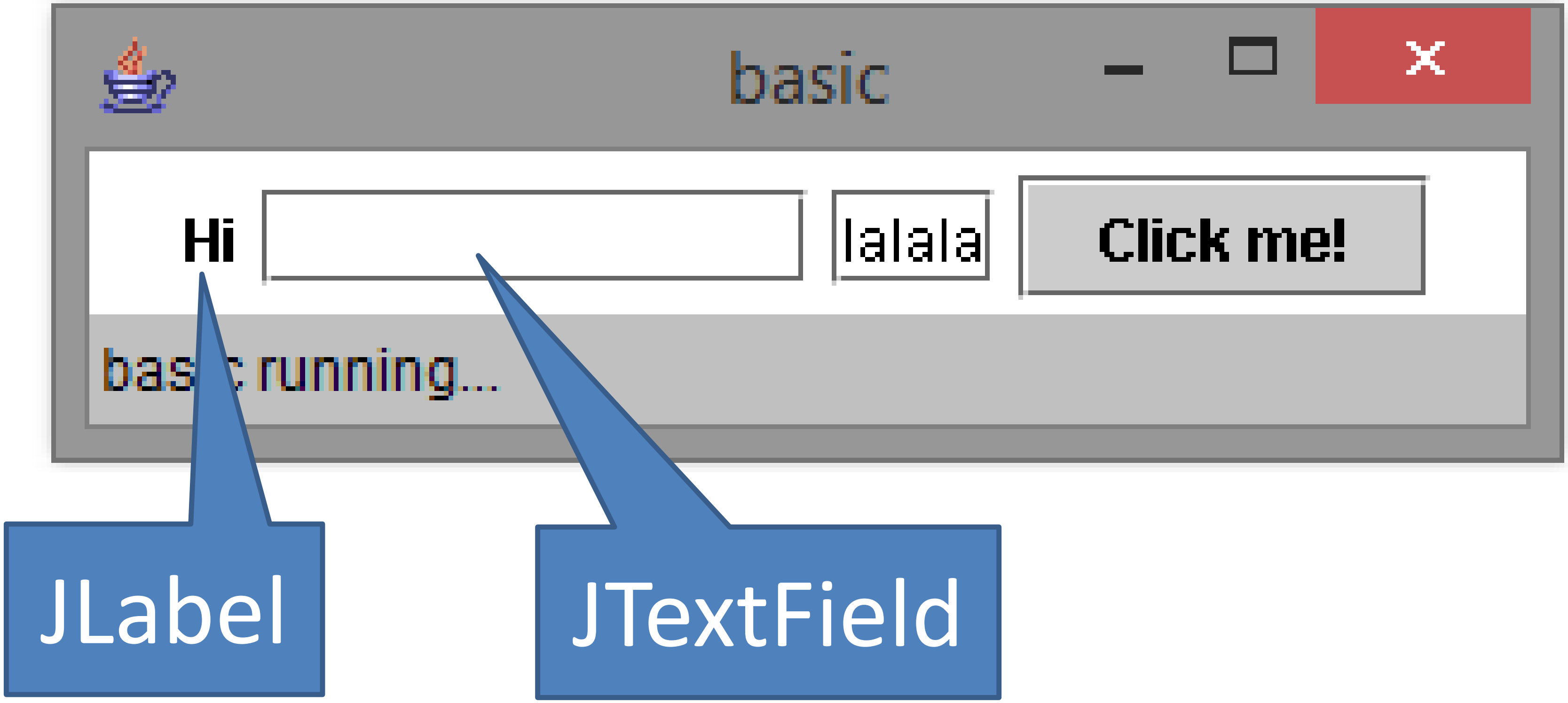


How many widgets are on the screen?
What type are they?

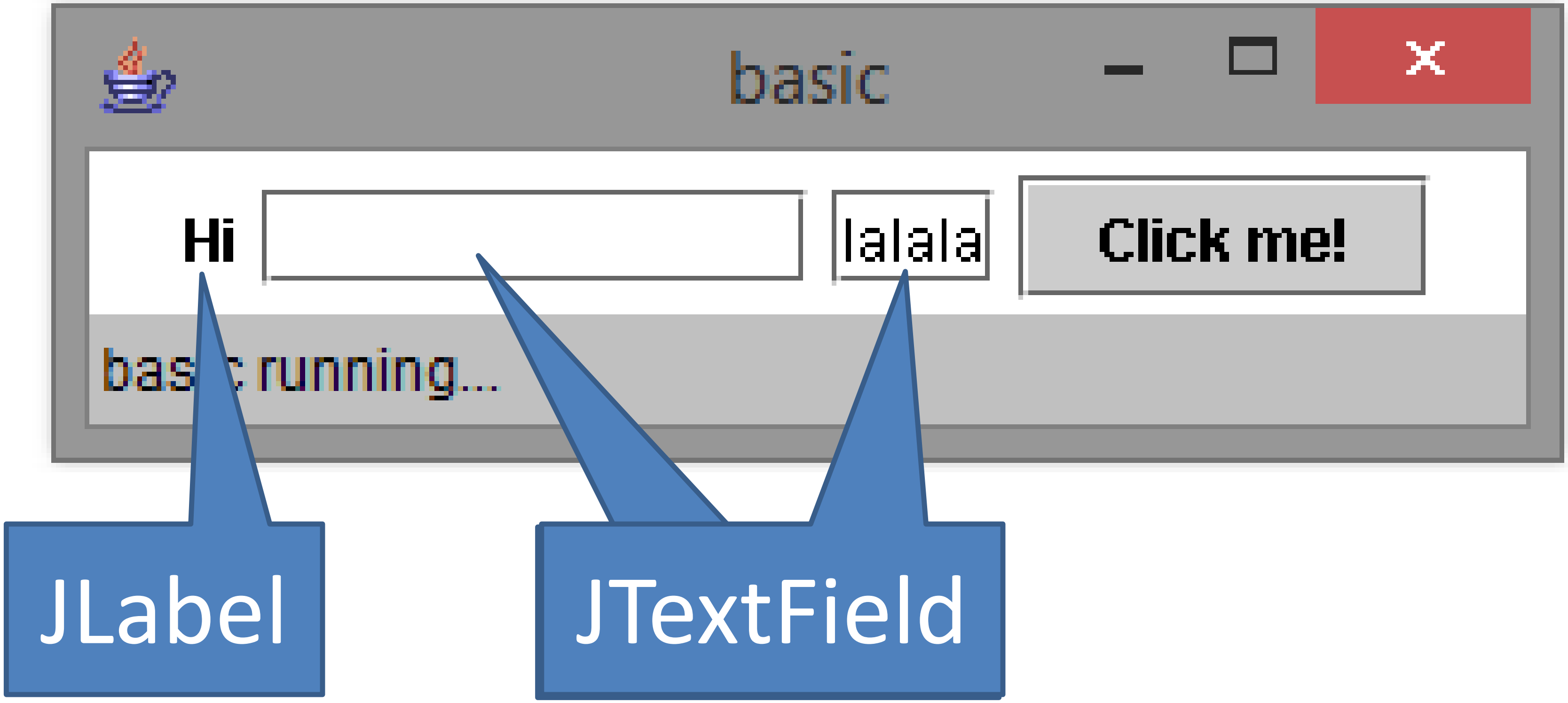


JLabel

How many widgets are on the screen? What type are they?



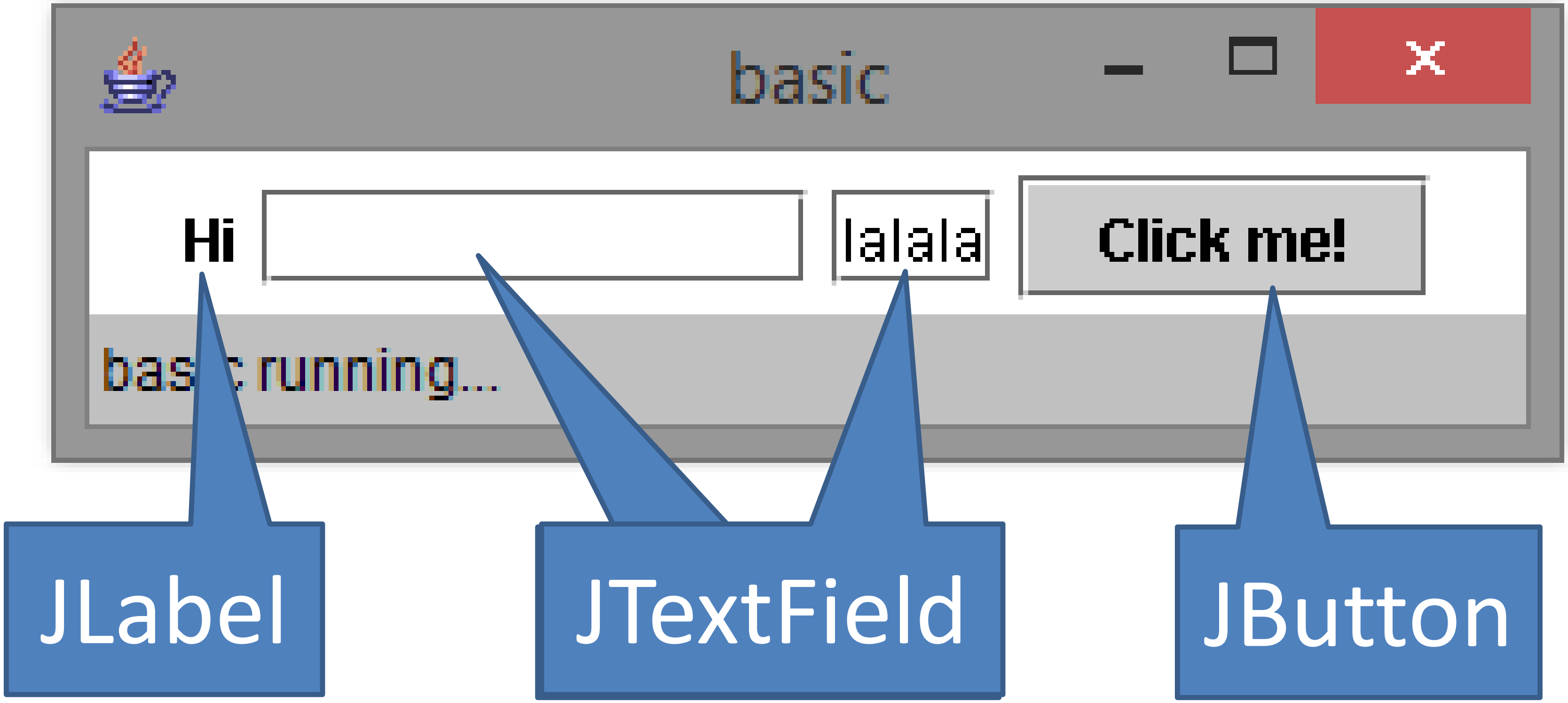
How many widgets are on the screen?
What type are they?



JLabel

JTextField

How many widgets are on the screen?
What type are they?

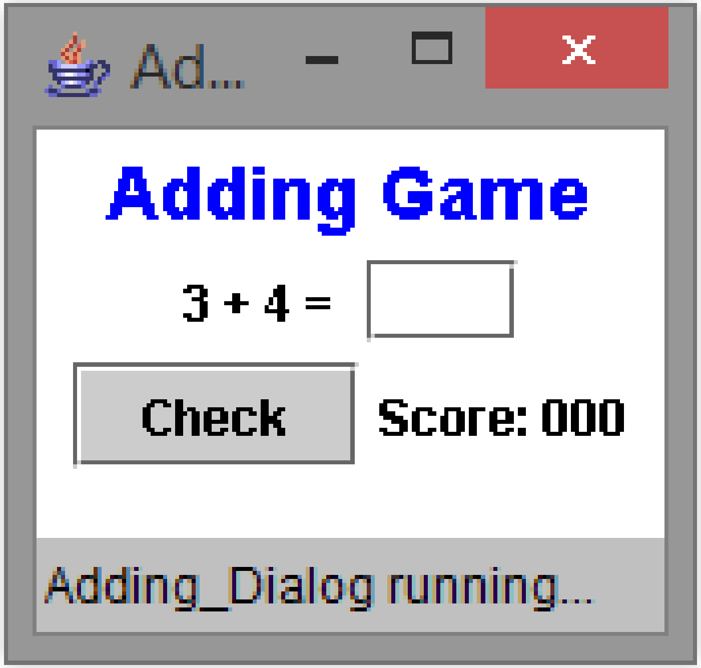


JLabel

JTextField

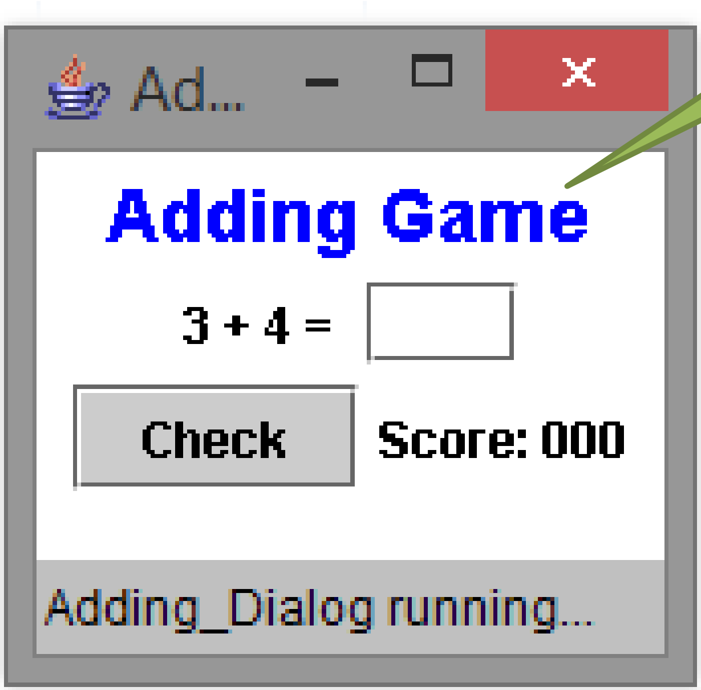
JButton

How many widgets are on the screen?
What type are they?



Adding_Dialog running...

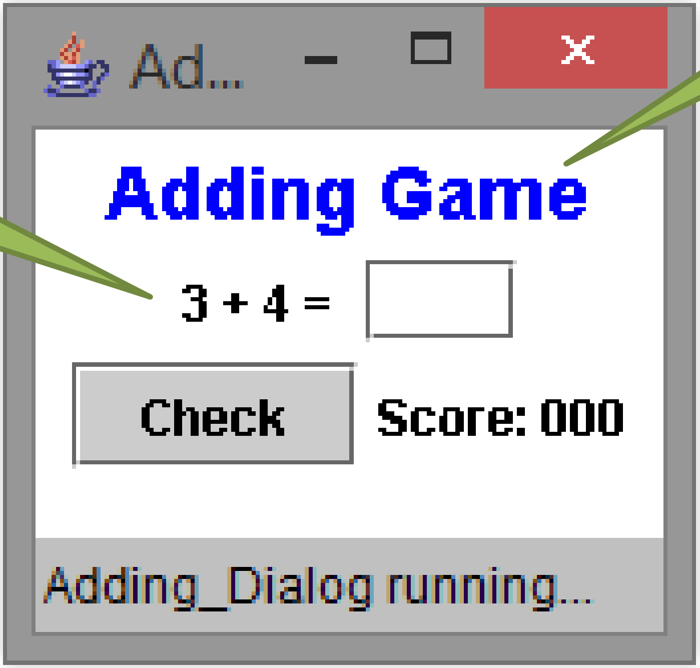
How many widgets are on the screen? What type are they?



JLabel

How many widgets are on the screen? What type are they?

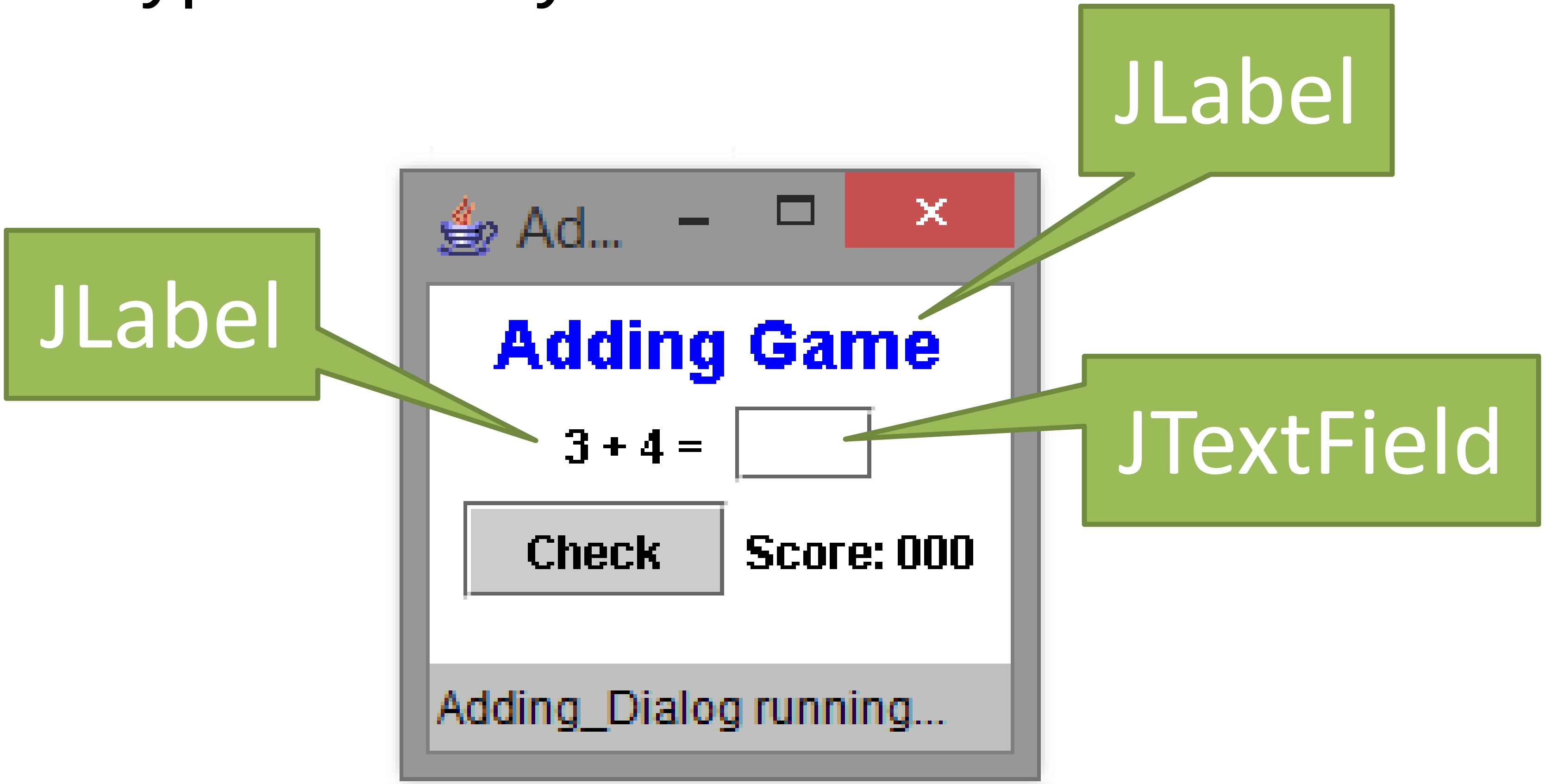
JLabel



Adding_Dialog running...

JLabel

How many widgets are on the screen?
What type are they?

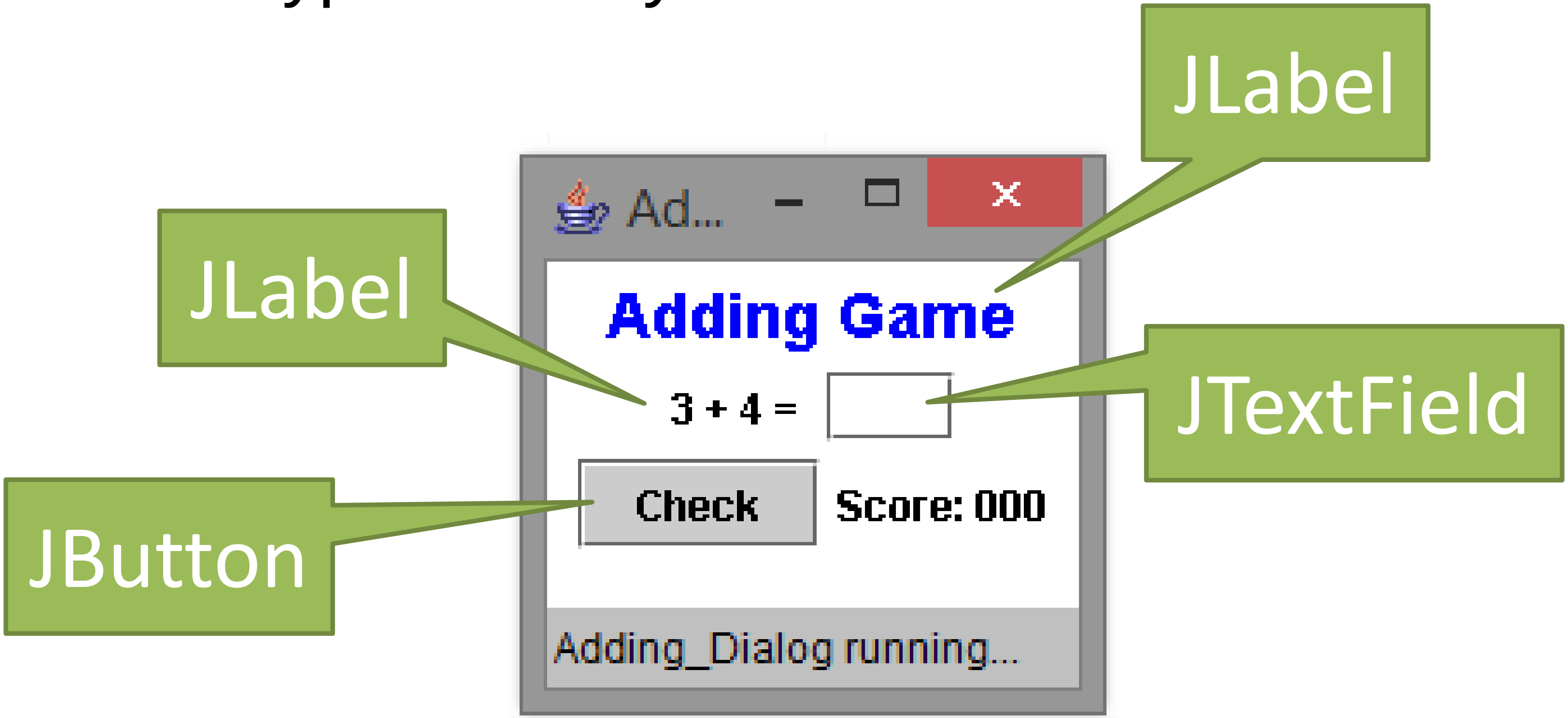


JLabel

JLabel

JTextField

How many widgets are on the screen?
What type are they?



Adding Game

3 + 4 =

Check

Score: 000

Adding_Dialog running...

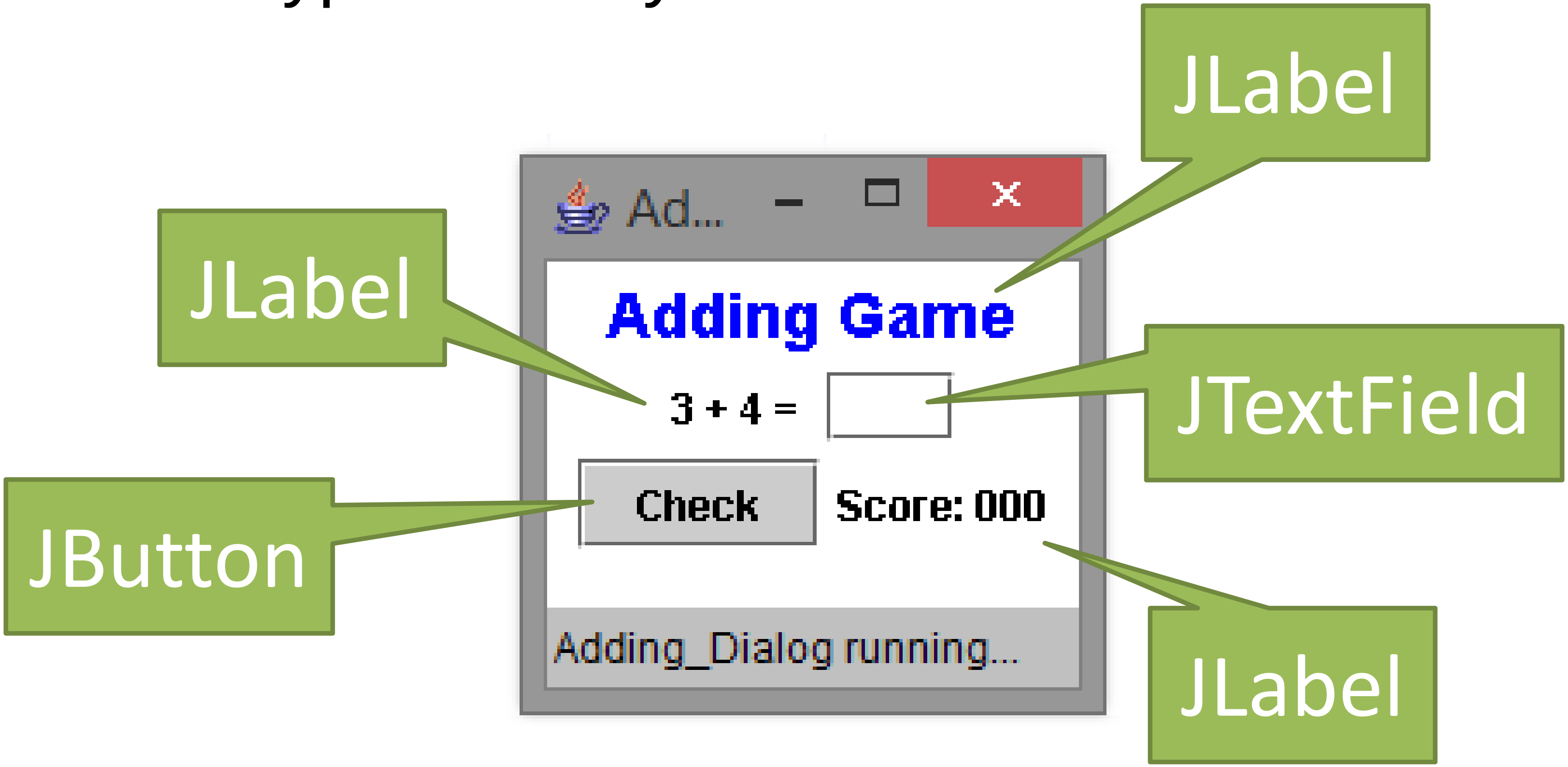
JLabel

JLabel

JTextField

JButton

How many widgets are on the screen?
What type are they?



Adding_Dialog running...

JLabel

JLabel

JTextField

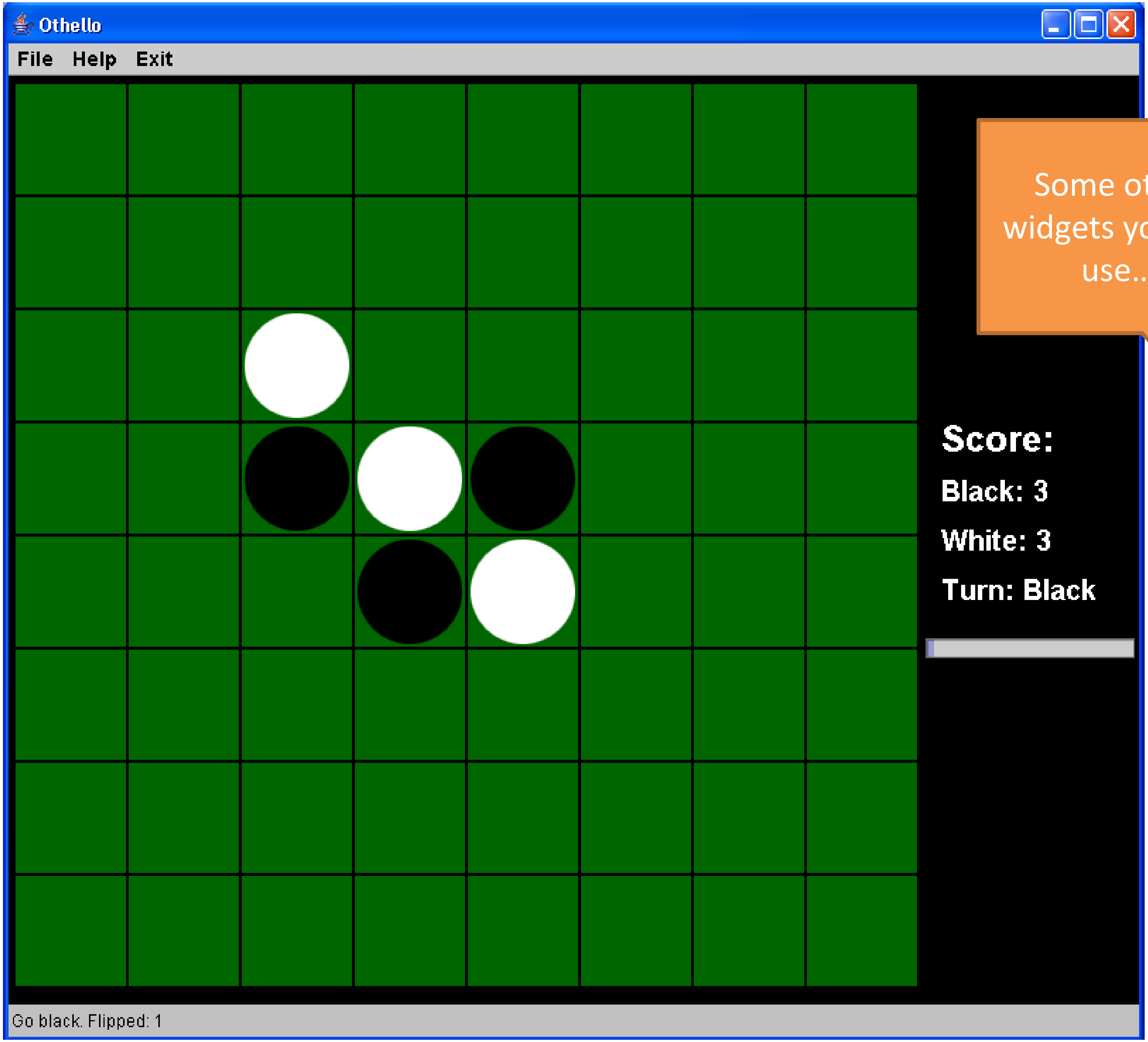
JButton

JLabel

A card you should write down.

Widget

- A widget is a piece of a user interface
- Users can see widgets, click on them, or type in them.
- They are more complex than simple variables (simple types are ints, doubles, chars). Because of their complex memory structures, we need to set them up in more complex ways and use special methods to edit them.
- The three most common widgets are: JLabel, JTextFields, JButtons



Score:

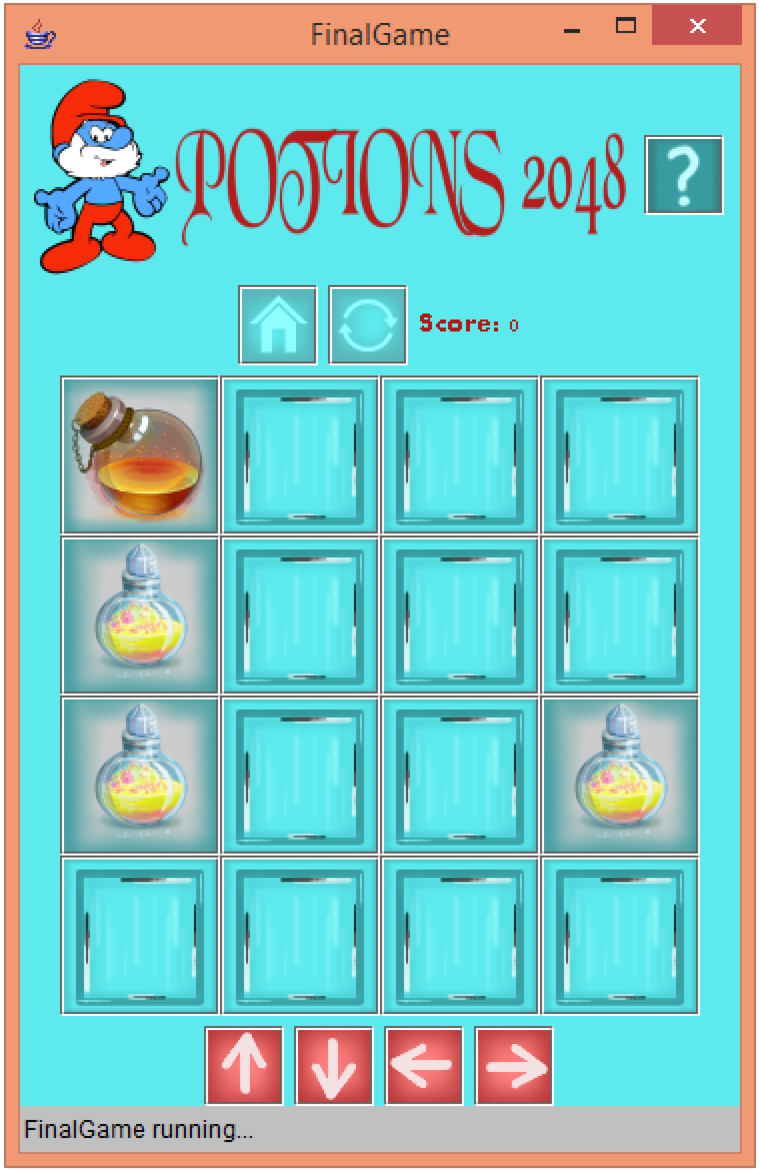
Black: 3

White: 3

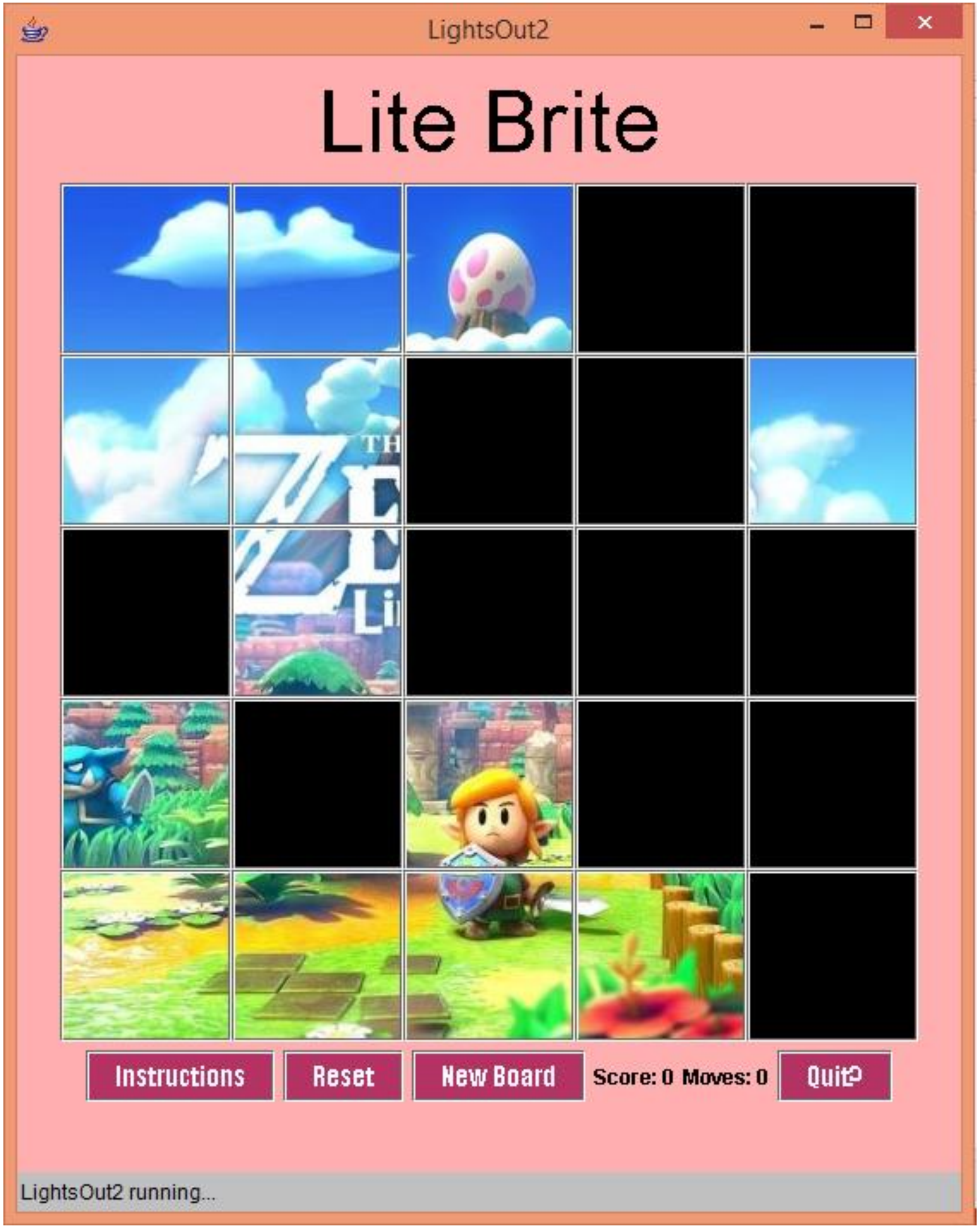
Turn: Black

Go black. Flipped: 1

Some other widgets you can use…



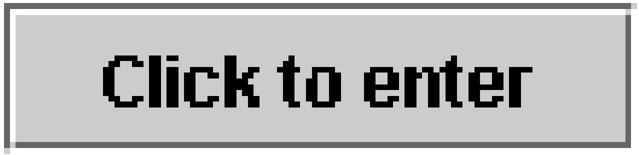
Ways you can use JLabels



LightsOut2 running...

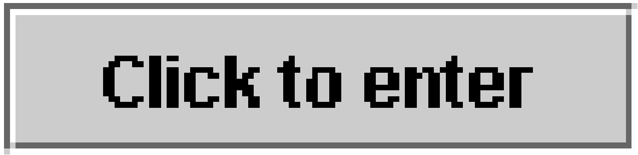
More ways you can use JLabels

How would you code this button?



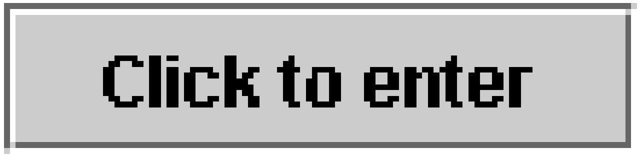
```
JButton __________ = new _____________ (“_____________________”);

add (___________);
```

How would you code this button?



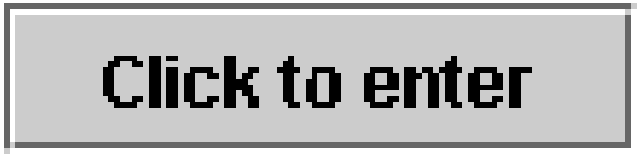
```
JButton click = new ____________ ("____________________");

add (__________);
```

How would you code this button?



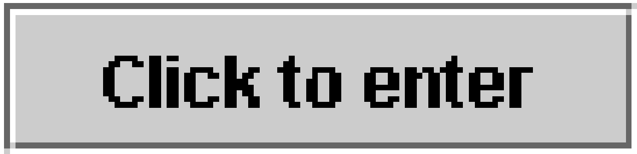
```
JButton click = new JButton (“____________________”);

add (__________);
```

How would you code this button?



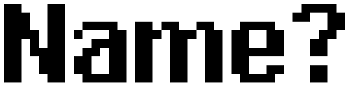
```
JButton click = new JButton (“Click to enter”);

add (__________);
```

How would you code this button?



```
JButton click = new JButton (“Click to enter”);
add (click);
```

How would you code this Label?



Name?

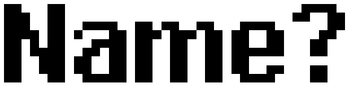
```
JLabel _____________ = new ____________ (“_____________”);

add (_____________);
```

How would you code this Label?

Name?

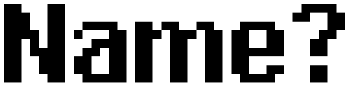
```
JLabel _prompt__________ = new ______________ (“_______________”);

add (_______________);
```

How would you code this Label?



```
JLabel _prompt_ = new _JLabel_ ("______________");

add (______________);
```

How would you code this Label?


Name?

```
JLabel ___prompt___ = new ___JLabel___ (“___Name?___”);

add (______________);
```

How would you code this Label?

Name?

```
JLabel _prompt_ = new _JLabel_ ("_Name?_");
add (_prompt_);
```

How would you code this textField?

```
JTextField ____________ = new ___________________ (______);

add (______________);
```

How would you code this textField?

```
JTextField ___name___ = new ________________ (______);

add (_____________);
```

How would you code this textField?

```
JTextField ___name___ = new ___JTextField___ (______);

add (_______________);
```

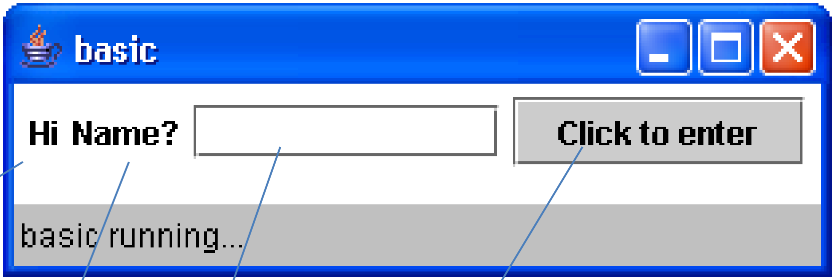
How would you code this textField?

```
JTextField __name__ = new __JTextField__ (__10__);

add (_______________);
```

How would you code this textField?

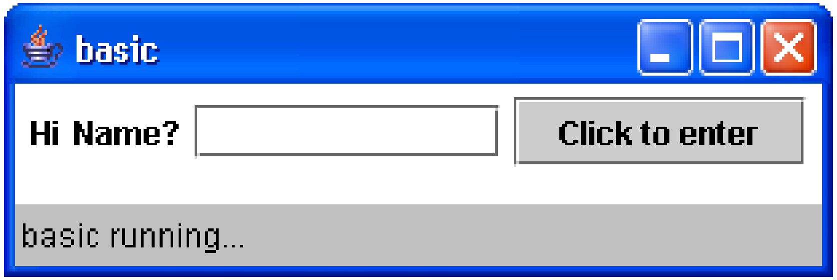
```
JTextField name = new JTextField (10);

add (name);
```

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;

public class basic extends Applet implements ActionListener
{
  public void init ()
  {   JLabel hi = new JLabel ("Hi");
      JLabel prompt = new JLabel("Name?");
      JTextField name = new JTextField (10);
      JButton click = new JButton ("Click to enter");
       add (hi);
      add (prompt);
      add (name);
      add (click);
  }
  public void actionPerformed (ActionEvent e)
  {
  }
}
```



basic running...

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;

public class basic extends Applet implements ActionListener
{
   public void init ()
  {   JLabel hi = new JLabel ("Hi");
      JLabel prompt = new JLabel("Name?");
      JTextField name = new JTextField (10);
      JButton click = new JButton ("Click to enter");
       add (hi);
      add (prompt);
      add (name);
      add (click);
   }
   public void actionPerformed (ActionEvent e)
   {
   }
}
```



basic running...

5. Output

Widget type   Variable name   Construction   Default Value

```
JLabel  hi  = new  JLabel ("Hi");
add (hi);
```

Add widget to the screen
so it shows up

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;

public class basic extends Applet implements ActionListener
{
    public void init ()
    {
        JLabel hi = new JLabel ("Hi");
        JTextField bye = new JTextField (10);
        JTextField noodle = new JTextField ("lalala");
        JButton click = new JButton ("Click me!");
        add (hi);
        add (bye);
        add (noodle);
        add (click);
    }

    public void actionPerformed (ActionEvent e)
    {
    }
}
```

These are **libraries**. They contain extra code to make your Applets run.

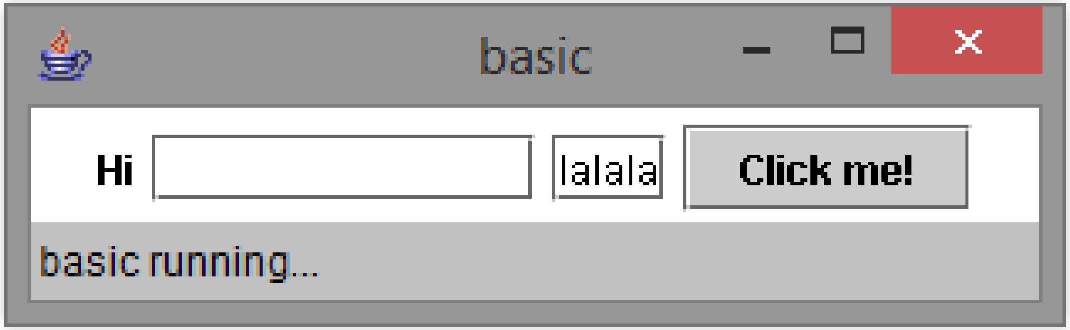
The class name is basic. It should be saved as basic.java

This is the **init** method. It is where we put the code to initialize and set up the screen.

There are 4 **widgets**: a JLabel, 2 JTextFields and a JButton.
A widget is something you can see or click on the screen.
The JLabel is named hi and it has the word "Hi" on it.

Once "newed" or **constructed**, each widget needs to be added to the screen so it shows up.

This is the **actionPerformed** method. It is where we put code to make the applet do something useful. This one is blank so our applet does nothing.

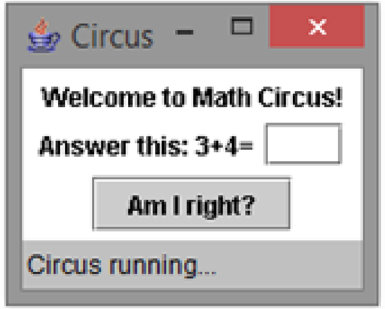


basic running...

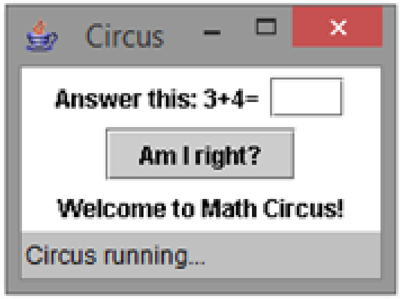
6. Match the adding order in the init method with the output. These are the widgets:

```
JLabel title = new JLabel ("Welcome to Math Circus!");
JLabel ques = new JLabel ("Answer this: 3+4=");
JTextField ans = new JTextField (3);
JButton right = new JButton ("Am I right?");
```

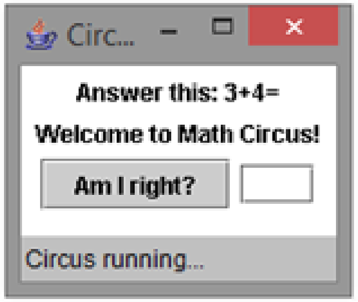
(a) matches with _____



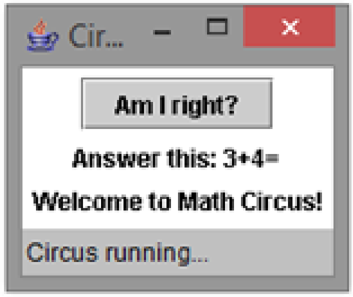
(b) matches with _____



(c) matches with _____



(d) matches with _____



(1) `add (ques);`
`add (ans);`
`add (right);`
`add (title);`
	(2) `add (right);`
	`add (ques);`
	`add (title);`
		(3) `add (title);`
		`add (ques);`
		`add (ans);`
		`add (right);`
			(4) `add (ques);`
			`add (title);`
			`add (right);`
			`add (ans);`

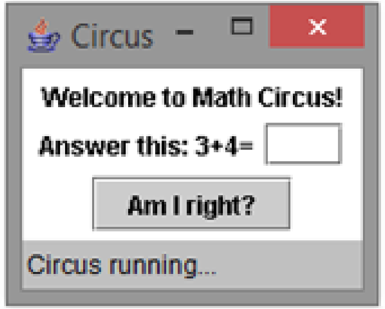
title

Welcome to Math Circus!

6. Match the adding order in the init method with the output. These are the widgets:

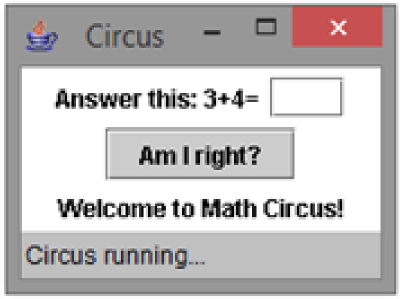
```
JLabel title = new JLabel ("Welcome to Math Circus!");
JLabel ques = new JLabel ("Answer this: 3+4=");
JTextField ans = new JTextField (3);
JButton right = new JButton ("Am I right?");
```

(a) matches with _____ (b) matches with ____ (c) matches with ____ (d) matches with ____



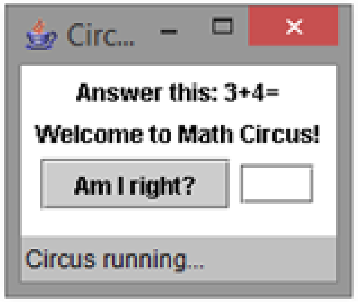
Welcome to Math Circus!
Answer this: 3+4=

Circus running...



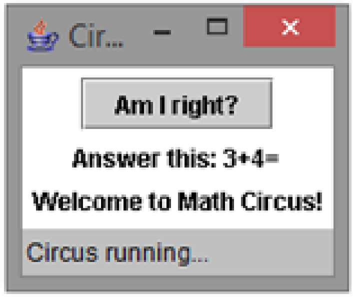
Answer this: 3+4=

Welcome to Math Circus!
Circus running...



Answer this: 3+4=
Welcome to Math Circus!

Circus running...



Answer this: 3+4=
Welcome to Math Circus!
Circus running...

(1) add (ques);
add (ans);
add (right);
add (title);
	(2) add (right);
	add (ques);
	add (title);
		(3) add (title);
		add (ques);
		add (ans);
		add (right);
			(4) add (ques);
			add (title);
			add (right);
			add (ans);

title

Welcome to Math Circus!

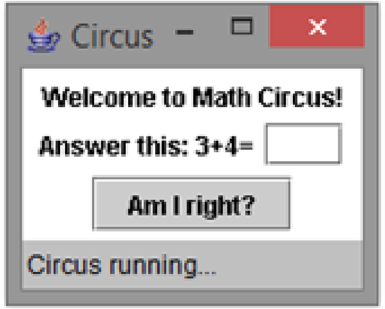
ques

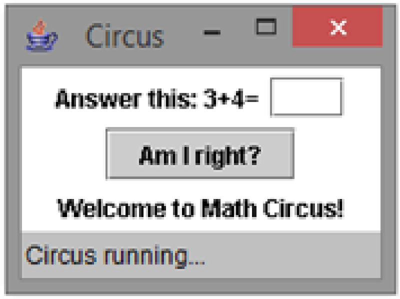
Answer this: 3+4=

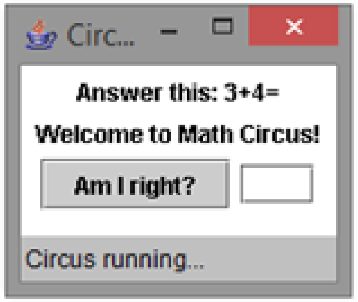
6. Match the loading order in the init method with the output. These are the widgets:

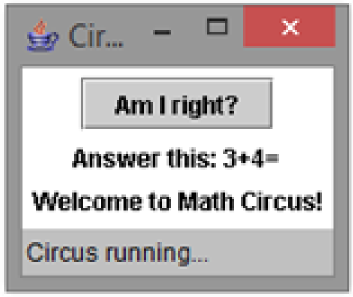
```
    JLabel title = new JLabel ("Welcome to Math Circus!");
    JLabel ques = new JLabel ("Answer this: 3+4=");
    JTextField ans = new JTextField (3);
    JButton right = new JButton ("Am I right?");
```

(a) matches with _____ (b) matches with ____ (c) matches with ____ (d) matches with ____









(1) `add (ques);`
`add (ans);`
`add (right);`
`add (title);`
	(2) `add (right);`
	`add (ques);`
	`add (title);`
		(3) `add (title);`
		`add (ques);`
		`add (ans);`
		`add (right);`
			(4) `add (ques);`
			`add (title);`
			`add (right);`
			`add (ans);`

title

Welcome to Math Circus!

ques

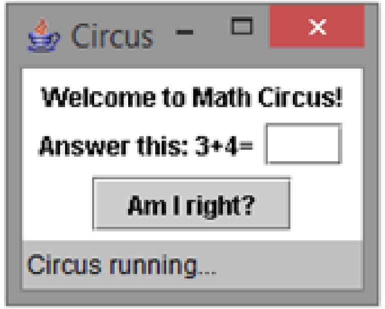
Answer this: 3+4=

ans

6. Match the ading order in the init method with the output. The are the wid

```
JLabel title = new JLabel ("Welcome to Math Circus!");
JLabel ques = new JLabel ("Answer this: 3+4=");
JTextField ans = new JTextField (3);
JButton right = new JButton ("Am I right?");
```

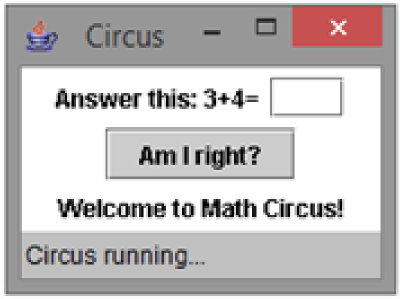
(a) matches with _____ (b) matches with ____ (c) matches with ____ (d) matches with ____



Welcome to Math Circus!

Answer this: 3+4=

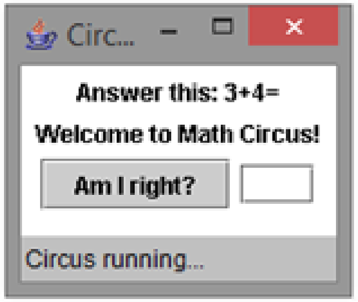
Circus running...



Answer this: 3+4=

Welcome to Math Circus!

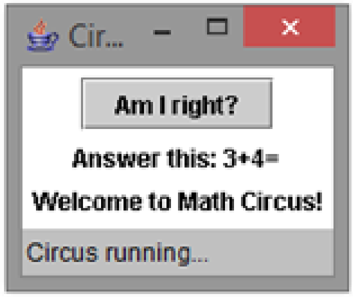
Circus running...



Answer this: 3+4=

Welcome to Math Circus!

Circus running...



Answer this: 3+4=

Welcome to Math Circus!

Circus running...

(1) `add (ques);`
`add (ans);`
`add (right);`
`add (title);`
	(2) `add (right);`
	`add (ques);`
	`add (title);`
		(3) `add (title);`
		`add (ques);`
		`add (ans);`
		`add (right);`
			(4) `add (ques);`
			`add (title);`
			`add (right);`
			`add (ans);`

title

Welcome to Math Circus!

ques

Answer this: 3+4=

ans

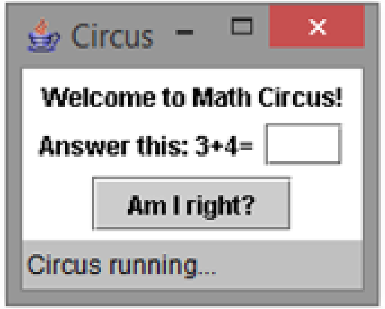
right

Am I right?

6. Match the adding order in the init method with the output. These are the wid

```
    JLabel title = new JLabel ("Welcome to Math Circus!");
    JLabel ques = new JLabel ("Answer this: 3+4=");
    JTextField ans = new JTextField (3);
    JButton right = new JButton ("Am I right?");
```

(a) matches with _____ (b) matches with ____ (c) matches with ____ (d) matches with ____



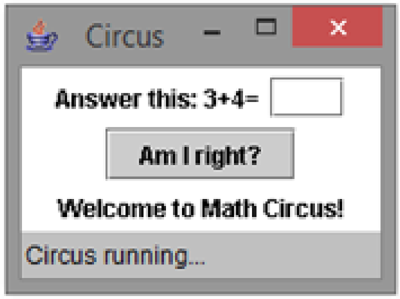
Circus

Welcome to Math Circus!

Answer this: 3+4=

Am I right?

Circus running...



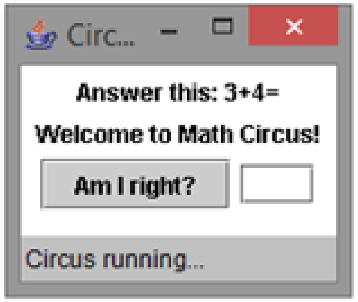
Circus

Answer this: 3+4=

Am I right?

Welcome to Math Circus!

Circus running...



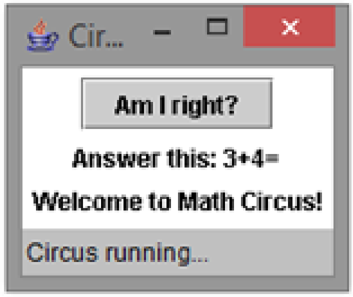
Circ...

Answer this: 3+4=

Welcome to Math Circus!

Am I right?

Circus running...



Cir...

Am I right?

Answer this: 3+4=

Welcome to Math Circus!

Circus running...

(1) add (ques);
add (ans);
add (right);
add (title);
	(2) add (right);
	add (ques);
	add (title);
		(3) add (title);
		add (ques);
		add (ans);
		add (right);
			(4) add (ques);
			add (title);
			add (right);
			add (ans);

title

Welcome to Math Circus!

ques

Answer this: 3+4=

ans

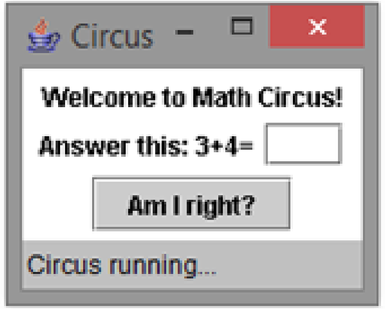
right

Am I right?

6. Match the ading order in the init method with the output. The are the wid

```
    JLabel title = new JLabel ("Welcome to Math Circus!");
    JLabel ques = new JLabel ("Answer this: 3+4=");
    JTextField ans = new JTextField (3);
    JButton right = new JButton ("Am I right?");
```

(a) matches with _____ (b) matches with ____ (c) matches with ____ (d) matches with ____

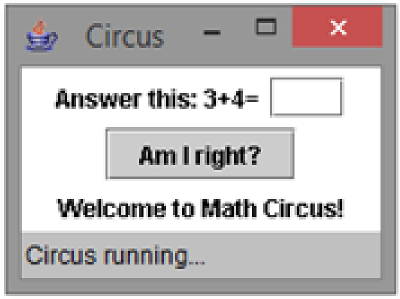


Welcome to Math Circus!

Answer this: 3+4=

Am I right?

Circus running...

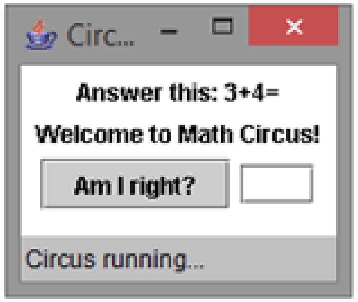


Answer this: 3+4=

Am I right?

Welcome to Math Circus!

Circus running...

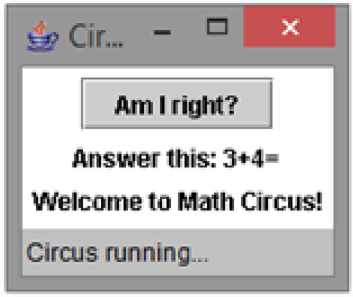


Answer this: 3+4=

Welcome to Math Circus!

Am I right?

Circus running...



Am I right?

Answer this: 3+4=

Welcome to Math Circus!

Circus running...

(1) add (ques);
add (ans);
add (right);
add (title);
	(2) add (right);
	add (ques);
	add (title);
		(3) add (title);
		add (ques);
		add (ans);
		add (right);
			(4) add (ques);
			add (title);
			add (right);
			add (ans);

ques

ans

right

title

title

Welcome to Math Circus!

ques

Answer this: 3+4=

ans

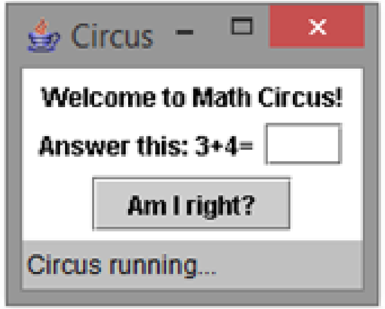
right

Am I right?

6. Match the ading order in the init method with the output. The are the wid

```
    JLabel title = new JLabel ("Welcome to Math Circus!");
    JLabel ques = new JLabel ("Answer this: 3+4=");
    JTextField ans = new JTextField (3);
    JButton right = new JButton ("Am I right?");
```

(a) matches with _____ (b) matches with ____ (c) matches with ____ (d) matches with ____

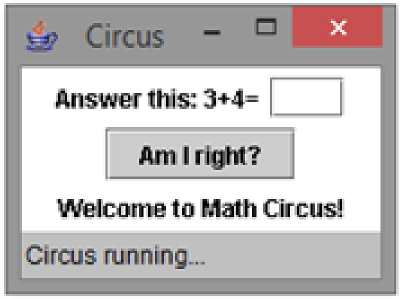


Welcome to Math Circus!

Answer this: 3+4=

Am I right?

Circus running...

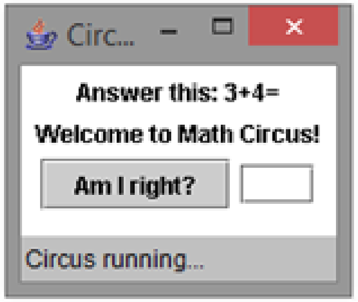


Answer this: 3+4=

Am I right?

Welcome to Math Circus!

Circus running...

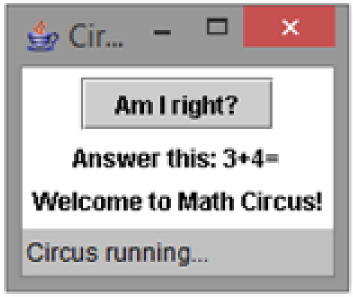


Answer this: 3+4=

Welcome to Math Circus!

Am I right?

Circus running...



Am I right?

Answer this: 3+4=

Welcome to Math Circus!

Circus running...

(1) add (ques);
add (ans);
add (right);
add (title);
	(2) add (right);
	add (ques);
	add (title);
		(3) add (title);
		add (ques);
		add (ans);
		add (right);
			(4) add (ques);
			add (title);
			add (right);
			add (ans);

ques

ans

right

title

Answer this: 3+4=

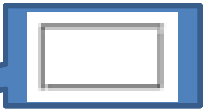
title

Welcome to Math Circus!

ques

Answer this: 3+4=

ans



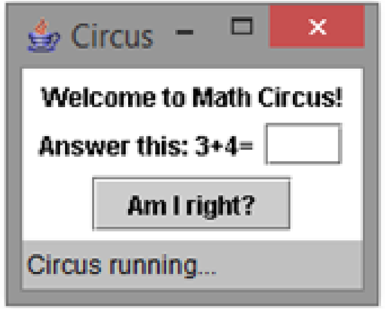
right

Am I right?

6. Match the adding order in the init method with the output. The are the wid

```
JLabel title = new JLabel ("Welcome to Math Circus!");
JLabel ques = new JLabel ("Answer this: 3+4=");
JTextField ans = new JTextField (3);
JButton right = new JButton ("Am I right?");
```

(a) matches with _____ (b) matches with ____ (c) matches with ____ (d) matches with ____

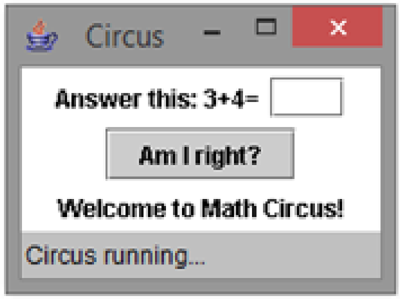


Welcome to Math Circus!

Answer this: 3+4=

Am I right?

Circus running...

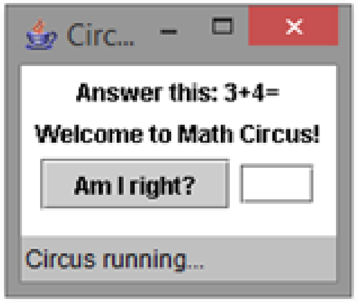


Answer this: 3+4=

Am I right?

Welcome to Math Circus!

Circus running...

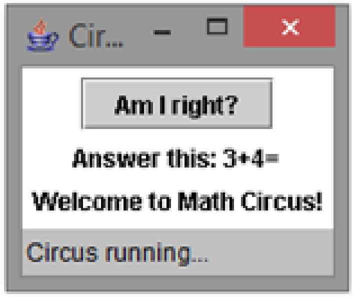


Answer this: 3+4=

Welcome to Math Circus!

Am I right?

Circus running...



Am I right?

Answer this: 3+4=

Welcome to Math Circus!

Circus running...

(1) add (ques);
add (ans);
add (right);
add (title);
	(2) add (right);
	add (ques);
	add (title);
		(3) add (title);
		add (ques);
		add (ans);
		add (right);
			(4) add (ques);
			add (title);
			add (right);
			add (ans);

ques

Answer this: 3+4=

ans

right

title

title

Welcome to Math Circus!

ques

Answer this: 3+4=

ans

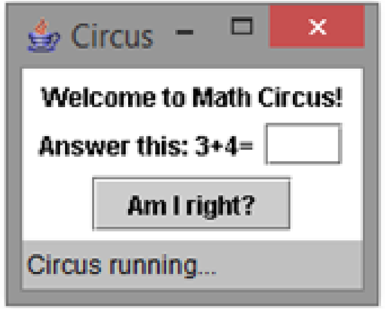
right

Am I right?

6. Match the ding order in the init method with the output. The are the wid

```
    JLabel title = new JLabel ("Welcome to Math Circus!");
    JLabel ques = new JLabel ("Answer this: 3+4=");
    JTextField ans = new JTextField (3);
    JButton right = new JButton ("Am I right?");
```

(a) matches with _____ (b) matches with ____ (c) matches with ____ (d) matches with ____

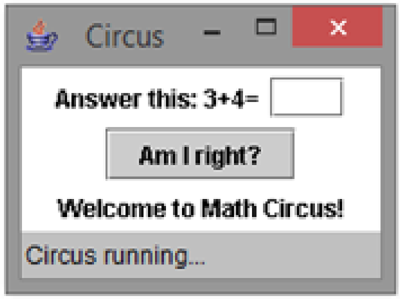


Welcome to Math Circus!

Answer this: 3+4=

Am I right?

Circus running...

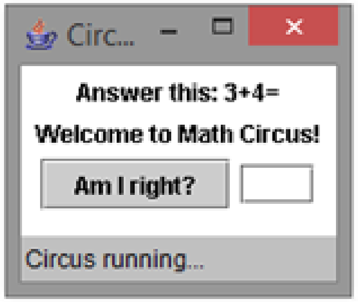


Answer this: 3+4=

Am I right?

Welcome to Math Circus!

Circus running...

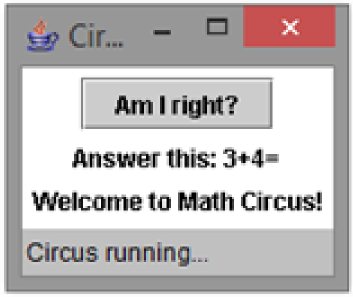


Answer this: 3+4=

Welcome to Math Circus!

Am I right?

Circus running...



Am I right?

Answer this: 3+4=

Welcome to Math Circus!

Circus running...

(1) add (ques);
add (ans);
add (right);
add (title);
	(2) add (right);
	add (ques);
	add (title);
		(3) add (title);
		add (ques);
		add (ans);
		add (right);
			(4) add (ques);
			add (title);
			add (right);
			add (ans);

ques Answer this: 3+4=

ans

right Am I right?

title

title — Welcome to Math Circus!

ques — Answer this: 3+4=

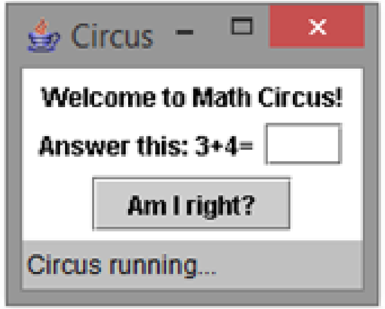
ans — □

right — Am I right?

6. Match the ading order in the init method with the output. The are the wid

```
JLabel title = new JLabel ("Welcome to Math Circus!");
JLabel ques = new JLabel ("Answer this: 3+4=");
JTextField ans = new JTextField (3);
JButton right = new JButton ("Am I right?");
```

(a) matches with _____ (b) matches with ____ (c) matches with ____ (d) matches with ____

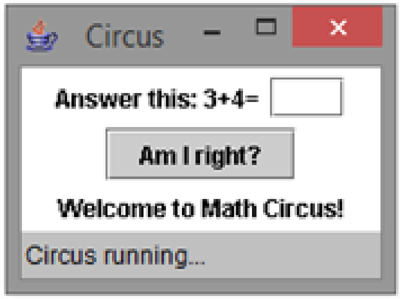

Circus

Welcome to Math Circus!

Answer this: 3+4= □

Am I right?

Circus running...

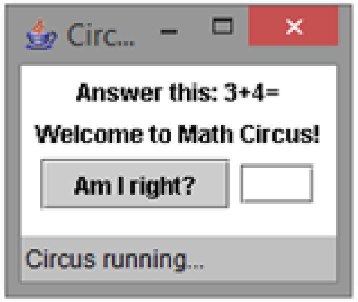

Circus

Answer this: 3+4= □

Am I right?

Welcome to Math Circus!

Circus running...

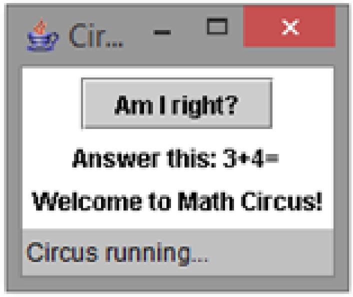

Circ...

Answer this: 3+4=

Welcome to Math Circus!

Am I right? □

Circus running...


Cir...

Am I right?

Answer this: 3+4=

Welcome to Math Circus!

Circus running...

(1) add (ques);
add (ans);
add (right);
add (title);
	(2) add (right);
	add (ques);
	add (title);
		(3) add (title);
		add (ques);
		add (ans);
		add (right);
			(4) add (ques);
			add (title);
			add (right);
			add (ans);

ques	Answer this: 3+4=
ans	□
right	Am I right?
title	Welcome to Math Circus!