

1. Создадим объект класса JFrame с пустым конструктором.

```
new *
6 ▶ public class Hello {
    new *
7 ▶ public static void main(String[] args) {
8     // объект класса JFrame с пустым конструктором
    new *
9     JFrame jFrame = new JFrame();
10 }
11 }
12 }
```

2. Добавим метод setVisible с параметром true.

```
new *
6 ▶ public class Hello {
    new *
7 ▶ public static void main(String[] args) {
8     // объект класса JFrame с пустым конструктором
    new *
9     JFrame jFrame = new JFrame();
10    // окно становится видимым и пользователь может взаимодействовать с ним
11    jFrame.setVisible(true);
12 }
13 }
```

3. Добавим метод setDefaultCloseOperation с параметром EXIT_ON_CLOSE.

```
new *
6 ▶ public class Hello {
    new *
7 ▶ public static void main(String[] args) {
8     // объект класса JFrame с пустым конструктором
    new *
9     JFrame jFrame = new JFrame();
10    // окно становится видимым и пользователь может взаимодействовать с ним
11    jFrame.setVisible(true);
12    // для установки реакции на закрытие окна JFrame
13    jFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
14 }
15 }
```

Попробовали запустить программу.

Результат:



- Добавим метод `setSize`.

```
new *
6 ▶ public class Hello {
    new *
7 ▶ public static void main(String[] args) {
8     // объект класса JFrame с пустым конструктором
    new *
9     JFrame jFrame = new JFrame();
10    // окно становится видимым и пользователь может взаимодействовать с ним
11    jFrame.setVisible(true);
12    // для установки реакции на закрытие окна JFrame
13    jFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
14    // устанавливаем начальные размеры окна при запуске приложения
15    jFrame.setSize( width: 500, height: 300);
16 }
17 }
```

Запустите программу. Обратите внимание на место появления окна программы.

- Добавим метод `setLocation`.

```
new *
6 ▶ public class Hello {
    new *
7 ▶ public static void main(String[] args) {
8     // объект класса JFrame с пустым конструктором
    new *
9     JFrame jFrame = new JFrame();
10    // окно становится видимым и пользователь может взаимодействовать с ним
11    jFrame.setVisible(true);
12    // для установки реакции на закрытие окна JFrame
13    jFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
14    // устанавливаем начальные размеры окна при запуске приложения
15    jFrame.setSize( width: 500, height: 300);
16    // устанавливает начальное положение окна на экране
17    jFrame.setLocation( x: 500, y: 400);
18 }
19 }
```

- Заменяем метод `setLocation` методом `setBounds`.

```

new *
6 ▶ public class Hello {
    new *
7 ▶ public static void main(String[] args) {
8     // объект класса JFrame с пустым конструктором
    new *
9     JFrame jFrame = new JFrame();
10    // окно становится видимым и пользователь может взаимодействовать с ним
11    jFrame.setVisible(true);
12    // для установки реакции на закрытие окна JFrame
13    jFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
14    // устанавливает начальное положение и размер окна при запуске приложения.
15    jFrame.setBounds(x: 500, y: 400, width: 500, height: 300);
16
17 }
18 }
19

```

- Добавим набор инструментов Toolkit. С его помощью мы можем получить размер нашего окна через метод `getScreenSize`, содержащий ширину и высоту экрана. Он позволит нам расположить окошко четко по середине экрана.

```

7 ▶ public class Hello {
    new *
8 ▶ public static void main(String[] args) {
9     // объект класса JFrame с пустым конструктором
    new *
10    JFrame jFrame = new JFrame();
11    // окно становится видимым и пользователь может взаимодействовать с ним
12    jFrame.setVisible(true);
13    // для установки реакции на закрытие окна JFrame
14    jFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
15    // системные ресурсы, настройки для графического интерфейса
16    Toolkit toolkit = Toolkit.getDefaultToolkit();
17    // размер экрана (разрешение), используя метод getScreenSize из объекта Toolkit
18    Dimension dimension = toolkit.getScreenSize();
19    // устанавливает начальное положение и размер окна jFrame, центрированное посередине экрана
20    jFrame.setBounds(x: dimension.width / 2 - 250, y: dimension.height / 2 - 150, width: 500, height: 300);
21    // устанавливает заголовок окна jFrame
22    jFrame.setTitle("And the dream was so close");
23
24 }
25 }

```

- Добавим метод `setTitle`.

```

3
4 import javax.swing.JFrame;
5 import java.awt.Toolkit;
6 import java.awt.Dimension;
7
8 new *
9 public class Hello {
10     new *
11     public static void main(String[] args) {
12         // объект класса JFrame с пустым конструктором
13         new *
14         JFrame jFrame = new JFrame();
15         // окно становится видимым и пользователь может взаимодействовать с ним
16         jFrame.setVisible(true);
17         // для установки реакции на закрытие окна JFrame
18         jFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
19         // устанавливает размер окна в пикселях
20         jFrame.setSize( width: 500, height: 300);
21         // устанавливает начальное положение окна на экране
22         jFrame.setLocation( x: 500, y: 400);
23         // устанавливает заголовок в верхней части окна
24         jFrame.setTitle("And the dream was so close");
25     }
26 }

```

Попробовали запустить программу.

Результат:

