

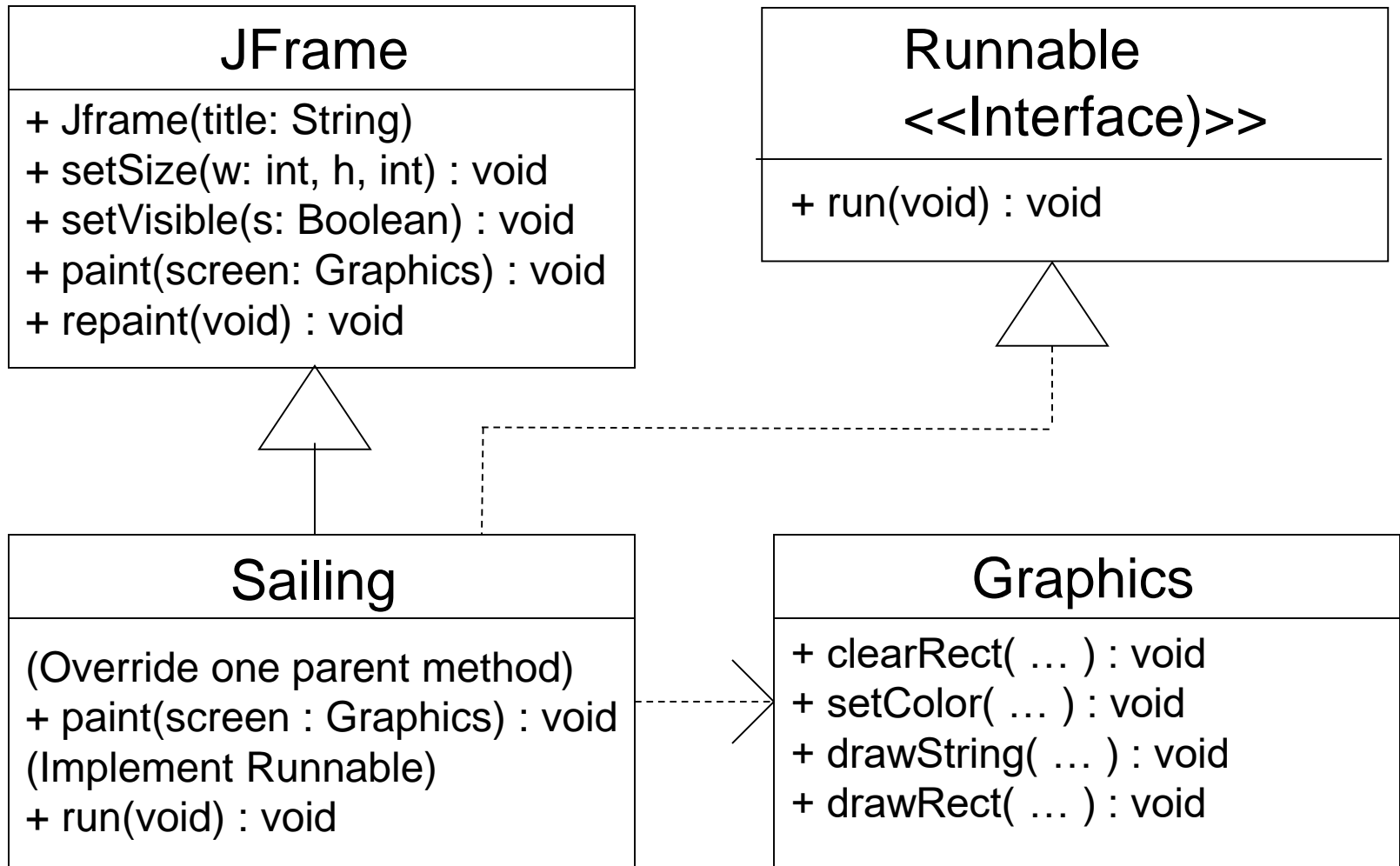
Java GUI Animation

- Example GUI with Animation
 - Sailing Class
 - Extending javax.swing.JFrame Class
 - Implementing Runnable Interface
 - Thread Class (Supports multi-threading)
 - Introduction to Graphics Class Methods
- Course Evaluation
- Reading for this lecture: Review

Java GUI Animation

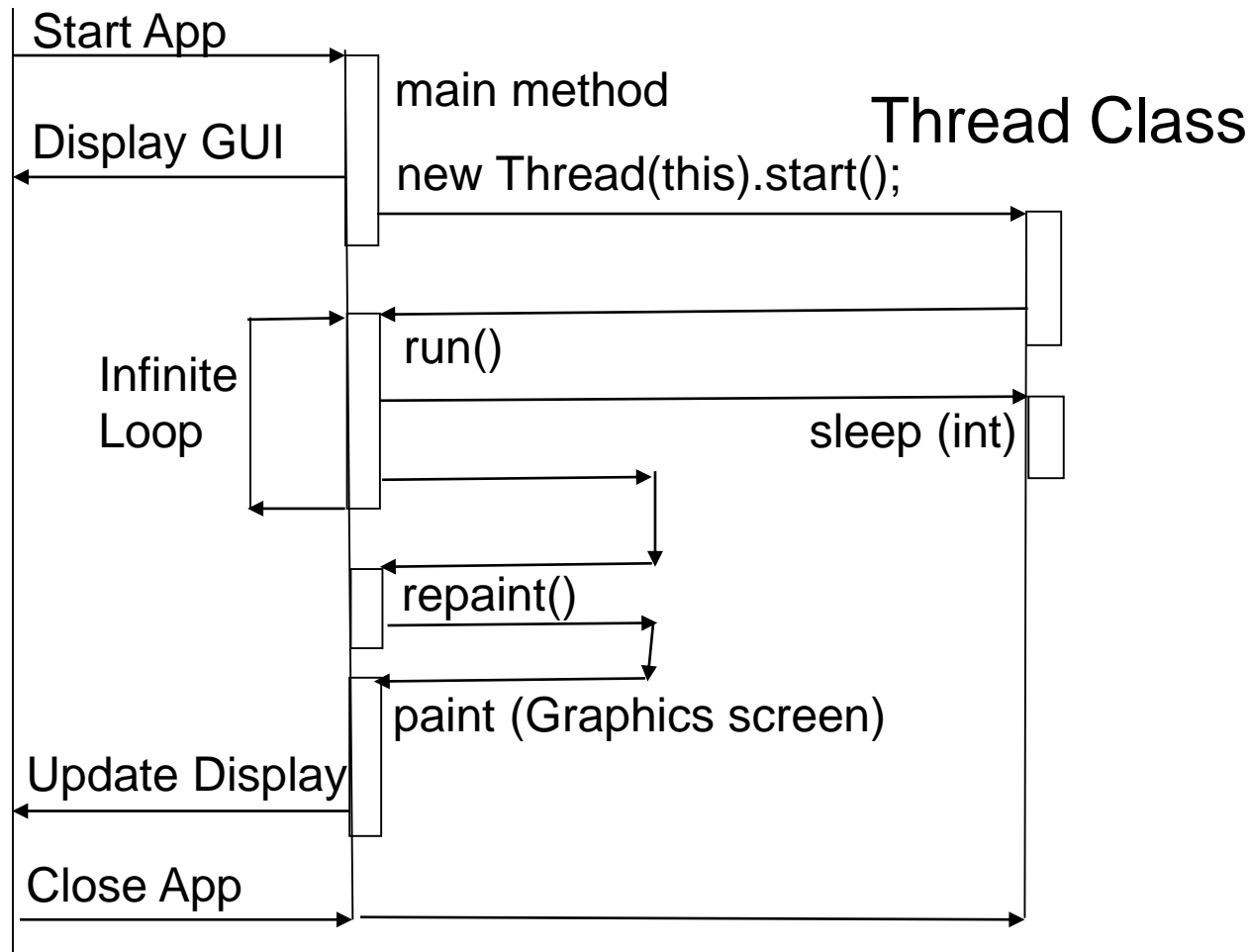
- Demo under Dr Java
- A link to the code is on the syllabus page

UML Class Diagram for Sailing



UML Sailing Sequence Diagram

User Sailing Class (including parent JFrame methods)



Sailing Class Design Explanation

- Sailing class extends JFrame class
 - All JFrame methods are inherited by Sailing
 - Overrides the paint method to display sailboat(s)
 - The animationState variable controls paint display
- Sailing class main method
 - Instantiates itself as a child of JFrame class
 - Calls its execute method to complete main thread
 - Now the main thread is out of static context

Sailing Class Design Explanation

- Sailing class constructor
 - Sets title bar text via parent's constructor
- Sailing class execute method
 - Completes initialization and display of frame
 - Instantiates a new Thread
 - Starts it with a reference to itself
 - Returns to main terminating the main thread
- The new Thread
 - Calls Sailing class run method via the reference

Sailing Class Design Explanation

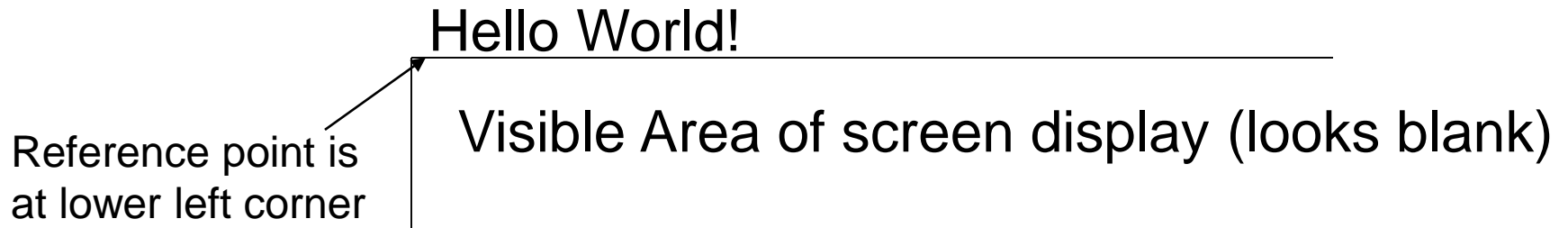
- The Sailing class run method
 - Loops forever
 - Calls its parent JFrame class repaint method
 - Delays for 1 second
 - Updates animationState variable's value
- JFrame class repaint method
 - Calls paint method with the frame's Graphics object
 - Sailing class paint method redraws display with the position of the sailboat and the waves depending on the value of the animationState variable

Graphics Class Methods

```
public void paint (Graphics screen)
{
    // look up/study the Graphics class methods
    // clear the GUI screen area
    screen.clearRect(0,0,this.getWidth(),
                    this.getHeight());
    // pick up a red pen for drawing
    screen.setColor(Color.RED);
    // Ubiquitous "Hello World" in upper left
    screen.drawString("Hello World!", 0, 10);
    // and surround it with a rectangle
    screen.drawRect(0, 0, 100, 10);
}
```

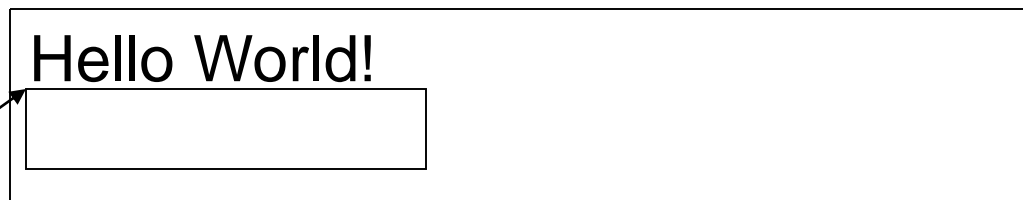

Graphics Class Methods

- Look carefully at the reference point for drawing something with each graphics method or you can have a problem
Example: `drawString("Hello World", 0, 0)`
- If drawn at 0,0: The text box will be above the visible area of the screen display



Graphics Class Methods

- Use `drawString` (“Hello World!”, 0, 10)
- But to surround the text with a rectangle, use `drawRect(0, 0, 100, 10)`
- If drawn at 0, 10, 100, 20, the rectangle will be drawn below the text – not around it.



Reference point is
at upper left corner

Course Evaluation

- Need a student volunteer to collect evaluations