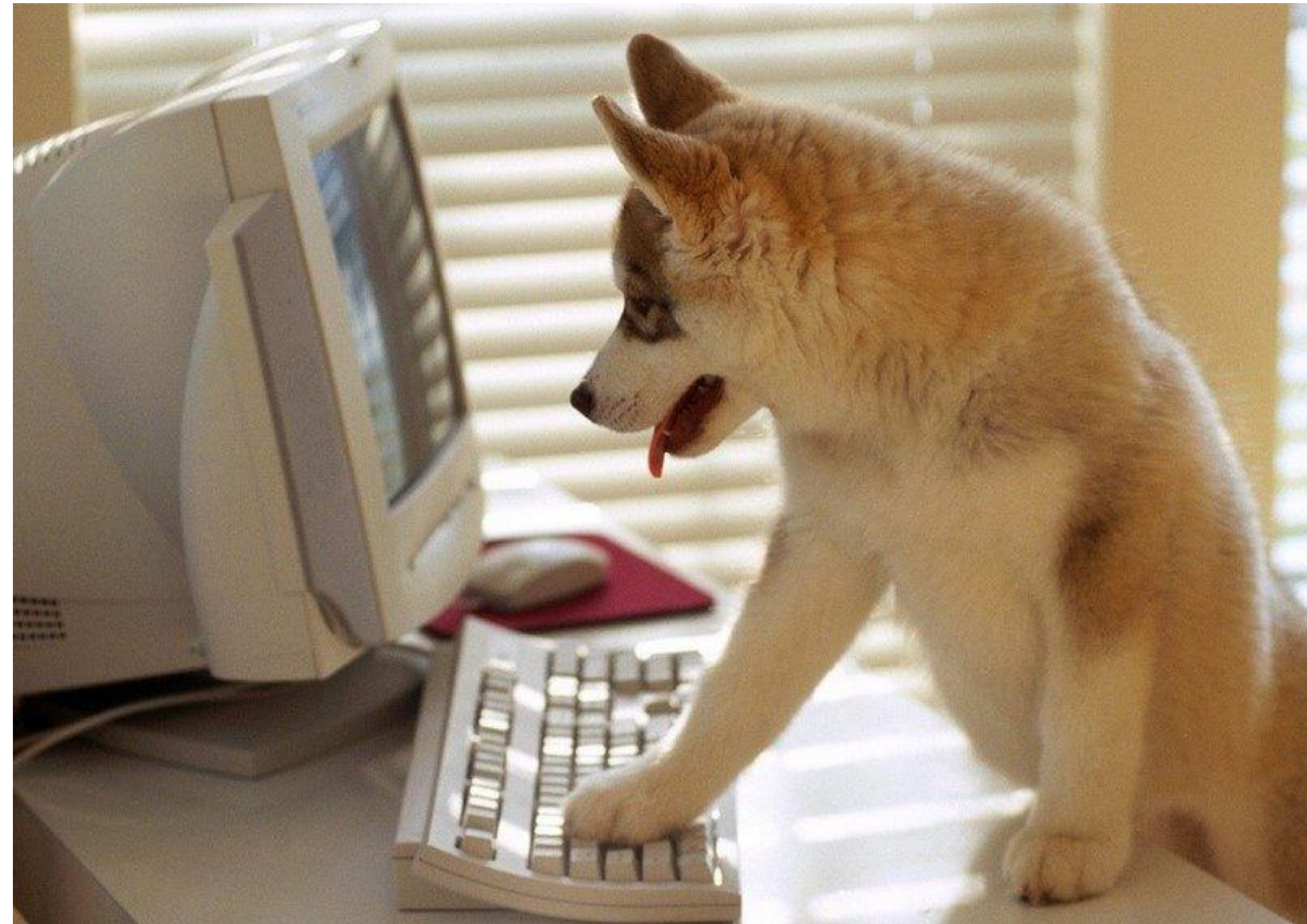


# Java Programming



# Organizational Stuff

18.03.: Structures

19.03.: Methods

20.03.: Recursion

21.03.: Arrays

22.03.: Strings

-----

25.03.: OOP1

26.03.: OOP2

27.03.: Generics

28.03.: Exceptions & **Testing**

29.03.: GUI

# Organizational Stuff

18.03.: Structures

19.03.: Methods

20.03.: Recursion

21.03.: Arrays

22.03.: Strings

-----

25.03.: OOP1

26.03.: OOP2

27.03.: Generics

28.03.: Exceptions & **Testing**

29.03.: GUI

Deadline for **all** assignments: 01 April 2019 9:15AM

Deadline for Miniproject: 02 May 2019 11:59 PM

# Organizational Stuff

## How to get a grade:

1. Complete **all** assignments **in time**
  - a. Deadline fixed in Github, late submissions will not reach us and therefore will not be considered!
2. Complete the Miniproject **in time**
3. Present your Miniproject

# Organizational Stuff

## Miniprojects

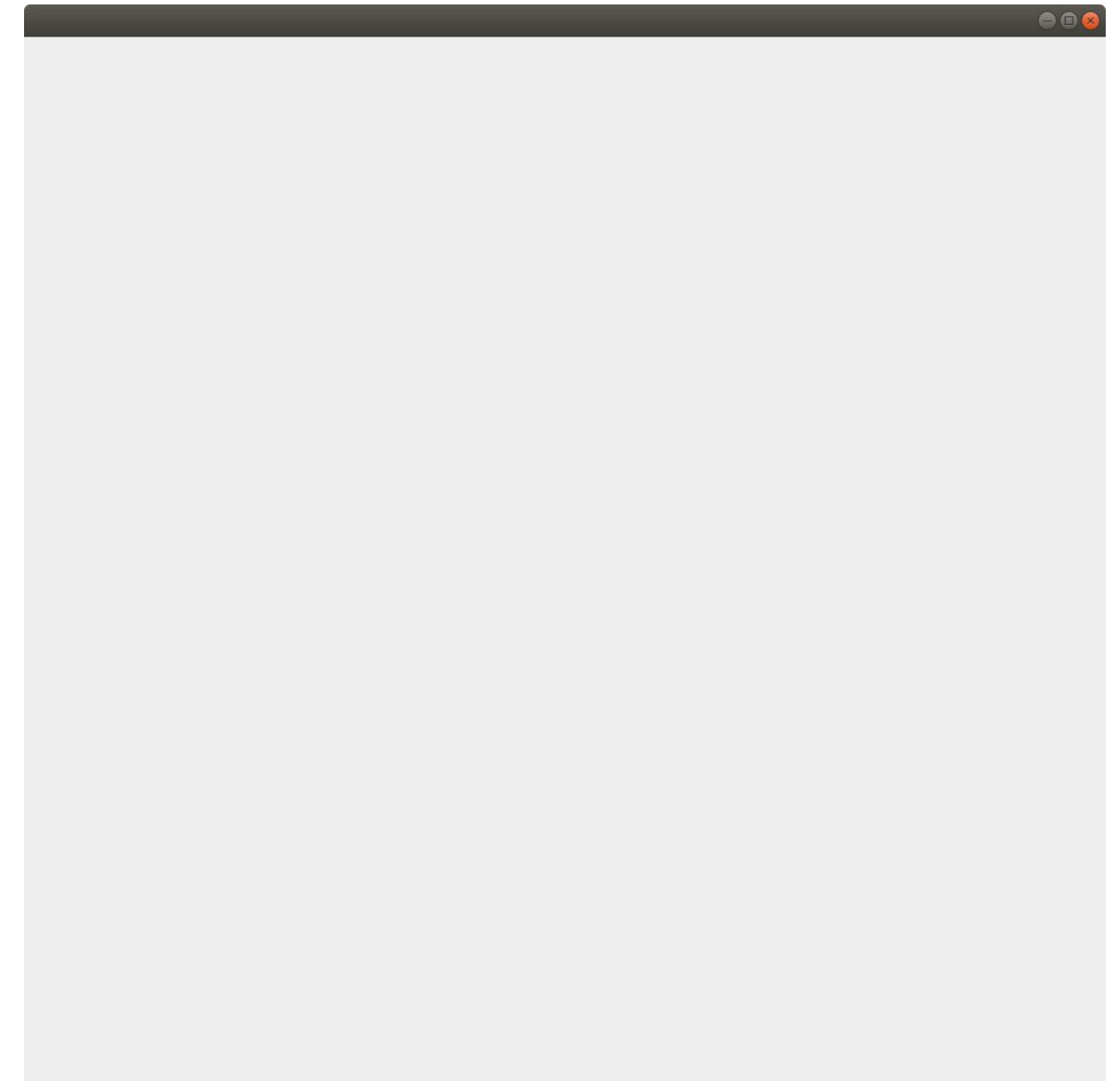
1. **After** submitting the assignments, we will assign you (and your team partner, if you have one) to a github repository.
2. There is already a starter code provided in the repository (usually a GUI)
3. A \*.pdf file on how to structure your documentation will be added to your repositories as well
4. A Miniproject consists of three parts:
  - a. Code
  - b. Documentation
  - c. Presentation
5. You will only receive a grade if you complete **all three parts in time!**

```
import javax.swing.*;
import java.awt.*;

public class Example extends JFrame {

    public Example() {
        setSize(1000,1000);
    }

    public static void main(String[] args) {
        EventQueue.invokeLater(() ->{
            Example ex = new Example();
            ex.setVisible(true);
        });
    }
}
```



# GUI

```
import javax.swing.*;
import java.awt.*;

public class Example extends JFrame {

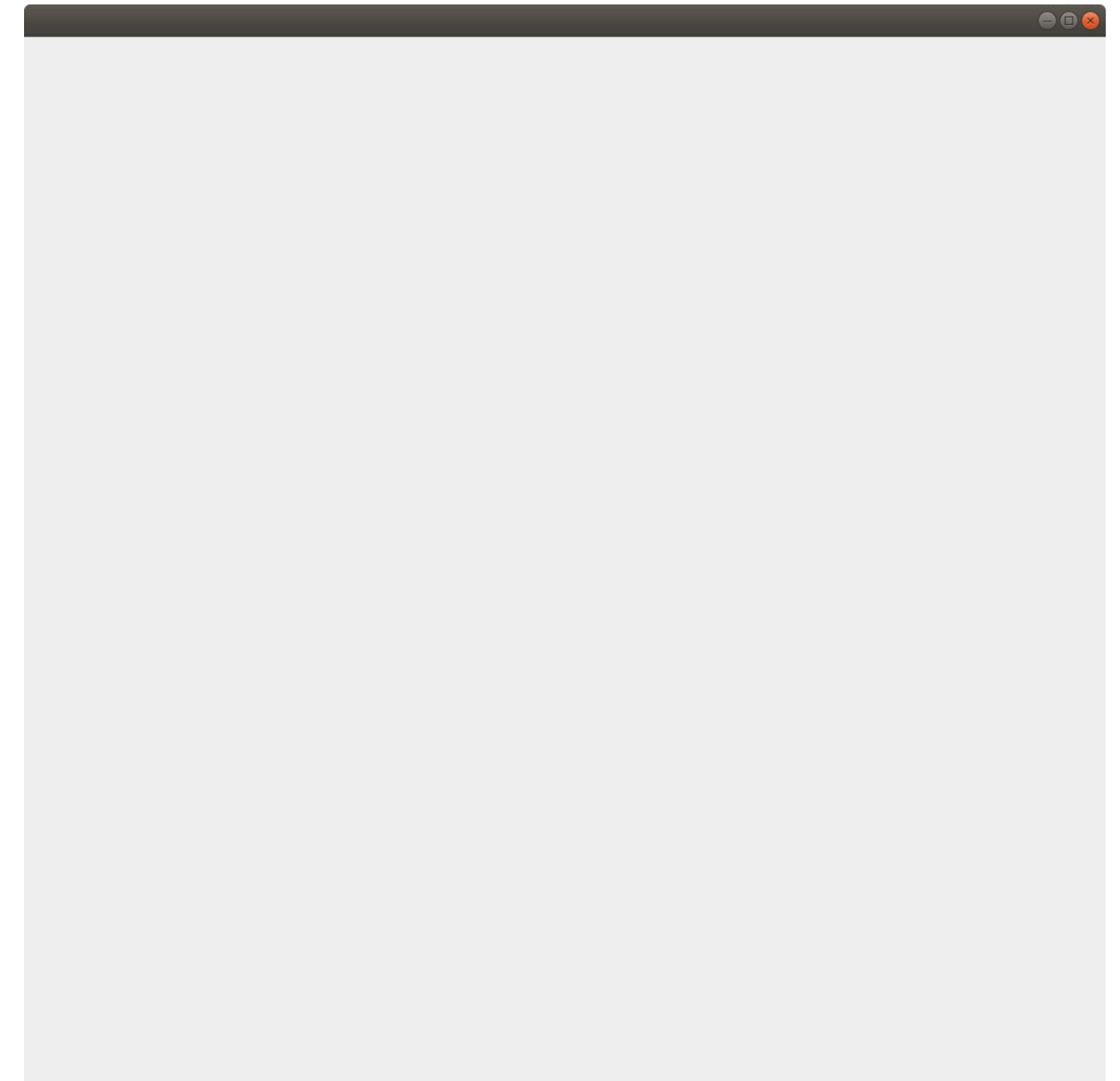
    private JPanel pane;

    public Example() {
        initUI();
        setSize(1000,1000);
    }

    private void initUI() {
        pane = (JPanel) getContentPane();
    }

    public static void main(String[] args) {
        ...
    }

}
```





# GUI

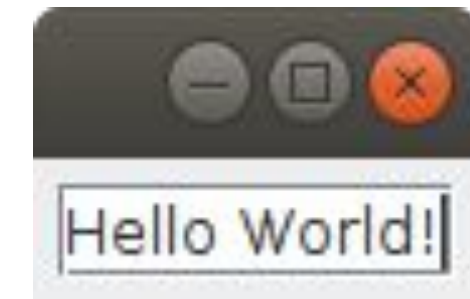
```
...
public class Example extends JFrame {

    private JPanel pane;
    private JTextField tf;

    public Example() {
        initUI();
    }

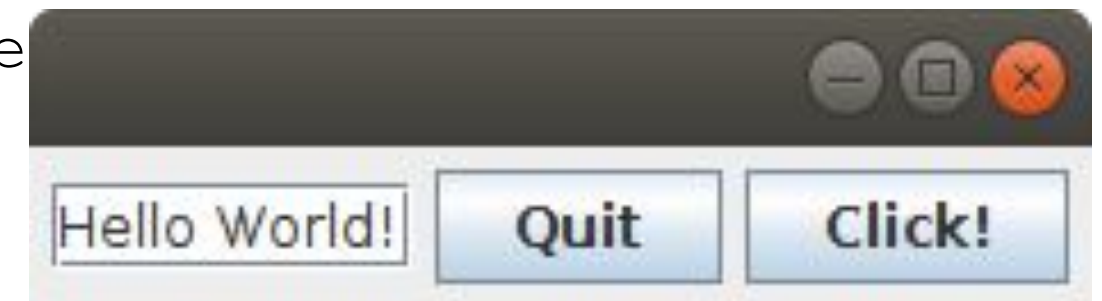
    private void initUI() {
        pane = (JPanel) getContentPane();
        tf = new JTextField();
        tf.setText("Hello World!");
        pane.add(tf);
        pane.setLayout(new FlowLayout());
        pack();
    }

    ...
}
```





```
...  
private JButton click;  
...  
private void initUI() {  
    pane = (JPanel) getContentPane();  
    tf = new JTextField();  
    tf.setText("Hello World!");  
    quitButton = new JButton("Quit");  
    quitButton.addActionListener((event) -> System.e  
    click = new JButton("Click");  
    click.addActionListener((event) -> onSubmit());  
    pane.add(tf);  
    pane.add(quitButton);  
    pane.add(click);  
    pane.setLayout(new FlowLayout());  
    pack();  
}  
private void onSubmit() {  
    tf.setText("Click!");  
}
```



**Today's Assignment:**

<https://classroom.github.com/a/E614h8aY>

