



# WEEK 4 DAY 2 REVIEW

Swing/Scanner/MC questions

# Scanner

2

- There are various ways to read input from the keyboard, the `java.util.Scanner` class is one of them.
- **Scanner** class breaks the input into tokens using a delimiter that is whitespace by default
- Scanner class must be imported from `java.util`
- `Scanner sc = new Scanner(System.in);`
- `System.out.println("Enter your guess");`
- `String guess=sc.next();`

# Swing-JFrame

3

- Main object interface is a FRAME (JFrame)
  - ▣ Can use constructor to set title or setTitle(String)
  - ▣ setSize(int int)
  - ▣ pack() sizes based on components
  - ▣ setVisible(Boolean)
  - ▣ setDefaultCloseOperation()
    - **EXIT\_ON\_CLOSE**
    - **DISPOSE\_ON\_CLOSE**
    - **DO\_NOTHING\_ON\_CLOSE**
    - **HIDE\_ON\_CLOSE**

# Components

4

- ❑ Superclass=JComponent
- ❑ setEnabled(Boolean)
- ❑ isEnabled()
- ❑ setVisible()
- ❑ isVisible
- ❑ setSize(int, int)
- ❑ getSize()returns dimension object

# Component-JButton

5

- **JBButton(Swing)** – A button labeled with the specific text
- **JBButton(Icon)** – A button that displays the specified graphical icon
- **JBButton(String, Icon)** – A button with the specified text and graphical icon

# Component-JLabel

6

- **JLabel(String)** – a label with specified text
- **JLabel(String, int)** – a label with specified text and alignment
- **JLabel(String, Icon, int)** – a label with a specified text, icon and alignment
- **SwingConstants** interface that are used to indicate alignment: **LEFT**, **CENTER** and **RIGHT**
- **setText()** **getText()** **setIcon()** **getIcon()**

# Component-JText

7

- **TEXT FIELD** is a location on an interface where a user can enter and modify text
- **TextField()** – An empty text field
- **TextField(int)** – A text field with a specified width
- **TextField(String, int)** – A text field with the specified text and width
- **setEditable(Boolean) isEditable(Boolean)**
- Password fields hide the characters the user enters. They are represented by the **JPasswordField** class which is a subclass of **TextField**

# Component-JText (2)

8

- **getText(), getSelectedText() and setText()**
- **append(String)** method that adds the specified text at the end of the current text
- **insert(String, int)** method to insert the specified text at the insertion point
- **setLineWrap(boolean)**
- **setWrapStyleWord(boolean)**



# Scrolling Panes

9

- A scrolling pane is associated with a component  
**JScrollPane(Component)** – A scrolling pane that contains the specified component
- **JScrollPane(Component, int, int)** – A scrolling pane with the specified component, vertical scrollbar configuration and horizontal scroll bar configuration
  - ▣ **VERTICAL\_SCROLLBAR\_ALWAYS**
  - ▣ **VERTICAL\_SCROLLBAR\_AS\_NEEDED**
  - ▣ **VERTICAL\_SCROLLBAR\_NEVER**

# Question 1

10

Which of the following loops will execute the body of loop even when condition controlling the loop is initially false?

- a) do-while
- b) while
- c) for
- d) None of the mentioned

# Question 2

11

What is the process of defining more than one method in a class differentiated by method signature?

- a) Function overriding
- b) Function overloading
- c) Function doubling
- d) None of the mentioned

# Question 3

12

Which method can be defined only once in a program?

- a) main method
- b) finalize method
- c) static method
- d) private method

# Question 4

13

Which of these access specifiers must be used for `main()` method?

- a) private
- b) public
- c) protected
- d) None of the mentioned

# Question 5

14

Which of these class is superclass of every class in Java?

- a) String class
- b) Object class
- c) Abstract class
- d) ArrayList class

# Question 6

15

Which of these keywords can be used to prevent inheritance of a class?

- a) super
- b) constant
- c) Class
- d) final

# Question 7

16

What is the use of try & catch?

- a) It allows us to manually handle the exception.
- b) It allows to fix errors.
- c) It prevents automatic terminating of the program in cases when an exception occurs.
- d) All of the mentioned.