1. In the following code, what does the number 100 represent?

1 / 1 балл

```
my_button = frame.add_button("My Label", button_handler, 100)
```

Use the CodeSkulptor documentation to look it up.

○ Vertical position of the button in pixels

○ Height of the button in pixels

○ Horizontal position of the button in pixels

● Width of the button in pixels

✓ **Правильно**

2. How many control objects are allowed in a frame?

1 / 1 балл

● Unlimited, i.e., 0 or more

○ 1

○ 0

✓ **Правильно**

Correct.

3. In SimpleGUI, one kind of object can be added to the control panel that doesn't allow any handler. Thus, this object can't respond to anything. What kind of object is that?

0 / 1 балл

Look at the documentation for SimpleGUI control objects.

● Frame

○ Input field

○ Label

○ Title

○ Canvas

○ Button

☒ **Неправильно**

That isn't a control object.

4. When you enter text into an input field and press enter, the text is passed to the input field's event handler. What is the data type of the text?

1 / 1 балл

● A string

○ A string or a number, depending on the text entered

○ A number

Правильно

The entered data is a string. The text entered into an input field is always passed to the input handler as a string even if the text corresponds to a number.

5. Consider the following conditional statement.

1 / 1 балл

```
if p == False:
    return False
elif q == False:
    return False
else:
    return True
```

That is equivalent to which of the following simpler statements?

Try to reason logically about each of the statements, but also try each in CodeSkulptor.

□
```
return p and (not q)
```

☑
```
return q and p
```

Правильно

□
```
return p or q
```

□
```
return (not p) and (not q)
```

6. Which of the following describes the mistake in the following code?

1 / 1 балл

```
def volume_cube(side):
    """ Returns the volume of a cube, given the length of its side. """
    print side ** 3

s = 5
print "The volume of a cube with sides", s, "long is", volume_cube(s), "."
```

● The function should return, not print, its result.

○ The call to `volume_cube` shouldn't be within a `print` statement. More generally, function calls usually shouldn't be within `print` statements.

○ All of the printing should be done within the function.

✓ **Правильно**

Correct. In most cases, functions should return their computed results. Furthermore, the documentation string here specifies that it should be returning that value.

7. What kind of errors can happen if you are missing a needed `global` declaration in one of your function definitions? For this question, you need only consider the case where the problem is in the function that is missing the `global` declaration.

1 / 1 балл

If you are having trouble with this question, watch this week's Programming Tips video again.

□ `NameError`

□ `AttributeError`

☑ An incorrect computation that generates no error message

✓ **Правильно**

If you only assign to the variable, without trying to use its current value, you won't get any error message. Instead, Python assumes the variable is local, which might lead to an unexpected result.

□ `SyntaxError`

☑ `Error: local variable '...' referenced before assignment`

✓ **Правильно**

8. Which of the following function definitions are in the recommended code style?

☑

```python
def f(x, y):
    """ Add the two inputs. """
    return x + y
```

Правильно

☑

```python
def f(x,y):
    """ Add the two inputs. """
    return x + y
```

Этот вариант не должен быть выбран

This does not follow recommended style, there should be a space after the comma.

□

```python
def f(x, y):
    return x + y # Add the two inputs.
```

☑

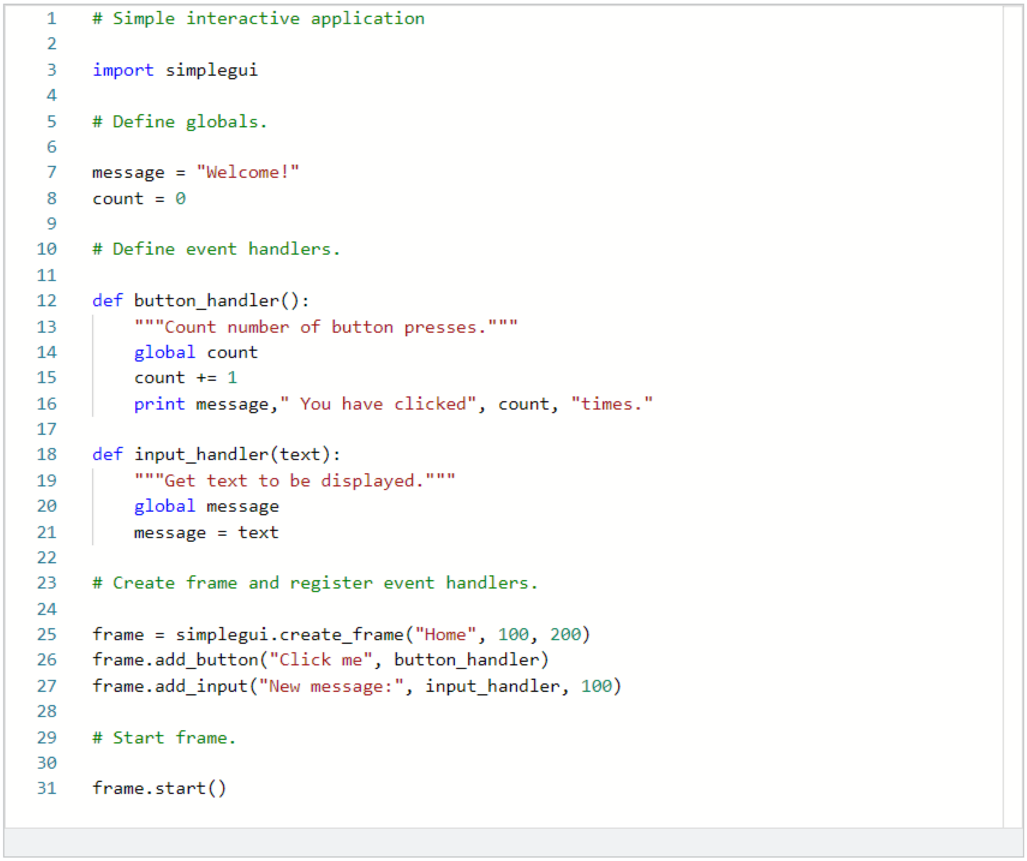
```python
def f (x, y):
    """ Add the two inputs. """
    return x + y
```

Этот вариант не должен быть выбран

This does not follow recommended style, there shouldn't be a space before the parenthesis.

9. Cut and paste the following code into CodeSkulptor. Run it and make an attempt to understand how it works.

1 / 1 балл



```
# Simple interactive application

import simplegui

# Define globals.

message = "Welcome!"
count = 0

# Define event handlers.

def button_handler():
    """Count number of button presses."""
    global count
    count += 1
    print message," You have clicked", count, "times."

def input_handler(text):
    """Get text to be displayed."""
    global message
    message = text

# Create frame and register event handlers.

frame = simplegui.create_frame("Home", 100, 200)
frame.add_button("Click me", button_handler)
frame.add_input("New message:", input_handler, 100)

# Start frame.

frame.start()
```

We'd like to modify the code so that the count is reset to zero whenever a new message is entered. Where would you need to modify this code to implement this change?

○ Add an assignment to `count` at the end of this code.

○ Add an assignment to `count` in the event handler for the button.

● Add an assignment to `count` in the event handler for the input field. Also add a `global count` declaration there.

○ Add an assignment to `count` in the initialization of global variables.

✓ **Правильно**

10. In the game “Guess the number”, what is the minimum number of guesses necessary to guarantee that the guesser can always win if the secret number is chosen in `range(0, 400)`?

1 / 1 балл

Review the mini-project description for "Guess the number" if you are having trouble with this problem.

○ 8 guesses

● 9 guesses

○ 10 guesses

○ 12 guesses

○ It's impossible to guarantee that you can always win at “Guess the number“.

Правильно