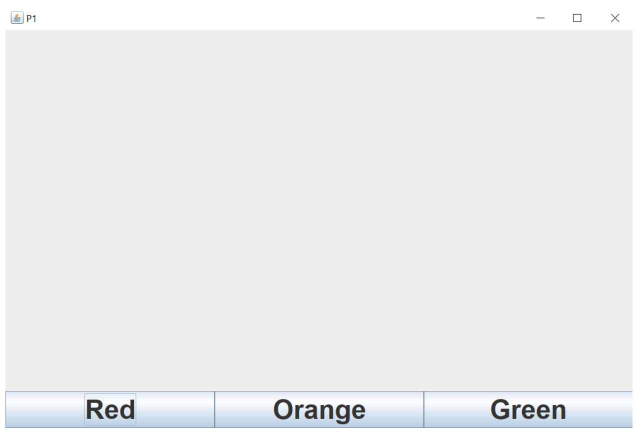
QUEENS COLLEGE   Department of Computer Science
CSCI 212   Second Midterm Exam   Fall 2019   11.26.19

Instructor: Alex Ryba
09.05am – 09.55am, Tuesday, November 26, 2019

1. Consider the following Java class that is used to create and display a GUI. The class has exactly three methods, two of which you will write.

```
public class P1 extends JFrame {

  JPanel center;  // instance variable represents the large empty central area

  // a helper method to create a button and add it to a specified panel
  void makeButton(JPanel panel, String buttonName) {
    JButton b = new JButton(buttonName);
    panel.add(b);
  }

  // There is a constructor method whose code has been omitted

  // There is a main method whose code has been omitted

}
```

The GUI window has title P1 and displays as follows:



(a) Write the main method.

Answer:

```
  public static void main(String args[]) {
    P1 gui = new P1();
    gui.setVisible(true);
  }
```

(b) Write the constructor. Your code must apply the method `makeButton` to make and add the three buttons.

Answer:

```
  public P1() {
    super();
    setTitle("P1");
    setSize(1200, 800);

    Container contentPane = getContentPane();
    contentPane.setLayout(new BorderLayout());

    center = new JPanel();
    contentPane.add(center, BorderLayout.CENTER);

    JPanel buttonPanel = new JPanel();
    contentPane.add(buttonPanel, BorderLayout.SOUTH);
    buttonPanel.setLayout(new GridLayout(1, 3));

    makeButton(buttonPanel, "Red");
    makeButton(buttonPanel, "Orange");
    makeButton(buttonPanel, "Green");
  }
```

2. In this problem you will provide modifications to the code given in Problem 1 so that the center panel changes color to Red, Orange or Green as the three buttons are clicked by a user. Your modifications should add one standard method to `class P1` to program all responses to user actions.

   (a) The title line of `class P1` must be modified. Write the new title line that should be used for the class.

   Answer:

```
public class P1 extends JFrame implements ActionListener
```

   (b) Write the full code for one new method that should be added to `class P1` to allow it to respond to button clicks.

   Answer:

```
public void actionPerformed(ActionEvent e) {
  if (e.getActionCommand().equals("Red")) center.setBackground(Color.RED);
  if (e.getActionCommand().equals("Orange")) center.setBackground(Color.ORANGE);
  if (e.getActionCommand().equals("Green")) center.setBackground(Color.GREEN);
}
```

   (c) A change is also needed to make each of the button objects listen for user interactions. Add one line of code at the end of the method `makeButton` so that this change is applied for each button as it is created. Write this one line of code.

   Answer:

```
b.addActionListener(this);
```

3. (a) Write a `class EmptyArrayException` whose objects are exceptions but are not run time exceptions. You should provide a constructor with a `String` parameter. No other methods should be given.

Answer:

```
public class EmptyArrayException extends Exception {
  public EmptyArrayException(String s) {
    super(s);
  }
}
```

Now consider the following `class P3` that has three methods `main`, `readMyArrayList` and `average`.

```
public class P3 {
  public static void main(String args[]) {
    Scanner input = new Scanner(System.in);
    for (int i = 0; i < 5; i++) {
      ArrayList<Integer> list = readMyArrayList(input);
      System.out.println(average(list));
    }
  }

  static ArrayList<Integer> readMyArrayList(Scanner input) {
    ArrayList<Integer> answer = new ArrayList<>();
    System.out.println("How many elements?");
    int size = input.nextInt();
    for (int i = 0; i < size; i++) answer.add(input.nextInt());
    return answer;
  }

  static double average(ArrayList<Integer> list) {
    int sum = 0;
    for (int i = 0; i < list.size(); i++) sum += list.get(i);
    return ((double) sum) / list.size();
  }
}
```

(b) Modify the method `average` method to throw an `EmptyArrayException` whenever its list parameter has size 0. Give the complete code for the modified method.

Answer:

```
  static double average(ArrayList<Integer> list)  throws EmptyArrayException {
    if (list.size() == 0) throw new EmptyArrayException("Empty array");
    int sum = 0;
    for (int i = 0; i < list.size(); i++)
      sum += list.get(i);
    return ((double) sum) / list.size();
  }
```

(c) Modify the main method to catch any exception thrown by the method `average`. After catching an exception, report an illegal empty list to the user and make an adjustment so that the for loop will have an extra iteration. This will make sure that the loop really does find 5 averages (if necessary by making the user enter extra data). To answer this part, write complete code for a modified main method.

Answer:

```
  public static void main(String args[]) {
    Scanner input = new Scanner(System.in);
    for (int i = 0; i < 5; i++) {
      ArrayList<Integer> list = readMyArrayList(input);
      try {
        System.out.println(average(list));
      } catch (EmptyArrayException e) {
        System.out.println("That's empty!");
        i--;
      }
    }
  }
```

4. Consider the following class:

```
public class B implements A {
  public void printMessage() {
    printAnswer();   System.out.println("unknown");
  }
  void printAnswer() {  System.out.print("The answer is ");  }
  public void printMessage(int x) {
    for (int i = 0; i < x; i++) printMessage();
  }

  public static void main(String args[]) {
    A a = new C();
    B b = new B();
    a.printMessage();
    b.printMessage();
    b = (B) a;
    b.printMessage(2);
  }
}
```

(a) Write an interface A to require the two `printMessage` methods.

Answer:

```
public interface A {
  void printMessage();
  void printMessage(int x);
}
```

(b) Write a subclass C of B that has only one method. This method should override the method `public void printMessage()` and make the overridden method print: `The answer is 42` .

Answer:

```
public class C extends B {
  public void printMessage() {
    printAnswer();
    System.out.println(42);
  }
}
```

(c) Suppose that interface A and class C are as described in parts (a) and (b). What is the output when the main method of class B is run?

Answer:

```
The answer is 42
The answer is unknown
The answer is 42
The answer is 42
```