**Code for Assignment #4**

**Eric Dockery**

**Problem #1**

**public** **class** Student {

**private** String lastName;

**private** **static** **int** *score1*;

**private** **static** **int** *score2*;

**private** **static** **int** *score3*;

**private** **static** **int** *average*;

**private** String firstName;

**private** Address homeAddress, schoolAddress;

**public** Student(String first, String last, **int** score\_1,**int** score\_2,**int** score\_3){

**this**.firstName = first;

**this**.lastName = last;

*score1* = score\_1;

*score2* =score\_2;

*score3*= score\_3;

}

**public** String toString()

{

String result;

result = firstName + "" +lastName + "\n";

result +="Home Address: \n" + homeAddress+ "\n";

result +="School Address: \n" +schoolAddress;

result += "TEST Scores : \n" + *score1*+ "\n"+*score2*+"\n"+*score3*+"\n";

result += "Average: \n"+*average*;

**return** result;

}

**public** **void** setTestScore(**int** Test, **int** score){

**if** (Test==1 ){

*score1* = score;

}

**else** **if** (Test ==2){

*score2* = score;

}

**else**{

*score3* = score;

}

}

**public** **static** **int** setAverage (){

*average* = (*score1*+ *score2*+ *score3*)/3;

**return** *average*;

}

**public** String getLastName() {

**return** lastName;

}

**public** **void** setLastName(String lastName) {

**this**.lastName = lastName;

}

**public** String getFirstName() {

**return** firstName;

}

**public** **void** setFirstName(String firstName) {

**this**.firstName = firstName;

}

}

**public** **class** Driver {

**public** **static** **void** main(String[] args)

{

Course course = **new** Course("CS");

course.addStudent(**new** Student("Eric", "Dockery", 84,95,76));

**int** x = Student.*setAverage*();

course.addStudent(**new** Student("Joe", "Schmo", 88,99,100));

**int** y = Student.*setAverage*();

course.addStudent(**new** Student("Fred", "Jones", 88,45,76));

**int** z = Student.*setAverage*();

course.addStudent(**new** Student("Arlan", "Dock", 0,0,0));

**int** w = Student.*setAverage*();

course.addStudent(**new** Student("Andrew", "Geni", 100,100,100));

**int** e = Student.*setAverage*();

course.Roll();

**int** Exams= course.Total\_Average( x, y, z, w, e);

System.*out*.println("Test Average "+ Exams);

course.addStudent(**new** Student("Andrew", "Geni", 100,100,100));

course.Roll();

}

}

**import** java.util.ArrayList;

**import** java.util.List;

**public** **class** Course {

**private** String This\_Class;

**private** List<Student> students;

**public** **int** Test\_Average;

**private** **int** num= 0;

Course(String Course){

**this**.This\_Class = Course;

students = **new** ArrayList<Student>();

}

**public** **void** addStudent( Student student)

{

students.add(student);

num++;

}

**public** **void** Roll(){

System.*out*.println("Course: " +This\_Class );

System.*out*.println("Students");

**if** (num <= 5){

**for**(Student student: students){

System.*out*.println(student.getFirstName()+" "+student.getLastName());

}

}

**else**

{

System.*out*.println("There have been too many students added");

}

}

**public** **int** Total\_Average(**int** studenttestaverage1, **int** studenttestaverage2, **int** studenttestaverage3, **int** studenttestaverage4, **int** studenttestaverage5){

Test\_Average = (studenttestaverage1 +studenttestaverage2 +studenttestaverage3+studenttestaverage4+studenttestaverage5)/5;

**return** Test\_Average;

}

}

**Problem #2**

**public** **class** PriorityDriver {

**public** **static** **void** main(String[] args){

Task This\_Task = **new** Task();

This\_Task.setPriority(1);

System.*out*.println("The priority for sleep is "+ This\_Task.getPriority());

Task New\_Task = **new** Task();

New\_Task.setPriority(-2);

System.*out*.println("The priority for classwork is "+ New\_Task.getPriority());

}

}

**public** **class** Task **implements** Priority{

**private** **int** Priority\_Level;

**public** **void** setPriority(**int** priority) {

Priority\_Level = priority +3;

}

**public** **int** getPriority() {

**return** Priority\_Level;

}

}

**public** **interface** Priority{

**public** **void** setPriority(**int** priority);

**public** **int** getPriority();

}

**Problem #3**

**import** javax.swing.JFrame;

**public** **class** PigLatin {

**public** **static** **void** main(String[] args){

//pig latin program

JFrame frame = **new** JFrame("PIG LATIN TRANSLATOR");

frame.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);

frame.getContentPane().add(**new** TranslatorGUI());

frame.pack();

frame.setVisible(**true**);

}

}

**import** java.awt.\*;

**import** java.awt.event.ActionEvent;

**import** java.awt.event.ActionListener;

**import** java.util.Scanner;

**import** javax.swing.\*;

@SuppressWarnings("serial")

**public** **class** TranslatorGUI **extends** JPanel{

**private** JButton To\_Pig\_Latin;

**private** JLabel PIG\_Latin\_Label;

**private** JTextField THIS\_TO\_PIG\_LATIN;

**public** TranslatorGUI(){

// gui for pig latin program

setLayout(**new** BoxLayout(**this**, BoxLayout.*Y\_AXIS*));

setBackground(Color.*pink*);

// Prompt of what the program does

JLabel Prompt = **new** JLabel ("Enter a sentence (no punctuation):");

// Text field that will be translated

THIS\_TO\_PIG\_LATIN = **new** JTextField(50);

THIS\_TO\_PIG\_LATIN.setEditable(**true**);

THIS\_TO\_PIG\_LATIN.setBorder(BorderFactory.*createLoweredBevelBorder*());

// Button to change the Translation Label to correct information

To\_Pig\_Latin = **new** JButton("TO Pig Latin!!");

To\_Pig\_Latin.addActionListener(**new** Pig\_Listener());

PIG\_Latin\_Label= **new** JLabel();

//adding the fields and buttons to the gui

add(Prompt);

add(THIS\_TO\_PIG\_LATIN);

add(To\_Pig\_Latin);

add(PIG\_Latin\_Label);

add(Box.*createRigidArea*(**new** Dimension(0,20)));

}

**public** **class** Pig\_Listener **implements** ActionListener {

@Override

**public** **void** actionPerformed(ActionEvent event) {

**if** (event.getSource() == To\_Pig\_Latin);

String text = THIS\_TO\_PIG\_LATIN.getText();

text=text.trim();

text=text.toLowerCase();

PIG\_Latin\_Label.setText(PigLatinTranslator(text));

}

**private** String PigLatinTranslator(String sentence) {

// **TODO** Auto-generated method stub

String result ="";

sentence= sentence.toLowerCase();

@SuppressWarnings("resource")

Scanner scan = **new** Scanner (sentence);

**while** (scan.hasNext()){

result +=translateWord (scan.next());

result += "";

}

**return** result;

}

**private** String translateWord(String word){

String result ="";

**if** (beginsWithVowel(word)){

result = word +"yay";

}

**else** **if** (beginsWithBlend(word)){

result =word.substring(2)+word.substring(0,2)+ "ay";

}

**else**{

result = word.substring(1) +word.charAt(0)+"ay";

}

**return** result;

}

**private** **boolean** beginsWithVowel (String word){

String vowels = "aeiou";

**char** letter = word.charAt(0);

**return**(vowels.indexOf(letter) != -1);

}

**private** **boolean** beginsWithBlend( String word){

**return**(word.startsWith("bl")||word.startsWith("sc")||word.startsWith("br")||word.startsWith("sh")||word.startsWith("ch")||word.startsWith("sk")||word.startsWith("cl")||word.startsWith("sl")||word.startsWith("cr")||word.startsWith("sn")||word.startsWith("dr")||word.startsWith("sm")||word.startsWith("dw")||word.startsWith("sp")||word.startsWith("fl")||word.startsWith("sq")||word.startsWith("fr")||word.startsWith("st")||word.startsWith("gl")||word.startsWith("sw")||word.startsWith("gr")||word.startsWith("th")||word.startsWith("kl")||word.startsWith("tr")||word.startsWith("ph")||word.startsWith("tw")||word.startsWith("pl")||word.startsWith("wh")||word.startsWith("pr")||word.startsWith("wr"));

}

}

}

**Problem #4**

**import** java.awt.\*;

**import** java.awt.event.ActionEvent;

**import** java.awt.event.ActionListener;

**import** javax.swing.\*;

@SuppressWarnings("serial")

**public** **class** Keypad **extends** JPanel{

// build private JPanel for each object

**private** JPanel keys = **new** JPanel();

**private** String Entered\_Numbers= "";

// private build keys for all methods

**private** JButton end\_key = **new** JButton("End");

**private** JButton call\_key = **new** JButton ("Call");

**private** JButton delete\_key = **new** JButton("Clear");

**private** JButton one\_key = **new** JButton("1");

**private** JButton two\_key = **new** JButton("2");

**private** JButton three\_key = **new** JButton("3");

**private** JButton four\_key = **new** JButton("4");

**private** JButton five\_key = **new** JButton("5");

**private** JButton six\_key = **new** JButton("6");

**private** JButton seven\_key = **new** JButton("7");

**private** JButton eight\_key = **new** JButton("8");

**private** JButton nine\_key = **new** JButton("9");

**private** JButton zero\_key = **new** JButton("0");

**private** JButton astr\_key = **new** JButton("\*");

**private** JButton pound\_key = **new** JButton("#");

// private text field length of (20)

**private** JTextField displayTextField = **new** JTextField(20);

**public** Keypad()

{

setLayout( **new** BorderLayout());

setBackground(Color.*DARK\_GRAY*);

//add keys to border

Keys();

//add display to border

displayTextField.setBorder(BorderFactory.*createLineBorder*(Color.*DARK\_GRAY*, 15));

add(displayTextField,BorderLayout.*NORTH*);

add(keys,BorderLayout.*CENTER*);

}

**public** **void** Keys()

{

keys.setLayout (**new** GridLayout(5,3));

keys.setBackground (Color.*green*);

keys.setBorder(BorderFactory.*createLineBorder*(Color.*blue*, 5));

//adding keys to the gui

keys.add(delete\_key);

keys.add(call\_key);

keys.add(end\_key);

keys.add(one\_key);

keys.add(two\_key);

keys.add(three\_key);

keys.add(four\_key);

keys.add(five\_key);

keys.add(six\_key);

keys.add(seven\_key);

keys.add(eight\_key);

keys.add(nine\_key);

keys.add(astr\_key);

keys.add(zero\_key);

keys.add(pound\_key);

// listener for key presses

delete\_key.addActionListener( **new** PhoneButtonListener());

call\_key.addActionListener( **new** PhoneButtonListener());

end\_key.addActionListener( **new** PhoneButtonListener());

one\_key.addActionListener( **new** PhoneButtonListener());

two\_key.addActionListener( **new** PhoneButtonListener());

three\_key.addActionListener( **new** PhoneButtonListener());

four\_key.addActionListener( **new** PhoneButtonListener());

five\_key.addActionListener( **new** PhoneButtonListener());

six\_key.addActionListener( **new** PhoneButtonListener());

seven\_key.addActionListener( (ActionListener) **new** PhoneButtonListener());

eight\_key.addActionListener( **new** PhoneButtonListener());

nine\_key.addActionListener( **new** PhoneButtonListener());

astr\_key.addActionListener( **new** PhoneButtonListener());

zero\_key.addActionListener( **new** PhoneButtonListener());

pound\_key.addActionListener(**new** PhoneButtonListener());

}

**private** **class** PhoneButtonListener **implements** ActionListener {

@Override

**public** **void** actionPerformed(ActionEvent event) {

**if** (event.getSource()== one\_key){

Entered\_Numbers = displayTextField.getText() + "1";

displayTextField.setText(Entered\_Numbers);

}

**else** **if** (event.getSource()== two\_key){

Entered\_Numbers = displayTextField.getText() + "2";

displayTextField.setText(Entered\_Numbers);

}

**else** **if** (event.getSource()== three\_key){

Entered\_Numbers = displayTextField.getText() + "3";

displayTextField.setText(Entered\_Numbers);

}

**else** **if** (event.getSource()== four\_key){

Entered\_Numbers = displayTextField.getText() + "4";

displayTextField.setText(Entered\_Numbers);

}

**else** **if** (event.getSource()== five\_key){

Entered\_Numbers = displayTextField.getText() + "5";

displayTextField.setText(Entered\_Numbers);

}

**else** **if** (event.getSource()== six\_key){

Entered\_Numbers = displayTextField.getText() + "6";

displayTextField.setText(Entered\_Numbers);

}

**else** **if** (event.getSource()== seven\_key){

Entered\_Numbers = displayTextField.getText() + "7";

displayTextField.setText(Entered\_Numbers);

}

**else** **if** (event.getSource()== eight\_key){

Entered\_Numbers = displayTextField.getText() + "8";

displayTextField.setText(Entered\_Numbers);

}

**else** **if** (event.getSource()== nine\_key){

Entered\_Numbers = displayTextField.getText() + "9";

displayTextField.setText(Entered\_Numbers);

}

**else** **if** (event.getSource()== zero\_key){

Entered\_Numbers = displayTextField.getText() + "0";

displayTextField.setText(Entered\_Numbers);

}

**else** **if** (event.getSource()== astr\_key){

Entered\_Numbers = displayTextField.getText() + "\*";

displayTextField.setText(Entered\_Numbers);

}

**else** **if** (event.getSource()== pound\_key){

Entered\_Numbers = displayTextField.getText() + "#";

displayTextField.setText(Entered\_Numbers);

}

**else** **if** (event.getSource() ==delete\_key){

Entered\_Numbers="";

displayTextField.setText(Entered\_Numbers);

}

**else** **if** (event.getSource() == call\_key){

String Test\_Call = displayTextField.getText();

**int** Testing= Test\_Call.length();

**int** correct = 10 ;

**int** area\_code = 7;

**if** ( Testing == correct){

Entered\_Numbers= displayTextField.getText()+ " Calling";

displayTextField.setText(Entered\_Numbers);

}

**else** **if** (Testing == area\_code){

String Area\_Numbers = " Please add area Code";

displayTextField.setText(Area\_Numbers);

}

**else**

{

String Incorrect\_Numbers = "Inccorrect Phone #";

displayTextField.setText(Incorrect\_Numbers);

}

}

**else** **if** (event.getSource() == end\_key){

String END = "Ended Call";

displayTextField.setText(END);

}

}

}

}

**import** javax.swing.\*;

**public** **class** Phone {

**public** **static** **void** main(String[] args){

JFrame frame = **new** JFrame("Phone Keypad");

frame.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);

frame.getContentPane().add(**new** Keypad());

frame.pack();

frame.setVisible(**true**);

}

}