

JAVA QUIZ

223J Quiz Application

1) A variable can be referenced from its

☐ Range ☐ Space ☐ Domain ☐ Scope

2) Methods referenced with individual objects are

☐ Private ☐ Public ☐ Static ☐ NonStatic

3) Most Class Data Fields are Private

Select An Option ▼

4) Java classes are stored in a folder or

☐ Packet ☐ Package ☐ Bundle ☐ Gaggle

Submit **Score : /20**

JAVA QUIZ

223J Quiz Application

1) A variable can be referenced from its

☐ Range ☐ Space ☐ Domain ☒ Scope

2) Methods referenced with individual objects are

☐ Private ☐ Public ☐ Static ☒ NonStatic

3) Most Class Data Fields are Private

True ▼

4) Java classes are stored in a folder or

☐ Packet ☒ Package ☐ Bundle ☐ Gaggle

Submit **Score : 20/20**

JAVA QUIZ

223J Quiz Application

1) A variable can be referenced from its

☒ Range ☐ Space ☐ Domain ☐ Scope

2) Methods referenced with individual objects are

☐ Private ☒ Public ☐ Static ☐ NonStatic

3) Most Class Data Fields are Private

False

4) Java classes are stored in a folder or

☐ Packet ☐ Package ☐ Bundle ☒ Gaggle

Score : 0/20

JAVA QUIZ

223J Quiz Application

1) A variable can be referenced from its

☐ Range ☒ Space ☐ Domain ☐ Scope

2) Methods referenced with individual objects are

☐ Private ☐ Public ☐ Static ☒ NonStatic

3) Most Class Data Fields are Private

True

4) Java classes are stored in a folder or

☐ Packet ☐ Package ☒ Bundle ☐ Gaggle

Score : 10/20

```
1
2 public class eleven {
3
4   public static void main(String[] args) {
5     QuizApplication a = new QuizApplication();
6
7     a.setVisible(true);
8
9   }
10
11 }
12
```

```

1 import java.awt.*;
2 import java.awt.event.ActionEvent;
3 import java.awt.event.ActionListener;
4 import java.awt.event.ItemEvent;
5 import java.awt.event.ItemListener;
6
7 import javax.swing.*;
8
9 public class QuizApplication extends JFrame
10 implements ActionListener, ItemListener{
11     final int FRAME_WIDTH = 350;
12     final int FRAME_HEIGHT = 360;
13
14     JLabel heading;
15     JLabel questionOne;
16     JCheckBox range;
17     JCheckBox space;
18     JCheckBox domain;
19     JCheckBox scope;
20
21     JLabel questionTwo;
22     JCheckBox privAns;
23     JCheckBox pubAns;
24     JCheckBox staticAns;
25     JCheckBox nonStaticAns;
26
27     JLabel questionThree;
28     boolean qThreeBugFix = false;
29     String[] quesThreeTF = {"Select An Option", "False", "True"};
30     JComboBox quesThreeDropDown = new JComboBox(quesThreeTF);
31
32
33     JLabel questionFour;
34     JCheckBox packetAns;
35     JCheckBox packageAns;
36     JCheckBox bundleAns;
37     JCheckBox gaggleAns;
38
39     JButton submission;
40     JLabel totalScore;
41
42     int score = 0;
43
44
45 public QuizApplication() {
46     super("JAVA QUIZ");
47
48     heading = new JLabel("223J Quiz Application");
49     questionOne = new JLabel("1) A variable can be referenced from its");
50     range = new JCheckBox("Range", false);
51     space = new JCheckBox("Space", false);
52     domain = new JCheckBox("Domain", false);
53     scope = new JCheckBox("Scope", false);
54
55     questionTwo = new JLabel("2) Methods referenced with individual objects are");
56
57     privAns = new JCheckBox("Private", false);
58     pubAns = new JCheckBox("Public", false);
59     staticAns = new JCheckBox("Static", false);
60     nonStaticAns = new JCheckBox("NonStatic", false);
61
62     questionThree = new JLabel("3) Most Class Data Fields are Private");
63

```

```

64     questionFour = new JLabel("4) Java classes are stored in a folder or");
65
66     packetAns = new JCheckBox("Packet", false);
67     packageAns = new JCheckBox("Package", false);
68     bundleAns = new JCheckBox("Bundle", false);
69     gaggleAns = new JCheckBox("Gaggle", false);
70
71     ButtonGroup quesOneGroup = new ButtonGroup();
72     ButtonGroup quesTwoGroup = new ButtonGroup();
73     ButtonGroup quesFourGroup = new ButtonGroup();
74
75     submission = new JButton("Submit");
76     totalScore = new JLabel("Score : /20");
77
78
79
80     heading.setFont(new Font("Arial", Font.BOLD, 24));
81     questionOne.setFont(new Font("Arial", Font.ITALIC, 16));
82     questionTwo.setFont(new Font("Arial", Font.ITALIC, 14));
83     questionThree.setFont(new Font("Arial", Font.ITALIC, 16));
84     questionFour.setFont(new Font("Arial", Font.ITALIC, 17));
85     totalScore.setFont(new Font("Arial", Font.BOLD, 28));
86
87     setSize(FRAME_WIDTH, FRAME_HEIGHT);
88     setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
89
90     submission.addActionListener(this);
91     //question 1
92     range.addItemListener(this);
93     space.addItemListener(this);
94     domain.addItemListener(this);
95     scope.addItemListener(this);
96     //question 2
97     privAns.addItemListener(this);
98     pubAns.addItemListener(this);
99     staticAns.addItemListener(this);
100    nonStaticAns.addItemListener(this);
101    //question 3
102    quesThreeDropDown.addItemListener(this);
103    //question 4
104    packetAns.addItemListener(this);
105    packageAns.addItemListener(this);
106    bundleAns.addItemListener(this);
107    gaggleAns.addItemListener(this);
108
109    quesOneGroup.add(range);
110    quesOneGroup.add(space);
111    quesOneGroup.add(domain);
112    quesOneGroup.add(scope);
113
114    quesTwoGroup.add(privAns);
115    quesTwoGroup.add(pubAns);
116    quesTwoGroup.add(staticAns);
117    quesTwoGroup.add(nonStaticAns);
118
119    quesFourGroup.add(packetAns);
120    quesFourGroup.add(packageAns);
121    quesFourGroup.add(bundleAns);
122    quesFourGroup.add(gaggleAns);
123
124    setLayout(new FlowLayout());
125    add(heading);

```

```

126 //question 1
127 add(questionOne);
128 add(range);
129 add(space);
130 add(domain);
131 add(scope);
132
133 //question 2
134 add(questionTwo);
135 add(privAns);
136 add(pubAns);
137 add(staticAns);
138 add(nonStaticAns);
139
140 //question 3
141 add(questionThree);
142 add(quesThreeDropDown);
143
144 //question 4
145 add(questionFour);
146 add(packetAns);
147 add(packageAns);
148 add(bundleAns);
149 add(gaggleAns);
150
151 add(submission);
152 add(totalScore);
153
154 }
155 @Override
156 public void itemStateChanged(ItemEvent e) {
157     Object source = e.getItem();
158     int select = e.getStateChange();
159
160     //Question 1
161     if(source == range) {
162         if(select == ItemEvent.SELECTED) {
163             score += 0;
164         }
165         else {
166             score -= 0;
167         }
168     }
169     else if(source == space) {
170         if(select == ItemEvent.SELECTED) {
171             score += 0;
172         }
173         else {
174             score -= 0;
175         }
176     }
177     else if(source == domain) {
178         if(select == ItemEvent.SELECTED) {
179             score += 0;
180         }
181         else {
182             score -= 0;
183         }
184     }
185     else if(source == scope) {
186         if(select == ItemEvent.SELECTED) {
187             score += 5;
188         }
189     }

```

```

189         else {
190             score -= 5;
191         }
192     }
193
194     //Question 2
195     else if(source == privAns) {
196         if(select == ItemEvent.SELECTED) {
197             score += 0;
198         }
199         else {
200             score -= 0;
201         }
202     }
203     else if(source == pubAns) {
204         if(select == ItemEvent.SELECTED) {
205             score += 0;
206         }
207         else {
208             score -= 0;
209         }
210     }
211     else if(source == staticAns) {
212         if(select == ItemEvent.SELECTED) {
213             score += 0;
214         }
215         else {
216             score -= 0;
217         }
218     }
219     else if(source == nonStaticAns) {
220         if(select == ItemEvent.SELECTED) {
221             score += 5;
222         }
223         else {
224             score -= 5;
225         }
226     }
227     //question 4
228     else if(source == packetAns) {
229         if(select == ItemEvent.SELECTED) {
230             score += 0;
231         }
232         else {
233             score -= 0;
234         }
235     }
236     else if(source == packageAns) {
237         if(select == ItemEvent.SELECTED) {
238             score += 5;
239         }
240         else {
241             score -= 5;
242         }
243     }
244     else if(source == bundleAns) {
245         if(select == ItemEvent.SELECTED) {
246             score += 0;
247         }
248         else {
249             score -= 0;
250         }
251     }

```

```

252     else if(source == gaggleAns) {
253         if(select == ItemEvent.SELECTED) {
254             score += 0;
255         }
256         else {
257             score -= 0;
258         }
259     }
260
261     //question 3
262     Object source1 = e.getSource();
263     if(source1 == quesThreeDropDown) {
264         if(select == ItemEvent.SELECTED) {
265             int positionOfSelection = quesThreeDropDown.getSelectedIndex();
266             if(positionOfSelection == 0) {
267                 if(qThreeBugFix) {
268                     score -= 5;
269                     qThreeBugFix = false;
270                 }
271             }
272             else if(positionOfSelection == 1) {
273                 if(qThreeBugFix) {
274                     score -= 5;
275                     qThreeBugFix = false;
276                 }
277             }
278             else if(positionOfSelection == 2) {
279                 if(!qThreeBugFix) {
280                     score += 5;
281                     qThreeBugFix = true;
282                 }
283             }
284         }
285     }
286 }
287
288 @Override
289 public void actionPerformed(ActionEvent e) {
290     totalScore.setText("Score : " + score + "/20");
291 }
292
293
294 }

```