

ZAVRŠNI ISPIT IZ "PROGRAMIRANJA U JAVI" GRUPA A - STR. 1/5  
18.01.2013.

STUDENT: \_\_\_\_\_

1. Koja od navedenih klasa u Swingu implementira gumbе koji nakon pritiska ostanu u tom stanju? (1 BOD)

A	JButton	C	JToggleButton	D	JSimpleButton
B	JPressedButton	D	JOnClickButton	E	JComplexButton

2. Po čemu se razlikuju objekti tipa "Statement" i "PreparedStatement"? (1 BOD)

A	Objekti klase "Statement" predstavljaju upit koji se mora prevoditi kod svakog izvođenja, a objekti klase "PreparedStatement" predstavljaju upit koji se prevodi samo jednom.
B	Objekti klase "PreparedStatement" predstavljaju upit koji se mora prevoditi kod svakog izvođenja, a objekti klase "Statement" predstavljaju upit koji se prevodi samo jednom.
C	Objekti klase "Statement" predstavljaju upit koji ne sadrži "WHERE" dio upita, dok objekti klase "PreparedStatement" predstavljaju upit koji sadrži "WHERE" dio upita.
D	Objekti klase "PreparedStatement" predstavljaju upit koji ne sadrži "WHERE" dio upita, dok objekti klase "Statement" predstavljaju upit koji sadrži "WHERE" dio upita.
E	Objekti klase "Statement" predstavljaju upite kojima se unaprijed mora definirati broj redaka koji upiti vraćaju, dok kod objekata klase "PreparedStatement" nije potrebno unaprijed definirati broj redaka koji upiti vraćaju.
F	Objekti klase "PreparedStatement" predstavljaju upite kojima se unaprijed mora definirati broj redaka koji upiti vraćaju, dok kod objekata klase "Statement" nije potrebno unaprijed definirati broj redaka koji upiti vraćaju.

3. U kojoj od navedenih metoda klase "Thread" je potrebno napisati programski kod koji nit mora izvršavati? (1 BOD)

A	public void start()	C	public void body();	E	public void activity();
B	public void execute()	D	public void run();	F	public void process();

4. Koje dvije metode se kod apleta mogu izvršiti (i time promijeniti stanje apleta) u slučaju kad je prozor preglednika u kojem se aplet izvodi minimiziran (sam proces apleta se nalazi u *idle* stanju)? (1 BOD)

A	start()	B	stop()	C	destroy()	D	init()	E	delay()	F	restart()
---	---------	---	--------	---	-----------	---	--------	---	---------	---	-----------

5. Koja vrsta iznimke se dogodi u slučaju dohvaćanja nepostojećeg elementa polja? (1 BOD)

A	Error	C	ArrayException	E	CheckedException
B	CollectionException	D	Exception	F	RuntimeException

6. Koju od navedenih ključnih riječi je potrebno dodati u definiciji metode kako bi se naznačilo da je tu metodu potrebno implementirati u klasama koje nasljeđuju tu klasu u kojoj se nalazi ta metoda? (1 BOD)

A	public	C	protected	E	abstract
B	private	D	extends	F	implements

7. Što je od navedenog potrebno ubaciti na mjesto označeno s "///?" kako bi se iscrtalo grafičko sučelje kao na slici? (2 BODA)

```
JFrame okvir = new JFrame("GridBagLayoutTest");
okvir.getContentPane().setLayout(new GridBagLayout());

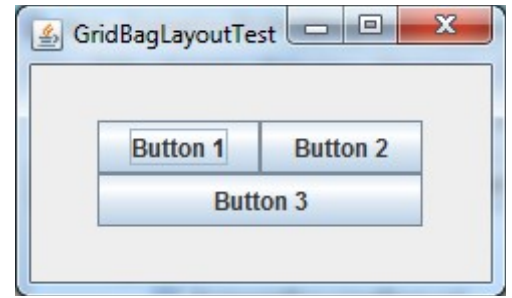
GridBagConstraints constraints = new GridBagConstraints();

constraints.fill = GridBagConstraints.HORIZONTAL;
JButton button = new JButton("Button 1");
constraints.gridx = 0;
constraints.gridy = 0;
okvir.getContentPane().add(button, constraints);

button = new JButton("Button 2");
constraints.gridx = 1;
constraints.gridy = 0;
okvir.getContentPane().add(button, constraints);

button = new JButton("Button 3");
///??
okvir.getContentPane().add(button, constraints);

okvir.setPreferredSize(new Dimension(300, 200));
okvir.pack();
okvir.setVisible(true);
```



A	constraints.x = 0; constraints.y = 1; constraints.width = 2;
B	constraints.gridx = 1; constraints.gridy = 0; constraints.width = 2;
C	constraints.x = 0; constraints.y = 1; constraints.weight = 2;
D	constraints.gridx = 0; constraints.gridy = 1; constraints.gridwidth = 2;
E	constraints.gridx = 1; constraints.gridy = 0; constraints.weight = 2;
F	constraints.gridx = 1; constraints.gridy = 0; constraints.gridwidth = 2;

8. Koji od navedenih redaka je potrebno ubaciti na mjesto označeno s "///?" kako bi se rezultat koji sadrži više redaka baze podataka prebacio u listu "listaStudenata"? (2 BODA)

```
ResultSet rs = preparedStatement.executeQuery();

List<Student> listaStudenata = new ArrayList<Student>();

///??
    int id = rs.getInt("id");
    String jmbag = rs.getString("jmbag");
    String ime = rs.getString("ime");
    String prezime = rs.getString("prezime");
    Date datumRodjenja = rs.getDate("datum_rodjenja");

    Student noviStudent = new Student();
    noviStudent.setId(id);
    noviStudent.setJmbag(jmbag);
    noviStudent.setPrezime(prezime);
    noviStudent.setIme(ime);

    noviStudent.setDatumRodjenja(datumRodjenja);
    listaStudenata.add(noviStudent);
}
```

A	do (rs.next()) {
B	if (rs.next()) {
C	while (rs.next()) {
D	do (rs.getNextRow()) {
E	if (rs.getNextRow ()) {
F	while (rs.getNextRow ()) {

9. Ako je implementirana nit uz pomoć klase "ZauzimanjeMjestaNit", što je potrebno ubaciti na mjesto označeno s "//??" kako bi se pokrenio pet niti koje oslobađaju sjedala u kino dvorani? (2 BODA)

```
ExecutorService executorServiceZauzimanje = Executors.newFixedThreadPool(5);
for (int brojacNiti = 0; brojacNiti < 5; brojacNiti++) {
    //??
}
```

A	executorServiceZauzimanje.execute(new ZauzimanjeMjestaNit(brojacNiti + 1));
B	executorServiceZauzimanje.start(new ZauzimanjeMjestaNit(brojacNiti + 1));
C	executorServiceZauzimanje.run(new ZauzimanjeMjestaNit(brojacNiti + 1));
D	executorServiceZauzimanje.process(new ZauzimanjeMjestaNit(brojacNiti + 1));
E	executorServiceZauzimanje.activity(new ZauzimanjeMjestaNit(brojacNiti + 1));
F	Ništa od navedenog.

10. Što je sve od navedenog potrebno učiniti kako bi se od sljedećeg Swing okvira dobio aplet (potrebno je zaokružiti tri od ponuđenih odgovora)? (2 BODA)

```
public class FrameAndText extends JFrame {
    Container container;
    JLabel tekst;
    public FrameAndText() {
        container = getContentPane();
        container.setLayout(new FlowLayout());
        tekst = new JLabel("Tekst unutar okvira");
        container.add(tekst);
    }
    public static void main(String[] args) {
        FrameAndText okvir = new FrameAndText();
        okvir.setTitle("Okvir s tekstom");
        okvir.setSize(300, 150);
        okvir.setVisible(true);
        okvir.setDefaultCloseOperation(
            JFrame.EXIT_ON_CLOSE);
    }
}
```

A	JPanel zamijeniti s JApplet.
B	"extends" zamijeniti s "implements".
C	Container zamijeniti s JApplet.
D	Programski kod konstruktora prebaciti u "init" metodu.
E	Naziv metode "main" zamijeniti nazivom "start".
F	Izbaciti "main" metodu.

11. Što će se ispisati sljedećim programskim odsječkom ako datoteka "brojevi.dat" sadrži sljedeće brojeve: 12 90 43 (svaki u svom retku)?(2 BODA)

```
private static final String FILENAME = "brojevi.dat";

public static void main(String[] args) {
    try (FileInputStream fis = new FileOutputStream(FILENAME)) {
        for (int n = fis.read(); n != -1;) {
            System.out.write(n);
        }
    } catch (IOException ex) {
        System.err.println("Pogreška kod čitanja datoteke " +
            FILENAME);
        ex.printStackTrace();
    }
}
```

A	12
B	90
C	43
D	12 90
E	12 90 43
F	Ništa od navedenog zbog pogreške u programu.

12. Što će se ispisati sljedećim programskim odsječkom? (2 BODA)

<pre> public class Osoba {     private float visina;     private float tezina;      public Osoba(float visina, float tezina)     {         this.visina = visina;         this.tezina = tezina;     }      public float getVisina() {         return visina;     }     public float getTezina() {         return tezina;     }     public void setVisina(float visina) {         this.visina = visina;     }     public void setTezina(float tezina) {         this.tezina = tezina;     } } </pre>	<pre> public class Usporedba implements Comparator&lt;Osoba&gt; {      @Override     public int compare(Osoba arg0, Osoba arg1) {         if (arg0.getVisina() == (arg1.getVisina())) {             return usporediTezinu(arg0, arg1);         }         else {             if (arg0.getVisina() &gt; arg1.getVisina())                 return -1;             else if (arg0.getVisina() &lt; arg1.getVisina())                 return 1;             else                 return 0;         }     }     private int usporediTezinu(Osoba arg0, Osoba arg1) {         if (arg0.getTezina() == arg1.getTezina())             return 0;         else if (arg0.getTezina() &lt; arg1.getTezina())             return -1;         else             return 1;     } } </pre>												
<table border="1"> <tr> <td>A</td><td>1.9 90.0 1.8 85.0</td></tr> <tr> <td>B</td><td>1.8 85.0 1.9 88.0</td></tr> <tr> <td>C</td><td>1.8 85.0 1.9 88.0 1.9 90.0</td></tr> <tr> <td>D</td><td>1.9 88.0 1.8 85.0 1.9 90.0</td></tr> <tr> <td>E</td><td>1.9 88.0 1.9 90.0 1.8 85.0</td></tr> <tr> <td>F</td><td>Ništa od navedenog zbog pogreške u programskom kodu.</td></tr> </table>	A	1.9 90.0 1.8 85.0	B	1.8 85.0 1.9 88.0	C	1.8 85.0 1.9 88.0 1.9 90.0	D	1.9 88.0 1.8 85.0 1.9 90.0	E	1.9 88.0 1.9 90.0 1.8 85.0	F	Ništa od navedenog zbog pogreške u programskom kodu.	<pre> public class Test {     public static void main(String[] args) {         Osoba prva = new Osoba(1.9f, 90f);         Osoba druga = new Osoba(1.8f, 85f);         Osoba treca = new Osoba(1.9f, 88f);          List&lt;Osoba&gt; nesortiranaLista = new ArrayList&lt;&gt;();         nesortiranaLista.add(prva);         nesortiranaLista.add(druga);         nesortiranaLista.add(treca);         Collections.sort(nesortiranaLista, new Usporedba());          for (Osoba os : nesortiranaLista) {             System.out.println(os.getVisina() + " " +                                 os.getTezina());         }     } } </pre>
A	1.9 90.0 1.8 85.0												
B	1.8 85.0 1.9 88.0												
C	1.8 85.0 1.9 88.0 1.9 90.0												
D	1.9 88.0 1.8 85.0 1.9 90.0												
E	1.9 88.0 1.9 90.0 1.8 85.0												
F	Ništa od navedenog zbog pogreške u programskom kodu.												

"main"? (2 BODA)

<pre>public class OzbiljnaPogreska     extends Exception {     public OzbiljnaPogreska(String poruka)     {         super(poruka);     } }</pre>	<pre>public class Glavna {     public static void main(String[] args) {         try {             test();             test2();         }         catch(VaznaPogreska   OzbiljnaPogreska ex) {             System.out.println("Pogreška: " +                                ex.getMessage());         }     }      private static void test() throws VaznaPogreska {         throw new VaznaPogreska("Test");     }      private static void test2() throws                                    OzbiljnaPogreska {         throw new OzbiljnaPogreska("Test2");     } }</pre>
--	---

A	Pogreška: Test2 Pogreška Test
B	Pogreška: Test Pogreška Test2
C	Pogreška: Test2
D	Pogreška: Test
E	Ništa od navedenog, jer programski kod nije moguće pokrenuti zbog pogreške kod prevođenja.
F	Program je moguće pokrenuti, ali se u konzoli neće ništa ispisati.