

Examcollection

<http://www.ipass4sure.com/examcollection.htm>

BCP-811

BlackBerry

Developing Java applications for the BlackBerry Platform

<http://www.ipass4sure.com/exams.asp?examcode=BCP-811>

The BCP-811 practice exam is written and formatted by Certified Senior IT Professionals working in today's prospering companies and data centers all over the world! The BCP-811 Practice Test covers all the exam topics and objectives and will prepare you for success quickly and efficiently. The BCP-811 exam is very challenging, but with our BCP-811 questions and answers practice exam, you can feel confident in obtaining your success on the BCP-811 exam on your FIRST TRY!

BlackBerry BCP-811 Exam Features

- Detailed questions and answers for BCP-811 exam
- Try a demo before buying any BlackBerry exam
- BCP-811 questions and answers, updated regularly
- Verified BCP-811 answers by Experts and bear almost 100% accuracy
- BCP-811 tested and verified before publishing
- BCP-811 examcollection vce questions with exhibits
- BCP-811 same questions as real exam with multiple choice options

Acquiring BlackBerry certifications are becoming a huge task in the field of I.T. More over these exams like BCP-811 exam are now continuously updating and accepting this challenge is itself a task. This BCP-811 test is an important part of BlackBerry certifications. We have the resources to prepare you for this. The BCP-811 exam is essential and core part of BlackBerry certifications and once you clear the exam you will be able to solve the real life problems yourself. Want to take advantage of the Real BCP-811 Test and save time and money while developing your skills to pass your BlackBerry BCP-811 Exam? Let us help you climb that ladder of success and pass your BCP-811 now!

DEMO EXAM

For Full Version visit

<http://www.ipass4sure.com/allexams.asp>

QUESTION: 1

Which block of code will ensure that a network connection has been closed?
(Choose one.)

A.

```

HttpURLConnection conn = null;
OutputStream out = null;
InputStream in = null;
String URL = "http://www.blackberry.com";
try {
    conn = (HttpURLConnection)Connector.open(URL);
    out = conn.openOutputStream();
    in = conn.openInputStream();

} catch( IOException e ) {

} finally {
    if (in != null) {
        try {
            in.close();
        } catch (IOException e) {
        }
    }
    if (out != null) {
        try {
            out.close();
        } catch (IOException e) {
        }
    }
    if (conn != null) {
        try {
            conn.close();
        } catch (IOException e) {
        }
    }
}
}

```

 C B.

```

HttpURLConnection conn = null;
OutputStream out = null;
InputStream in = null;
String URL = "http://www.blackberry.com";
try {
    conn = (HttpURLConnection)Connector.open(URL);
    out = conn.openOutputStream();
    in = conn.openInputStream();

} catch( IOException e ) {

}

in.close();
out.close();
conn.close();

```

C C.

```

HttpConnection conn = null;
OutputStream out = null;
InputStream in = null;
String URL = "http://www.blackberry.com";
try {
    conn = (HttpConnection)Connector.open(URL);
    out = conn.openOutputStream();
    in = conn.openInputStream();

} catch( IOException e ) {

}

if (in != null) {
    try {
        in.close();
    } catch (IOException e) {
    }
}
}

```

C D.

```

HttpConnection conn = null;
OutputStream out = null;
InputStream in = null;
String URL = "http://www.blackberry.com";
try {
    conn = (HttpConnection)Connector.open(URL);
    out = conn.openOutputStream();
    in = conn.openInputStream();
} catch( IOException e ) {

} finally {
    if (in != null) {
        in.close();
    }
    if (out != null) {
        out.close();
    }
    if (conn != null) {
        conn.close();
    }
}
}

```

C E.

```
HttpConnection conn = null;
OutputStream out = null;
InputStream in = null;
String URL = "http://www.blackberry.com";
try {
    conn = (HttpConnection)Connector.open(URL);
    out = conn.openOutputStream();
    in = conn.openInputStream();

} catch( IOException e ) {
    if (in != null) {
        try {
            in.close();
        } catch (IOException e) {
        }
    }
    if (out != null) {
        try {
            out.close();
        } catch (IOException e) {
        }
    }
    if (conn != null) {
        try {
            conn.close();
        } catch (IOException e) {
        }
    }
}

if (in != null) {
    try {
        in.close();
    } catch (IOException e) {
    }
}

if (out != null) {
    try {
        out.close();
    } catch (IOException e) {
    }
}
```

```

    }
    if (conn != null) {
        try {
            conn.close();
        } catch (IOException e) {
        }
    }
}

if (in != null) {
    try {
        in.close();
    } catch (IOException e) {
    }
}
if (out != null) {
    try {
        out.close();
    } catch (IOException e) {
    }
}
if (conn != null) {
    try {
        conn.close();
    } catch (IOException e) {
    }
}
}

```

- A. Exhibit A
- B. Exhibit B
- C. Exhibit C
- D. Exhibit D
- E. Exhibit E

Answer: A

QUESTION: 2

Consider the code below:

```

int width = Math.abs( ...)
...
int midpoint = width / 2;

```

What is a more efficient way of calculating the midpoint? (Choose one.)

- A. `int midpoint = (int) ((double) width) / 2.0;`
- B. `int midpoint = Fixed32.div(width, 2);`
- C. `int midpoint = (int) (width * 0.5f);`
- D. `int midpoint = width >> 1;`
- E. `int midpoint = width >> 2;`

Answer: D

QUESTION: 3

Which two of the following operations should be avoided on the main event thread?
(Choose two.)

- A. Pushing a screen onto the display stack
- B. Displaying a Dialog screen
- C. Performing network communication
- D. Calling the sleep method
- E. Capturing of keyboard input

Answer: C, D

QUESTION: 4

A BlackBerry device application collects information about several hundred books. The application needs to sort books by title. Which combination of data structures should be used to manage the objects? (Choose one.)

- A. `net.rim.device.api.util.SimpleSortingVector` and `net.rim.device.api.util.StringComparator`
- B. `java.util.Hashtable` and `net.rim.device.api.util.StringComparator`
- C. `net.rim.device.api.util.StringRepository` and `net.rim.device.api.util.StringPattern`
- D. `java.util.TreeSet` and `net.rim.device.api.util.StringComparator`
- E. `java.util.Stack` and `net.rim.device.api.util.StringComparator`

Answer: A

QUESTION: 5

An application receives notification that a new data object is being saved in the `RuntimeStore`. It may take up to 60 seconds to save this data. Which method should the application use to obtain the object from the `RuntimeStore`? (Choose one.)

- A. `get(long objectId)`
- B. `getInstance()`
- C. `fetch(Class objectClass, long timeout)`
- D. `waitFor(long objectId)`

E. put(long objectId, Object anObject)

Answer: D

QUESTION: 6

The following application is set to auto-run at startup:

```
public final class TestApp extends UiApplication
{
    public static void main(String[] args)
    {
        UiApplication theApp = new TestApp();
        theApp.enterEventDispatcher();
    }

    public TestApp()
    {
        MainScreen mainScreen = new MainScreen();

        MenuItem menu = new MenuItem("Do Something",
            1, 1)
        {
            public void run()
            {
                MyThread myTh = new MyThread();
                myTh.start();
            }
        };
        mainScreen.addMenuItem(menu);
        pushScreen(mainScreen);
    }
}
```

Which approach would most reliably solve the problem this application will encounter? (Choose one.)

- A. It should call Thread.sleep() for ten seconds before pushing the screen to verify that the BlackBerry device has completed its boot up process
- B. Threads should be moved to their own standalone class so that they do not cause a security exception
- C. Theint values in the MenuItem constructor should equal the y MenuString length to leave enough room for the string
- D. It should use theApplicationManager.inStartup() before pushing the screen to verify that the BlackBerry device has completed its boot up process
- E. It should use theApplicationManager.inStartup() before pushing the screen to verify that the BlackBerry device has completed its boot up process

- F. It should use the `ApplicationManager.inStartup()` before pushing the screen to verify that the BlackBerry device has completed its boot up process
- G. The `makeMenu` method of `MainScreen` must be overridden to use a menu

Answer: D, E, F

QUESTION: 7

A developer is designing a BlackBerry device application that provides summaries of large amounts of data. The summaries require extensive computations. Which design approach will meet these requirements? (Choose one.)

- A. Store the full set of data locally on removable media using a compressed format
- B. Supplement memory available on the BlackBerry device and compute the results locally
- C. Use URL encodings to expedite the transfer of data between a remote server and the BlackBerry device
- D. Compute the summary on a remote server and access the results through a BlackBerry Enterprise Server connection
- E. Use local storage and the floating point processor on the BlackBerry device to speed up computations

Answer: B

QUESTION: 8

A customer needs an application that will store temporary data and expose it to other BlackBerry device applications. Which API will meet this requirement? (Choose one.)

- A. `RuntimeStore`
- B. `PersistentStore`
- C. Location Based Services
- D. Global Events
- E. Connector

Answer: A

QUESTION: 9

An application requires the creation of a custom field that can display animation. Which class and method combination should be used to accomplish this task? (Choose one.)

- A. Field.paint(Graphics graphics)
- B. MainScreen.paint(Graphics graphics)
- C. Field.subpaint(Graphics graphics)
- D. Screen.draw(Graphics graphics)
- E. GameCanvas.paint(Graphics graphics)

Answer: A

QUESTION: 10

Consider the Counter class whose code is below:

```

1 public class Counter implements Runnable {
2     private static int _count;
3
4     public void run() {
5         for (int i=0; i<1000000; i++) {
6             _count++;
7         }
8     }
9
10    public static void go() throws InterruptedException {
11        _count = Integer.MAX_VALUE - 1000000;
12        Thread t1 = new Thread(new Counter());
13        Thread t2 = new Thread(new Counter());
14        t1.start(); t2.start();
15        t1.join(); t2.join();
16        System.out.println("FINAL COUNT: " + _count);
17    }
18 }
```

Assuming that the go() method is always invoked serially, why is a different count printed at the end almost every time that it is invoked? (Choose one.)

- A. The run() method is not declared "synchronized"
- B. The _count variable is not declared "synchronized"
- C. The ++ operator is not atomic
- D. The _count variable overflows
- E. The Counter class is not declared "synchronized"

Answer: C



Pass4sure Certification Exam Features;

- Pass4sure offers over **2500** Certification exams for professionals.
- More than **98,800** Satisfied Customers Worldwide.
- Average **99.8%** Success Rate.
- Over **120** Global Certification Vendors Covered.
- Services of **Professional & Certified Experts** available via support.
- Free **90 days** updates to match real exam scenarios.
- **Instant Download Access!** No Setup required.
- Price as low as **\$19**, which is 80% more **cost effective** than others.
- **Verified answers** researched by industry experts.
- Study Material **updated** on regular basis.
- Questions / Answers are downloadable in **PDF** format.
- Mobile Device Supported (**Android, iPhone, iPod, iPad**)
- **No authorization** code required to open exam.
- **Portable** anywhere.
- **Guaranteed Success.**
- **Fast**, helpful support **24x7**.



View list of All certification exams offered;
<http://www.ipass4sure.com/all exams.asp>

View list of All Study Guides (SG);
<http://www.ipass4sure.com/study-guides.asp>

View list of All Audio Exams (AE);
<http://www.ipass4sure.com/audio-exams.asp>

Download Any Certification Exam DEMO.
<http://www.ipass4sure.com/samples.asp>

To purchase Full version of exam click below;
<http://www.ipass4sure.com/all exams.asp>

3COM	CompTIA	Filemaker	IBM	LPI	OMG	Sun
ADOBE	ComputerAssociates	Fortinet	IISFA	McAfee	Oracle	Sybase
APC	CWNP	Foundry	Intel	McData	PMI	Symantec
Apple	DELL	Fujitsu	ISACA	Microsoft	Polycom	TeraData
BEA	ECCouncil	GuidanceSoftware	ISC2	Mile2	RedHat	TIA
BICSI	EMC	HDI	ISEB	NetworkAppliance	Sair	Tibco
CheckPoint	Enterasys	Hitachi	ISM	Network-General	SASInstitute	TruSecure
Cisco	ExamExpress	HP	Juniper	Nokia	SCP	Veritas
Citrix	Exin	Huawei	Legato	Nortel	See-Beyond	Vmware
CIW	ExtremeNetworks	Hyperion	Lotus	Novell	Google	

and many others.. See complete list [Here](#)

