



Content

Android Lesson 2

Review Test Submission: [Raw] Lesson 2 Quiz B

Review Test Submission: [Raw] Lesson 2 Quiz B

User	Heng Jing Han .
Course	1930 ISTD - 50.001 : Introduction to Information Systems & Programming
Test	[Raw] Lesson 2 Quiz B
Started	11/15/19 4:37 PM
Submitted	11/15/19 4:38 PM
Status	Completed
Attempt Score	12 out of 12 points
Time Elapsed	1 minute
Results Displayed	Submitted Answers, Incorrectly Answered Questions

Question 1

1 out of 1 points



Consider the following code. When main is executed, suppose `zz = 4`, what will be printed out at the console?

```
3 ▶ public class TestExceptions {
4
5 ▶   public static void main (String[] args){
6
7     try{
8       int zz = ?;
9       f( zz );
10      System.out.println("A");
11    }catch(IllegalArgumentException ex){
12      System.out.println("B");
13    }
14
15  }
16
17  public static void f(int x) throws IllegalArgumentException{
18
19    if( x > 0 && x < 10){
20      System.out.println("C");
21    }else{
22      throw new IllegalArgumentException();
23    }
24  }
25 }
```

Selected Answer: C
A

Question 2

1 out of 1 points



Consider the following class definition of **Tea**.

```
3 public class Tea {
4
5     private boolean sugar;
6     private boolean milk;
7
8     @
9     private Tea(boolean sugar, boolean milk){
10         this.sugar = sugar;
11         this.milk = milk;
12     }
13
14     @
15     public static Tea teh(){
16         return new Tea( sugar: true, milk: true);
17     }
18
19     @
20     public static Tea tehkosong(){
21         return new Tea( sugar: false, milk: true );
22     }
23
24     @Override
25     public String toString(){
26
27         String out = "teh";
28
29         if(!milk){
30             out = out + "+o";
31         }
32         if(!sugar){
33             out = out + "+kosong";
34         }
35         return out;
36     }
37 }
```

Suppose the following code is executed in another class.

What is printed at the console?

```
Tea t = Tea.tehkosong();
System.out.println(t.toString());
```

your answer is case-sensitive.

Selected Answer: teh+kosong

Question 3

3 out of 3 points



In the code below, the class **Fate** is to be instantiated by a static inner class **FateBuilder**.

```

3      public class Fate {
4
5          private String servantClass;
6          private int initialHp;
7
8      @
9      private Fate(FateBuilder fateBuilder){
10         this.servantClass = fateBuilder.servantClass;
11         this.initialHp = fateBuilder.initialHp;
12     }
13
14     static class FateBuilder {
15         private String servantClass;
16         private int initialHp;
17
18         A setServantClass(String servantClass) {
19             this.servantClass = servantClass;
20             return B ;
21         }
22
23         A setInitialHp(int initialHp) {
24             this.initialHp = initialHp;
25             return B ;
26         }
27
28         Fate createFate() {
29             return C ;
30         }
31
32     public String getServantClass() {
33         return servantClass;
34     }
35     public int getInitialHp() {
36         return initialHp;
37     }
38 }

```

In **FateBuilder**, for the setters, the return type is marked **A** (Line 17 and 22) and the return variable is marked **B** (Line 19 and 24).

The return variable/object for **createFate()** is marked **C** (Line 28).

What should **A**, **B** and **C** be?

A: **[a]** (case-sensitive)

B: **[b]** (case-sensitive)

C: **[c]** (case-sensitive)

Specified Answer for: a FateBuilder

Specified Answer for: b this

Specified Answer for: c new Fate(this)

Question 4

2 out of 2 points



Consider the following class definition of **Tea**.

```
3 public class Tea {
4
5     private boolean sugar;
6     private boolean milk;
7
8     @
9     private Tea(boolean sugar, boolean milk){
10         this.sugar = sugar;
11         this.milk = milk;
12     }
13
14     @
15     public static Tea teh(){
16         return new Tea( sugar: true, milk: true);
17     }
18
19     @
20     public static Tea tehkosong(){
21         return new Tea( sugar: false, milk: true );
22     }
23
24     @Override
25     public String toString(){
26
27         String out = "teh";
28
29         if(!milk){
30             out = out + "+o";
31         }
32         if(!sugar){
33             out = out + "+kosong";
34         }
35         return out;
36     }
37 }
```

Select all possible ways to obtain an instance of **Tea**.

Selected Answers: Tea tea = Tea.tehkosong()
Tea tea = Tea.teh();

Question 5

2 out of 2 points



Consider the following code. When main is executed, what values assigned to zz will cause only "B" to be seen at the console?

A wrong option selected will result in points deducted. However, the minimum score for this question is 0.

```

3  ▶ public class TestExceptions {
4
5  ▶     public static void main (String[] args){
6
7      try{
8          int zz = ?;
9          f( zz );
10         System.out.println("A");
11     }catch(IllegalArgumentException ex){
12         System.out.println("B");
13     }
14
15     }
16
17     public static void f(int x) throws IllegalArgumentException{
18
19         if( x > 0 && x < 10){
20             System.out.println("C");
21         }else{
22             throw new IllegalArgumentException();
23         }
24     }
25 }

```

Selected Answers: 0

-1

11

Question 6

1 out of 1 points



Consider the following code, which attempts to display the result of 8 divided by 7 using the BigDecimal class.

What is printed at the console? Your answer must not contain any trailing or leading spaces.

```

int numberOfDigits = 5;
BigDecimal a = new BigDecimal( s: "8.0");
BigDecimal b = new BigDecimal( s: "7.0");
MathContext mathContext = new MathContext(numberOfDigits, RoundingMode.HALF_UP);
System.out.println( a.divide(b,mathContext) );

```

Selected Answer: 1.1429

Question 7

2 out of 2 points



You are writing a unit test for a class in your android project.

The class, Calculation, has a static method called getDefaultValue() which returns an int of 10 when invoked. It does not take in any parameters.

Your unit test is supposed to verify that this is indeed the case.

The code for the unit test is given below. Complete the missing parts A and C.

B holds the expected result, which is 10.

The files for the class and the unit test reside in their usual respective folders.

```

A
public void someUnitTest(){
    assertEquals( B , C );
}

```

A: **[a]** (case-sensitive)

C: **[c]** (case-sensitive)

Specified Answer for: a @Test

Specified Answer for: c Calculation.getDefaultValue()

Friday, November 15, 2019 4:38:25 PM SGT

← OK