Maxwell Ugbebor

Test Correction

Original Test Score: 90/100

Question 7. Original Score: 8/10

Consider the following class:

public class Radio  
{  
private double frequency;  
private String name;  
   
  // overloaded constructor  
   
  // accessors (get methods)  
   
  // mutators (set methods)  
   
  // other methods  
}

Code the toString method. Include a DecimalFormat object to format frequency.  The String returned should be informative and professional.

Wrong answer

public class Radio  
{  
private double frequency;  
private String name;  
   
  // overloaded constructor  
   
  // accessors (get methods)  
   
  // mutators (set methods)  
   
  // other methods  
}  
@Override  
public class toString{  
DecimalFormat decimal = new DecimalFormat("0.0");  
return  
"name "+name+"frequency "+decimal.(frequency);  
}

Right answer

public class Radio  
{  
private double frequency;  
private String name;  
   
  // overloaded constructor  
   
  // accessors (get methods)  
   
  // mutators (set methods)  
   
  // other methods  
}  
@Override  
public class toString{  
DecimalFormat decimal = new DecimalFormat("0.0");  
return  
"name "+name+"frequency "+decimal.format(frequency);  
}

Explanation: I forgot to add format

Question 8. Original Score: 8/10

Consider the following class:

public class Radio  
{  
private double frequency;  
private String name;  
   
 public Radio()  
 {  
 setFrequency(0.0);  
 setName("");  
 }  
   
 public Radio(double f, String n)  
 {  
 setFrequency(f);  
 setName(n);  
 }  
  // accessors (get methods)  
   
  // mutators (set methods)  
   
  // other methods  
}

Copy the code above to your answer and add a static variable to keep track of the number of radios.  Add code that will increment the variable when a radio is instantiated by a client class.  Add a static method that will return the value of the static variable.

Wrong answer

public class Radio  
{  
private double frequency;  
private String name;  
private int numRadio = 0  
   
 public Radio()  
 {  
 setFrequency(0.0);  
 setName("");  
 numRadio++;  
 }  
   
 public Radio(double f, String n)  
 {  
 setFrequency(f);  
 setName(n);  
 numRadio++;  
 }  
  // accessors (get methods)  
   
  // mutators (set methods)  
   
  // other methods  
public static numRadio{  
return numRadio;  
}  
}

Right answer

public class Radio  
{  
private double frequency;  
private String name;  
private static int numRadio = 0  
   
 public Radio()  
 {  
 setFrequency(0.0);  
 setName("");  
 numRadio++;  
 }  
   
 public Radio(double f, String n)  
 {  
 setFrequency(f);  
 setName(n);  
 numRadio++;  
 }  
  // accessors (get methods)  
   
  // mutators (set methods)  
   
  // other methods  
public static numRadio{  
return numRadio;  
}  
}

Explanation: I wasn’t sure if I should add static or not, so I decided not to

Question 9. Original Score: 9/10

A partial listing of class Radio is below:

public class Radio  
{  
 private double frequency;  
 private String name;  
   
 public Radio()  
 {  
 setFrequency(0.0);  
 setName("");  
 }  
   
 public Radio(double f, String n)  
 {  
 setFrequency(f);  
 setName(n);  
 }  
  // accessors (get methods)  
   
  // mutators (set methods)  
   
  // other methods  
 public boolean equals(Object obj)  
 {  
 //implementation of equals  
 }  
}

In a *client* class and inside the *main* method:

1. *instantiate*an object using the default constructor.
2. *instantiate*an object using the argument constructor.
3. set the frequency of the object instantiated in #1.
4. set the name of the object instantiated in #1.
5. determine if the objects are equal.  If they are print an appropriate message

public static void main(String[] args)  
{  
 //your code starts here  
}

Wrong answer

public class Radio  
{  
 private double frequency;  
 private String name;  
   
 public Radio()  
 {  
 setFrequency(0.0);  
 setName("");  
 }  
   
 public Radio(double f, String n)  
 {  
 setFrequency(f);  
 setName(n);  
 }  
  // accessors (get methods)  
   
  // mutators (set methods)  
   
  // other methods  
 public boolean equals(Object obj)  
 {  
 //implementation of equals  
 }  
}  
public static void main(String[] args)  
{  
Radio radio1 = new Radio();  
Radio radio2 = new Radio(1000, DIO);  
radio1.frequency = 1100;  
radio1.name = Jotaro;  
if (radio1.equals(radio2))  
System.out.println("They are equal");  
}

Right answer

public class Radio  
{  
 private double frequency;  
 private String name;  
   
 public Radio()  
 {  
 setFrequency(0.0);  
 setName("");  
 }  
   
 public Radio(double f, String n)  
 {  
 setFrequency(f);  
 setName(n);  
 }  
  // accessors (get methods)  
   
  // mutators (set methods)  
   
  // other methods  
 public boolean equals(Object obj)  
 {  
 //implementation of equals  
 }  
}  
public static void main(String[] args)  
{  
Radio radio1 = new Radio();  
Radio radio2 = new Radio(1000, “DIO”);  
radio1.frequency = 1100;  
radio1.name = “Jotaro”;  
if (radio1.equals(radio2))  
System.out.println("They are equal");  
}

Explanation: Forgot to add double quotes

Question 10. Original Score: 4/5

What is the *output* of the following lines of code?

int x=7;  
while (x>1){  
  
     System.out.println(x);  
  
     x=(x%2==0)? x-2: x-3;  
  
 }

Wrong answer

7

4

2

0

Right answer

7

4

2

Explanation: I messed up

Question 12. Original Score: 4/5

Write Java statements to instantiate a random number generator object and generate a random integer between -20 and 150 inclusive.

Wrong answer

int randomInt= 0;

Random randomNum = new Random;

randomInt = Random.randomNum(171)-20;

System.out.println(randomInt);

Right answer

int randomInt= 0;

Random randomNum = new Random;

randomInt = randomNum.nextInt(171)-20;

System.out.println(randomInt);

Explanation: I got confused

Question 13. Original score: 7/10

A String variable named email contains the email of a person in this format: username@serviceProvider.extension. Examples are mike32@yahoo.com and jane21@gmail.com.For simplicity, assume that there is exactly one @ character and one . (dot) character in the String email, and that the @ character is before the . (dot) character. Output the number of characters in the email address and username.

Wrong Answer

public static void main(String[] args)

{

String email = username@serviceProvider.extension;

System.out.println(email.length());

System.out.println(0,email.charAt(@)-1);

}

Right Answer

public static void main(String[] args)

{

System.out.println(email.length());

System.out.println(0,email.charAt(‘@’)-1);

}

Explanation: I wasn’t quite sure what the question was asking