**Antworten zum Applet:**

1. Unterklassen, Subklassen, Kinderklasse

2. Der new-Operator dient zum Erzeugen eines neuen Exemplars einer Klasse. Es wird ein neuer Speicherbereich belegt, der das neue Objekt aufnimmt.

3. Die Klasse Geometrische Form erbt die Attribute und Methoden der Oberklasse JApplet.

4. Konstruktoren sind spezielle methodenähnliche Klassenstrukturen, die den Namen ihrer Klasse tragen und beim Erzeugen von Objekten der Klasse über das Schlüsselwort new aufgerufen werden.

5. Methoden haben einen Kopf und einen Rumpf. Eine Methodendeklaration besteht aus einem Methodenkopf und einem Methodenkörper. Methoden können in Java nur dann ausgeführt werden, wenn sie von einem Objekt aufgerufen werden. Diese Objekte müssen aber vorher erzeugt werden.

/\* MethodExample.java \*/

public class MethodExample {

public static void method1() {

System.out.println("Hello");

}

public void method2() {

System.out.println("World!");

}

}

6. Set- und get-Methoden dienen dazu, die Properties einer Klasse abzufragen und gegebenenfalls zu setzen. Der Namensteil einer Zugriffsmethode hinter dem Wort set oder get wird als Name der Property identifiziert.

7. Die Zugriffsklasse public erlaubt »weltweiten« Zugriff. Es kann sowohl von außen als auch von allen Unterklassen aus auf das Datenelement oder die Methoden zugegriffen werden.

Die Zugriffsklasse private ist gewissermaßen das Gegenteil von public. private-Bestandteile sind nur in der Klasse verfügbar, in der sie definiert sind. Von Objekten, die nicht zu dieser Klasse gehören, besteht kein Zugriff.

8. Ja (Java muss installiert sein!)

9. Globale Variablen sind alle Variablen, die "lose" im **Klassen**-Rumpf  
stehen, und nicht in einer Methode/Konstruktor.  
Lokale Variablen haben eine "Lebensspanne" nur innerhalb   
des Rumpfes, in dem sie deklariert wurden.

10. Ein Objekt wird mit Hilfe des new-Operators und dem Konstruktor erzeugt.

11. Mit getRadius();

12. Container dienen der Strukturierung und Gruppierung der eigentlichen Bedienelemente.

13. Ja

14. Ja

15. Ja

16.

17. Double, Float

18. Math.round(2\*VARIABLE);

19. this zeigt immer auf das aktuelle Objekt.

20. Es können Klassen, Methoden, Attribute und Parameter als final bezeichnet (deklariert) werden. Einfach ausgedrückt bedeutet final in Java "du kannst mich **jetzt** nicht überschreiben". Für finale Klassen bedeutet dies, dass man von ihr nicht erben kann (man kann keine Unterklasse erzeugen). Sie kann also nicht als Vorlage für eine neue Klasse dienen.