Welche Arten von Kommentaren sind in den folgenden Codes enthalten?

    / \*\*

      \* Hauptprogrammklasse.

      \* /

    public class Main {

          public static void main {String [] args {

                / \* druckt Antwort \* /

                System.out.println (42); // die antwort ist 42

}

}

: c1 einzelne Zeile

  : c2 mehrzeilig

  : c3 Dokumentation

  : C4-Kompilierung

: c1: c2: c3

--

Die Nummer 256 457 kann in Datentypen gespeichert werden

:c1 byte

:c2 short

:c3 int

:c4 long

:c3:c4 ok ex

--

Die Aufgabe der Java Virtual Machine ist

  : r1 Programme mit derselben Umgebung bereitstellen

  : r2 Code kompilieren

  : r3 GUI für die Programmerstellung

:r1 ok

--

Die Nummer 315 kann in Datentypen gespeichert werden

:c1 byte

:c2 short

:c3 int

:c4 long

:c2:c3:c4 ok ex

--

Die Nummer 2.6 kann in Datentypen gespeichert werden

:c1 int

:c2 long

:c3 float

:c4 double

:c3:c4 ok ex

--

Was ist der Wert von Variable cint a = 5;

int b = 6;

boolean c = a&lt;=b;

:c1 true

:c2 false

:c3 chyba

:c1 ok ex

--

Was ist der Wert von s

int x = 82;

String s = "Ihr Gewicht ist:" + x + "Kg";

  : c1 "Ihr Gewicht ist:" + x + "Kg"

  : c2 Ihr Gewicht beträgt: 42 kg

  : c3 42

  : c4 Fehler

:c2 ok ex

--

Welchen Wert wird S3 haben?

String s1 = "jdk", s2 = "7.0";

String s3 = s1 + s2;

:c1 s1 + s2

:c2 jdk7.0

:c3 jdk

:c4 7.0

:c2 ok ex

--

Wir verwenden Zeichen für einen einzeiligen Kommentar

:c1 ++

:c2 //

:c3 --

:c4 \*\*

:c2 ok ex

--

Was wird die Ausgabe sein:int m = 6;

System.out.printf( " Afrikanischer Elefant wiegt% d Tonnen ", m);

  : r1 Der afrikanische Elefant wiegt% d Tonnen

  : r2 afrikanischer Elefant wiegt 6d Tonnen

  : R3 Afrikanischer Elefant wiegt 6 Tonnen

  : r4 Fehler

: r3 ok

-

Was ist der Wert von y?

int x = 1;

int y = x++;

:r1 0

:r2 1

:r3 2

:r4 3

:r2 ok

--

Wert a wird sein:

int x=0;

int y=5;

boolean a = x == 0 &amp;&amp; y &lt;= 0;

:r1 false

:r2 true

:r3 chyba

:r1 ok

--

Wert a wird sein:

int x=0;

int y=5;

boolean a = x == 0 || y &lt;= 0;

:r1 true

:r2 false

:r3 chyba

:r1 ok

--

Wert x wird sein:

int x=24;

x /= 2;

:r1 24

:r2 2

:r3 12

:r4 chyba

:r3 ok

--

Wert y wird sein:

int x = 5;

int y = x + ++x;

:r1 9

:r2 10

:r3 11

:r4 12

:r3 ok

--

Liste:

int month = 3;

   Boolescher Wert ist My = (month == 5);

    if (isYou) {

          System.out.println ("ist Liebeszeit");

}

else {

System.out.println( " neni maj ")

}

  : r1 ist Liebeszeit

  : r2 neni maj

  : r3 wahr

  : r4 falsch

: r2 ok

--

Dieser Zyklus wird sein:

int x = 5;

while( x &gt;= 0 ) {

System.out.println( x );

x --;

}

:r1 nicht einmal

:r2 2x

:r3 3x

:r4 4x

:r5 5x

:r6 6x

:r6 ok

--

Dieser Zyklus wird sein:

int x = 5;

do {

System.out.println( x );

x --;

} while (x &lt; 0);

:r1 unendlich oft

:r2 0x

:r3 1x

:r4 2x

:r5 5x

:r1 ok

--

Wie oft wird dieser Zyklus ausgeführt:

for( int a = 1; a &lt; 4; a++ ) {

System.out.println( a );

}

:r1 0x

:r2 1x

:r3 2x

:r4 3x

:r5 4x

:r6 unendlich oft

:r4 ok

--

Wie oft wird dieser Zyklus ausgeführt:

int s = 99;

while( s &gt; 0 ) {

if( s%10 == 0 ) {

break;

}

s = s-1;

System.out.println( s );

}

:n

:n="9" ok

--

Wofür wird Vererbung verwendet?

  : c1 Die endgültigen Methoden erben die fragmentierte Klasse, können sie jedoch nicht überlappen.

  : c2 Die übergeordnete Klasse erbt alle Methoden der übergeordneten Klasse

  : c3 Die erweiterte Klasse erbt alle nicht privaten Methoden und Klassenvariablen des Vorfahren

  : c4 So erstellen Sie eine Klassenhierarchie

: c1: c3: c3 ok ex

-

Wofür werden Methoden angewendet?

  : r1 Mathematische Berechnungen

  : r2 Für statistische Berechnungen

  : r3 Dieselbe Befehlsfolge mehrmals verwenden

  : r4 Objekte erstellen

: r3 ok

--

Was kann eine Rückgabe für Methoden tun?

  : c1 Sofortige Beendigung der Methode

  : c2 Kehrt zum Anfang des Codes zurück

  : c3 Gibt den Wert zurück

  : c4 Kehrt zum Anfang der Methode zurück

: c3: c3 ok ex

-

Was ist der Wert eines Array-Elements mit Index 3?

int[] p = new int[10];

for( int i = 0; i &lt; p.length; i++ ) {

p[i] = i+2;

}

:c1 3

:c2 4

:c3 5

:c4 6

:c3 ok ex

--

Was ist der Wert eines Array-Elements mit Index 3?

int[]cisla = { 3, 5, 6, 7};

:c1 3

:c2 5

:c3 6

:c4 7

:c4 ok ex

--

Was ist der Wert eines Array-Elements mit Index

p[2][3]

int[][] p = new int[4][4];

for( int i = 0; i &lt; p.length; i++ ) {

for( int j = 0; j &lt; p.length; j++ ) {

p[i][j] = j;

}

}

:c1 1

:c2 2

:c3 3

:c4 4

:c3 ok ex

--

Was Klassen können

  : c1 Zum Beispiel Erstellung

  : c2 Objekte erstellen

  : c3 Vererbung

  : c4 Um ein Feld zu definieren

: c1: c2: c3--

Wenn ich den Protectec Access Specifier für die Klasse benutze. Welche Klassen können auf Behandlungen und Variablen zugreifen?

  : c1 jede Klasse

  : c2 Zugriff nur von dieser Klasse

  : c3 Sie können auf Klassen desselben Pakets oder von einem untergeordneten Element der Klasse aus zugreifen

  : c4 Aus einer beliebigen Klasse desselben Pakets

c3 ok ex

-

Was Konstrukteure tun

  : c1 Werte für eine Instanz der Klasse eingeben

  : c2 Erstellt eine Klasse

  : c3 löst eine Klasseninstanz aus

  : c4 Erstellt standardmäßig eine weitere Instanz einer Klasse

c1 ok ex

--

Wie viele Methoden (nicht vererbt) stehen dem Mitarbeiterobjekt zur Verfügung?

class Zamestnanec {

public Zamestnanec(int vek, int plat) {

this.vek = vek;

this.plat = plat;

}

protected int vek = 1;

public int vratVek() { return vek; }

public void nastavVek(int vek) { this.vek = vek; }

protected int plat = 1;

public int vratPlat() { return plat; }

public void nastavPlat(int plat) { this.plat = plat; }

public void predstavSe(){

System.out.println("Muj vek a plat jsou " + vek+"let "+plat+"Korun");

}

public static void main(String[] args) {

Zamestnanec zamestnanec = new Zamestnanec(30,100);

}

}

:r1 1

:r2 2

:r3 3

:r4 4

:r5 5

:r5 ok

--

What kind of comments are in the following codes?

/\*\*

\* Main program class.

\*/

public class Main {

public static void main( String[] args ) {

/\* Prints the answer \*/

System.out.println( 42 ); // answer is 42

}

}

:c1 one line

:c2 Multiline

:c3 Documentation

:c4 Compilation

:c1:c2:c3 ok ex

--

256 457 can be stored in data types

:c1 byte

:c2 short

:c3 int

:c4 long

:c3:c4 ok ex

--

Die Java Virtual Machine-Aufgabe lautet:

  : r1 Programme immer in der gleichen Umgebung bereitstellen

  : r2 Code kompilieren

  : r3 GUI zum Erstellen eines Programms

: r1 ok--

The number 315 can be stored in data types

:c1 byte

:c2 short

:c3 int

:c4 long

:c2:c3:c4 ok ex

--