

Решение тестов Intuit

тг: georg_nb

★ игра брюс 2048

Поиск по сайту

[Главная](#) / [Программирование](#) / [Язык программирования Java и среда NetBeans](#) / Тест 6

Язык программирования Java и среда NetBeans – тест 6

Всем привет! Меня зовут Брюс, я коренной житель городка Красноярск.

Мой хозяин очень старается и решает тесты, чтобы выкладывать их сюда и делиться с вами.

Будет классно, если вы скинете немного своих кровно заработанных мне на корм, если вам нравится то, что делает мой человек.

Всем удачи с тестами)

[Помочь прямо сейчас](#)

Упражнение 1:

Номер 1

Какие утверждения являются верными?

Ответ:

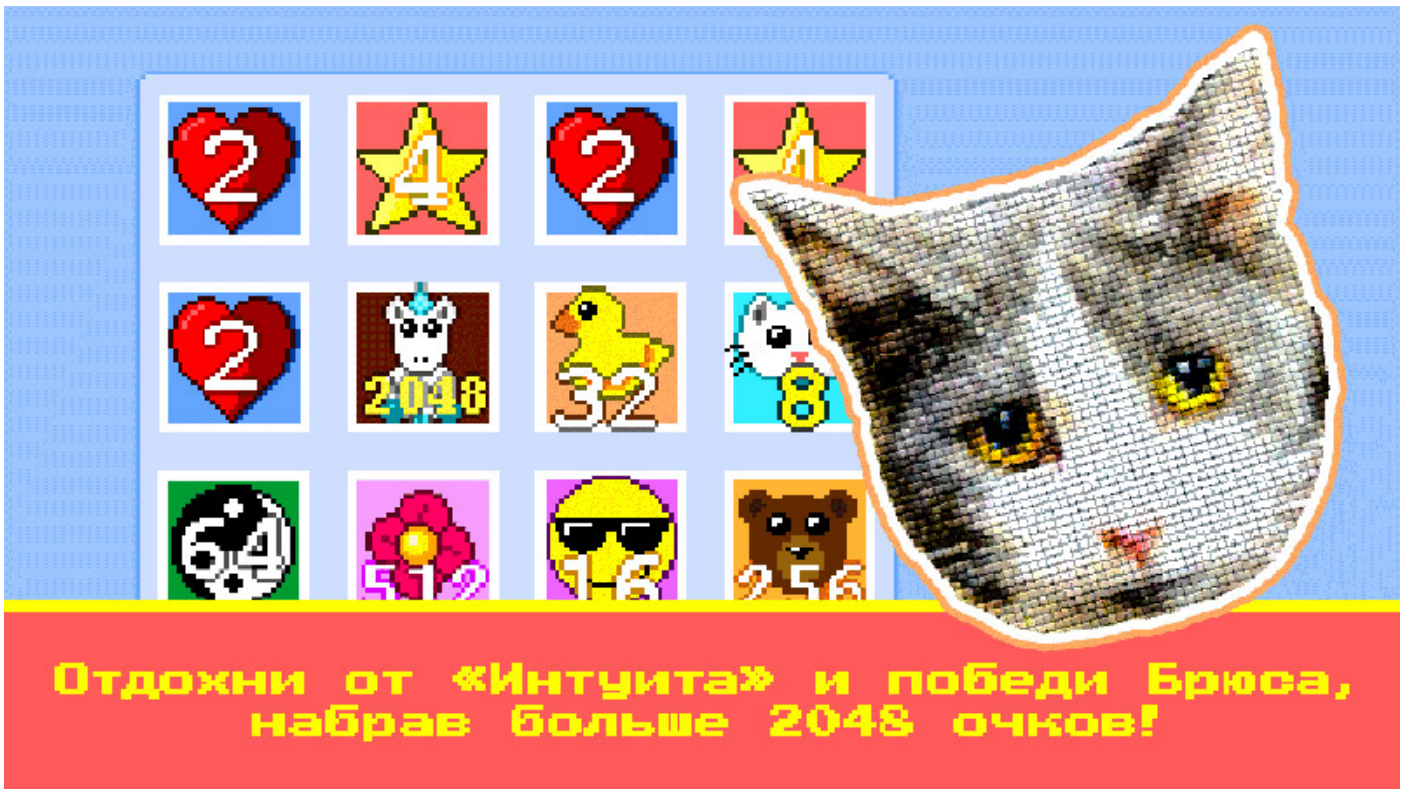
- (1) полиморфизм опирается только на наследование
- (2) наследование опирается на инкапсуляцию
- (3) инкапсуляция опирается на полиморфизм

Номер 2

Какие утверждения являются неверными?

Ответ:

- (1) полиморфизм опирается только на наследование
- (2) наследование опирается на инкапсуляцию
- (3) инкапсуляция опирается на полиморфизм



Номер 3

Какие утверждения являются верными?

Ответ:

- (1) модифицируемость кода при процедурном программировании выше, чем при объектно-ориентированном
- (2) коэффициент повторного использования программного кода при объектно-ориентированном программировании выше, чем при процедурном программировании
- (3) создание полиморфного кода возможно только в объектно-ориентированном программировании

Упражнение 2:

Номер 1

Что такое базовый класс?

Ответ:

- (1) любой класс, связанный отношением наследования, с каким-либо классом
- (2) класс, стоящий во главе иерархии, от которого прямо или опосредованно унаследованы все остальные классы
- (3) класс, стоящий во главе иерархии, от которого непосредственно унаследованы все остальные классы

Номер 2

Что такое полиморфизм?

Ответ:

- (1) возможность порождать один класс от другого с сохранением всех свойств и методов класса-предка
- (2) принцип объектно-ориентированного программирования, согласно которому пользователю класса доступна только интерфейсная часть класса, но не его внутренняя реализация
- (3) принцип объектно-ориентированного программирования, согласно которому методу класса с одним и тем же именем соответствует разный программный код в зависимости от того, объект какого класса используется при вызове данного метода

Номер 3

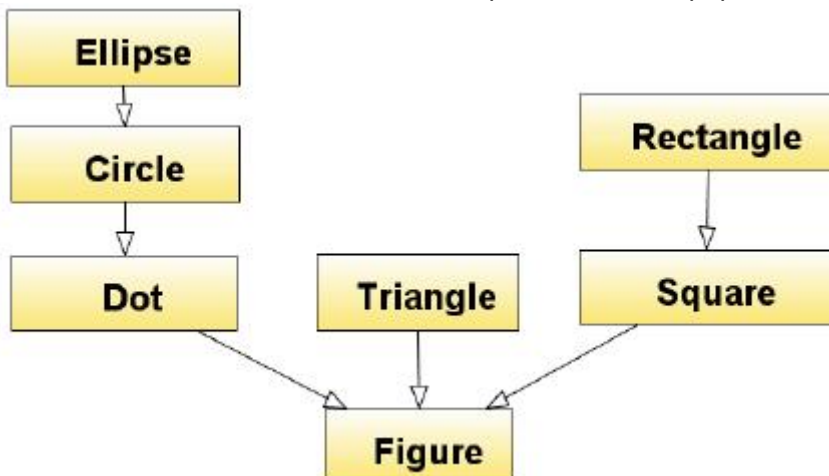
Как называется принцип объектно-ориентированного программирования, согласно которому пользователю класса доступна только интерфейсная часть класса, но не его внутренняя реализация?

Ответ:

- (1) наследование
- (2) инкапсуляция
- (3) полиморфизм

Упражнение 3:**Номер 1**

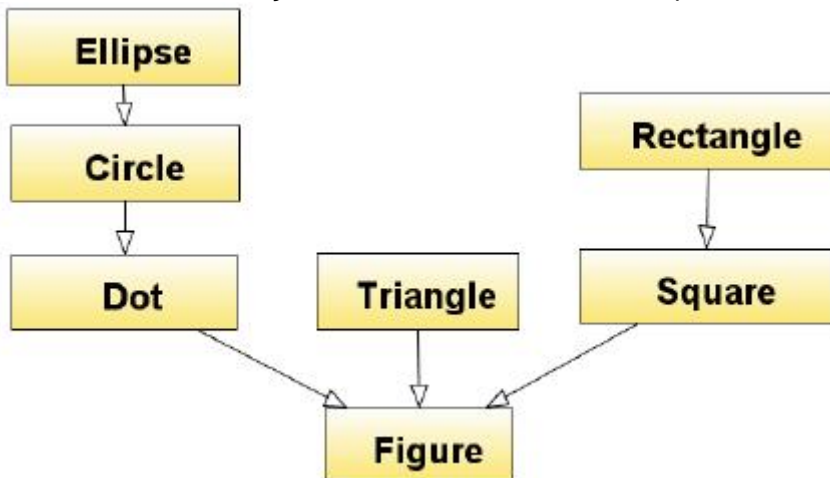
Какой класс является базовым в приведенной иерархии?

**Ответ:**

- (1) Ellipse
- (2) Dot
- (3) Triangle
- (4) Rectangle

(5) Figure**Номер 2**

Какой класс наследуется от класса Circle в приведенной иерархии?

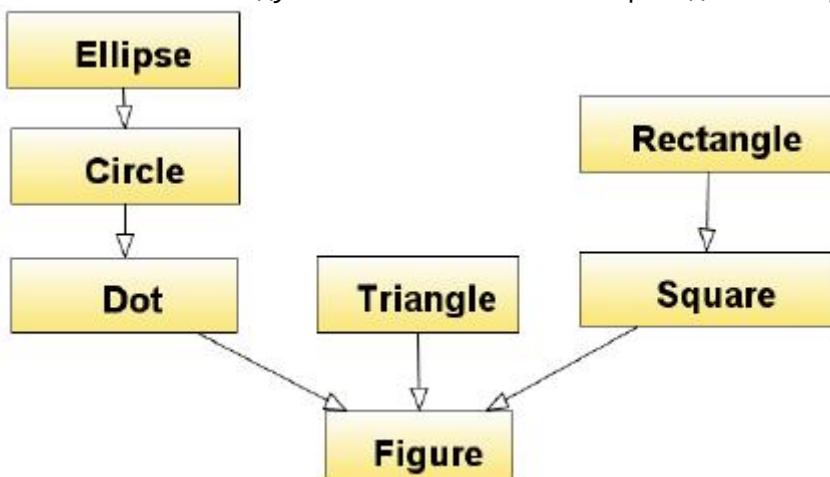


Ответ:

- (1) Ellipse
- (2) Dot
- (3) Figure

Номер 3

Какой класс наследуется от класса Dot в приведенной иерархии?

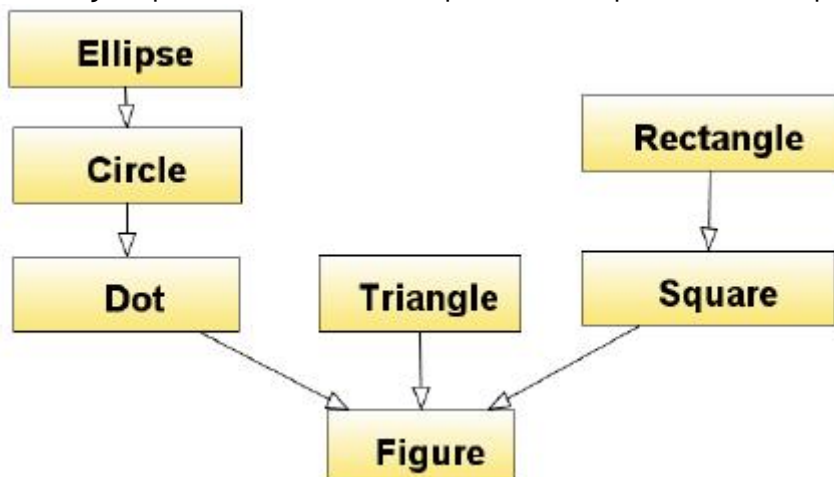


Ответ:

- (1) Ellipse
- (2) Circle
- (3) Figure

Упражнение 4:**Номер 1**

Какие утверждения являются верными для приведенной иерархии классов?

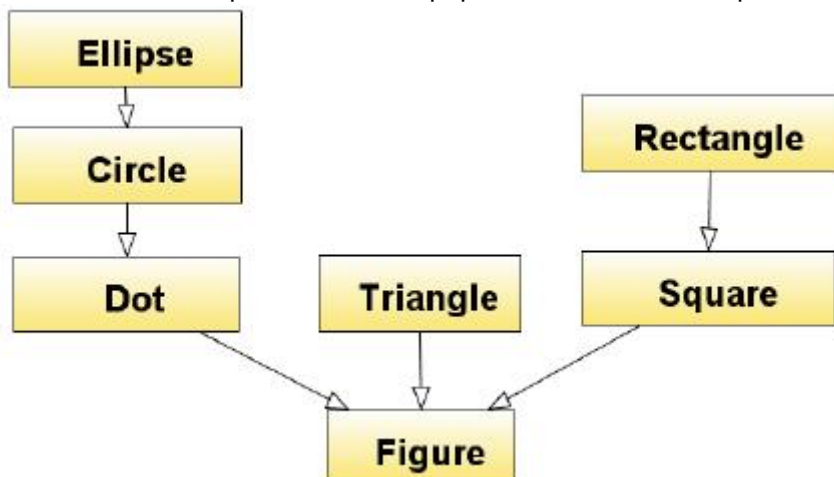


Ответ:

- (1) самым общим и универсальным классом является класс **Figure**
- (2) класс **Figure** имеет самое сложное устройство по сравнению с остальными классами
- (3) класс **Figure** обладает самым простым поведением по сравнению с остальными классами

Номер 2

Какой класс в приведенной иерархии имеет самое простое поведение?

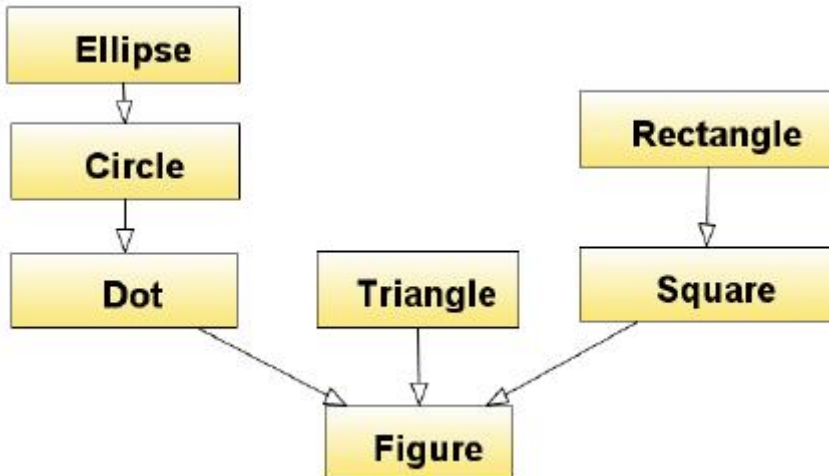


Ответ:

- (1) **Ellipse**
- (2) **Dot**
- (3) **Triangle**
- (4) **Rectangle**
- (5) **Circle**

(6) Figure**Номер 3**

Какие утверждения не являются верными для приведенной иерархии классов?



Ответ:

- (1) самым общим и универсальным классом является класс Figure
- (2) класс Figure имеет самое сложное устройство по сравнению с остальными классами
- (3) класс Figure обладает самым простым поведением по сравнению с остальными классами

Упражнение 5:**Номер 1**

Какой заголовок имеет метод:

```
public double mult1(double x, double y){  
    x++;  
    y++;  
    return x*y;  
}
```

Ответ:

- (1) mult1
- (2) mult1(double x, double y)
- (3) double mult1(double x, double y)
- (4) public double mult1(double x, double y)

Номер 2

Какой заголовок имеет метод:

```
... {  
int i;  
double r=0;  
for(i=1;i<=n;i++){  
    r+=i;  
};  
return r;  
}  
?
```

Ответ:

- (1) `public double sum1(int n)`
 - (2) `double sum1(int n)`
 - (3) `sum1(int n)`
 - (4) `sum1`
-

Номер 3

Какой заголовок имеет метод:

```
public void strAdd1(String s1,s2,s3){  
    s3=s1+s2;  
}
```

Ответ:

- (1) `strAdd1(String s1,s2,s3)`
 - (2) `void strAdd1(String s1,s2,s3)`
 - (3) `public void strAdd1(String s1,s2,s3)`
-

Упражнение 6:

Номер 1

Какой модификатор задает правила доступа к методу?

Ответ:

- (1) `public`
- (2) `static`
- (3) `final`
- (4) `private`

(5) `abstract`

(6) `protected`

Номер 2

Какой модификатор задает принадлежность к методам класса?

Ответ:

(1) `public`

(2) `static`

(3) `final`

(4) `private`

(5) `abstract`

(6) `protected`

Номер 3

Какой модификатор задает невозможность переопределения метода в потомках?

Ответ:

(1) `public`

(2) `static`

(3) `final`

(4) `private`

(5) `abstract`

(6) `protected`

Упражнение 7:

Номер 1

В классе `MyMath` описан метод `mult1`:

```
double mult1(double x, double y){  
    x++;  
    y++;  
    return x*y;  
}
```

В программе осуществляется вызов данного метода:


```
double a,b,c;  
MyMath obj1=new MyMath();  
c=obj1.mult1(a+1,b+1);
```

Какие параметры являются в таком случае фактическими?

Ответ:

- (1) x
- (2) y
- (3) a
- (4) b
- (5) a+1
- (6) b+1

Номер 2

В классе MyMath описан метод mult1:

```
double mult1(double x, double y){  
    x++;  
    y++;  
    return x*y;  
}
```

В программе осуществляется вызов данного метода:

```
double a,b,c;  
MyMath obj1=new MyMath();  
c=obj1.mult1(a+0.5,b);
```

Какие параметры являются в таком случае формальными?

Ответ:

- (1) x
- (2) y
- (3) a
- (4) b

Номер 3

В классе MyMath описан метод mult1:

```
double mult1(double x, double y){  
    x++;
```

```
        y++;  
        return x*y;  
    }
```

В программе осуществляется вызов данного метода:

```
double a,b,c;  
MyMath obj1=new MyMath();  
c=obj1.mult1(a,b);
```

Какие параметры являются в таком случае фактическими?

Ответ:

- (1) x
- (2) y
- (3) a
- (4) b
- (5) c

Упражнение 8:

Номер 1

Что такое заглушка метода?

Ответ:

- (1) заготовка метода, имеющая только имя и список параметров, но с отсутствующей реализацией
- (2) заготовка метода, имеющая имя, список параметров и неотлаженную реализацию
- (3) заготовка метода, имеющая только имя

Номер 2

В классе MyMath описан метод mult1:

```
double mult1(double x, double y){  
    x++;  
    y++;  
    return x*y;  
}
```

Какие объявления переменных a, b и c допустимы для вызова метода mult1:

```
... // объявление переменных a, b, c  
MyMath obj1=new MyMath();  
c=obj1.mult1(a+0.5,b);
```

?

Ответ:(1) `double a,b,c;`(2) `int a,b,c;`(3) `float a,b,c;`

Номер 3В классе `MyMath` описан метод `mult1`:

```
float mult1(float x, float y){  
    x++;  
    y++;  
    return x*y;  
}
```

Какие объявления переменных `a`, `b` и `c` допустимы для вызова метода `mult1`:

```
... // объявление переменных a, b, c  
MyMath obj1=new MyMath();  
c=obj1.mult1(a+0.5f,b);
```

Ответ:(1) `double a,b,c;`(2) `char a,b,c;`(3) `float a,b,c;`

Упражнение 9:**Номер 1**

В каком случае при компиляции возникнет ошибка?

Ответ:(1) `double mult1(double x, final double y){ x++; y++; return x*y; }`(2) `double mult1(double x, final double y){ x++; return x*y; }`(3) `double mult1(double x, double y){ x++; y++; return x*y; }`

Номер 2

В каком случае при компиляции не возникнет ошибок?

Ответ:

- (1) `double mult1(double x, double y){ x++; y++; return x*y; }`
 - (2) `double mult1(double x, final double y){ x++; y++; return x*y; }`
 - (3) `double mult1(double x, final double y){ x++; return x*y; }`
-

Номер 3

В каком случае при компиляции не возникнет ошибок?

Ответ:

- (1) `double mult1(double x, final double y, double z){ x++; z=x+y; return x*y*z; }`
 - (2) `double mult1(double x, final double y, final double z){ x++; z=x+y; return x*y*z; }`
 - (3) `double mult1(double x, final double y, double z){ x++; y+=x; z=x+y; return x*y*z; }`
-

Упражнение 10:

Номер 1

Какие поля класса будут доступны только в методах данного класса?

```
class Vis1 {  
    private int x=10,y=10;  
    int p1=1;  
    protected int p2=1;  
    public int p3=1;  
}
```

Ответ:

- (1) x
 - (2) y
 - (3) p2
 - (4) p3
-

Номер 2

Какие поля класса будут доступны из любых классов и объектов (с квалификацией именем пакета, если соответствующий класс не импортирован)?

```
class Vis1 {  
    private int x=10,y=10;  
    int p1=1;  
    protected int p2=1;
```

```
        public int p3=1;
    }
```

Ответ:

- (1) x
- (2) y
- (3) p2
- (4) p3

Номер 3

Какие поля класса будут доступны в методах данного класса, данного пакета, а также классах-наследниках?

```
class Vis1 {
    private int x=10,y=10;
    int p1=1;
    protected int p2=1;
    public int p3=1;
}
```

Ответ:

- (1) x
- (2) y
- (3) p2
- (4) p3

Упражнение 11:

Номер 1

Какие действия относятся к рефакторингу?

Ответ:

- (1) перемещение элементов программы с одного места на другое
- (2) переименование элементов программы (классов, переменных, методов)
- (3) добавление нового функционала
- (4) инкапсуляция полей данных

Номер 2

Какие действия не относятся к рефакторингу?

Ответ:

- (1) перемещение элементов программы с одного места на другое
 - (2) переименование элементов программы (классов, переменных, методов)
 - (3) добавление нового функционала
 - (4) инкапсуляция полей данных
-

Номер 3

Какие данные имеют наименьшее время жизни?

Ответ:

- (1) локальная переменная
 - (2) глобальная переменная
 - (3) локальный объект
 - (4) глобальный объект
-

Упражнение 12:

Номер 1

Имеется класс `Location`, для которого задан конструктор:

```
public Location (int x, int y){  
    this.x=x;  
    this.y=y;  
}
```

Какие утверждения являются верными в этом случае?

Ответ:

- (1) в классе `Location` имеется поле с именем `x`
 - (2) в классе `Location` имеется поле с именем `y`
 - (3) в классе `Location` имеется поле с именем `this.x`
 - (4) в классе `Location` имеется поле с именем `this.y`
-

Номер 2

Имеется класс Location, для которого задан конструктор:

```
public Location (int x, int y){  
    this.x=x;  
    this.y=y;  
}
```

Какие утверждения являются верными в этом случае?

Ответ:

- (1) через ссылку this доступны глобальные переменные класса Location
- (2) в классе Location заданы поля с именами this.x и this.y
- (3) имена параметров в конструкторе Location и имена глобальных переменных класса Location совпадают

Номер 3

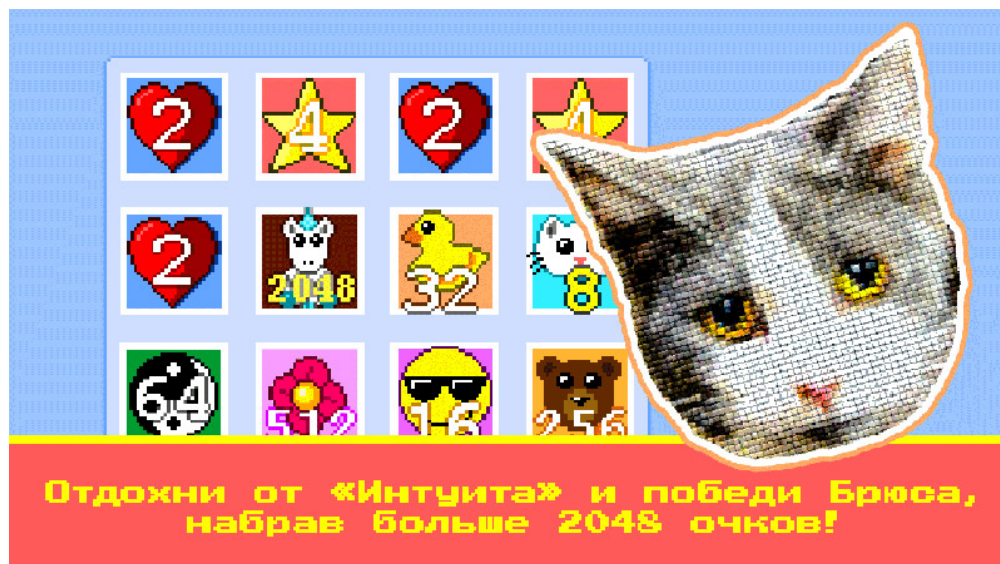
Имеется класс Location, для которого задан конструктор:

```
public Location (int x, int y){  
    this.x=x;  
    this.y=y;  
}
```

Какие утверждения являются неверными в этом случае?

Ответ:

- (1) через ссылку this доступны глобальные переменные класса Location
- (2) в классе Location заданы поля с именами this.x и this.y
- (3) имена параметров в конструкторе Location и имена глобальных переменных класса Location совпадают



[Главная](#) / [Программирование](#) / [Язык программирования Java и среда NetBeans](#) / Тест 6

Поиск правильных ответов по всему сайту

Секретная ссылка

От этого сайта «пригорает» у всех преподав
студенты закрывают сессию пиная *уи

2014 © Решение тестов Intuit