



Kuis Algorithms (OOP) - Kelas B
39 Questions

NAME : _____

CLASS : _____

DATE : _____

1. What is a Class?

- | | | | |
|----------------------------|---|----------------------------|--|
| <input type="checkbox"/> A | It is a realization of a particular item of an object | <input type="checkbox"/> B | A data field that has unique attributes and behavior |
| <input type="checkbox"/> C | A block of code which only runs when it is called. | <input type="checkbox"/> D | A blueprint that defines and describes the static attributes and dynamic behaviors common to all objects of the same kind. |

2. The best way to naming a class based on the Class Naming Convention are... (3 answers)

- | | | | |
|----------------------------|--|----------------------------|--|
| <input type="checkbox"/> A | They are denoted with a pair of parentheses | <input type="checkbox"/> B | Choose a meaningful and self-descriptive classname. |
| <input type="checkbox"/> C | It shall be a noun or a noun phrase made up of several words | <input type="checkbox"/> D | All the words shall be initial-capitalized (camel-case). |

3. What is a Constructor?

- | | | | |
|----------------------------|--|----------------------------|--|
| <input type="checkbox"/> A | It resembles an instance class in java but it's not a method as it doesn't have a return type. | <input type="checkbox"/> B | It only reveal internal mechanisms that are relevant for the use of other objects, hiding any unnecessary implementation code. |
| <input type="checkbox"/> C | A block of code that initializes the newly created object. | | |

4. A concept of having more than one constructor with different parameters list, in such a way so that each constructor performs a different task.

- | | | | |
|----------------------------|-------------------------|----------------------------|---------------------|
| <input type="checkbox"/> A | Constructor Overriding | <input type="checkbox"/> B | Constructor Listing |
| <input type="checkbox"/> C | Constructor Overloading | <input type="checkbox"/> D | Default Constructor |

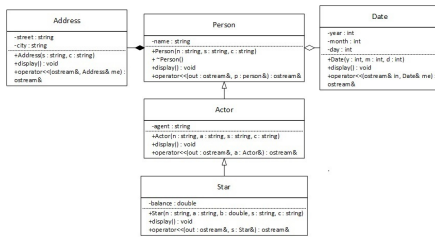
5. What are the ways of reusing existing classes in java? (2 choices)

- | | | | |
|----------------------------|-------------|----------------------------|-------------|
| <input type="checkbox"/> A | Composition | <input type="checkbox"/> B | Overloading |
| <input type="checkbox"/> C | Overriding | <input type="checkbox"/> D | Inheritance |

6. What syntax used for inheritance?

- | | |
|-----------------------|----------------------|
| <div>A</div> abstract | <div>B</div> extends |
| <div>C</div> private | <div>D</div> public |

7.



Look at the picture, which one is the Class Inheritance?

- | | |
|---|--|
| <div>A</div> Class Address to Class Person | <div>B</div> OutStream() in each class |
| <div>C</div> Class Person to Class Address and Class Date | <div>D</div> Class Actor to Class Person |

8. The mechanism of hiding of data implementation by restricting access to public methods, is the definition of?

- | | |
|---------------------------|----------------------------|
| <div>A</div> Polymorphism | <div>B</div> Abstraction |
| <div>C</div> Overloading | <div>D</div> Encapsulation |

9. _____ is wrapping of data into a single unit called class.

- | | |
|---------------------------|----------------------------|
| <div>A</div> Polymorphism | <div>B</div> Encapsulation |
| <div>C</div> Inheritance | <div>D</div> Abstraction |

10. _____ is the process in which objects of one class can link and share some common properties from the objects of another class.

- | | |
|---------------------------|----------------------------|
| <div>A</div> Inheritance | <div>B</div> Encapsulation |
| <div>C</div> Polymorphism | <div>D</div> Abstraction |

11. Sesuaikan pernyataan-pernyataan berikut ini sesuai dengan konvensi Groups: (a) Methods , (b) Class , (c) Constructors

- | | | | | | |
|----------------------------|---|-------|----------------------------|--|-------|
| <input type="checkbox"/> A | Tidak memiliki nilai pengembalian | _____ | <input type="checkbox"/> B | Kata pertama ditulis menggunakan huruf kecil, kata berikutnya diawali dengan huruf besar | _____ |
| <input type="checkbox"/> C | Hanya dapat dijalankan menggunakan operator "new" | _____ | <input type="checkbox"/> D | Nama constructor harus sama dengan nama kelas, termasuk cara penulisannya | _____ |
| <input type="checkbox"/> E | Menggunakan kata kerja | _____ | <input type="checkbox"/> F | Setiap kata harus diawali dengan huruf besar (camel case) | _____ |
| <input type="checkbox"/> G | Nama kelas harus berupa kata benda | _____ | | | |

12. Salah satu perbedaan antara pemrograman berorientasi objek dengan pemrograman terstruktur adalah....

- | | | | |
|----------------------------|--|----------------------------|--|
| <input type="checkbox"/> A | Pada pemrograman berorientasi objek algoritma diberikan penting sedangkan pemrograman terstruktur tidak | <input type="checkbox"/> B | Pemrograman terstruktur fokus pada model obyek |
| <input type="checkbox"/> C | Pada pemrograman berorientasi objek menggunakan method sedangkan pemrograman terstruktur menggunakan function | <input type="checkbox"/> D | Pada pemrograman berorientasi objek tidak ada enkapsulasi sedangkan pada pemrograman terstruktur ada enkapsulasi |
| <input type="checkbox"/> E | Pada pemrograman berorientasi objek fungsi panggilan digunakan sedangkan pemrograman terstruktur message passing digunakan | | |

13. Metode dimana suatu keadaan beberapa method mempunyai nama yang sama tetapi mempunyai fungsionalitas yang berbeda adalah

- | | | | |
|----------------------------|--------------------------|----------------------------|--------------------|
| <input type="checkbox"/> A | metode enkapsulasi | <input type="checkbox"/> B | metode overloading |
| <input type="checkbox"/> C | metode overriding method | <input type="checkbox"/> D | metode parameter |
| <input type="checkbox"/> E | metode superclass | | |

14. Di bawah ini yang termasuk dalam ciri – ciri dari metode overriding method adalah

- | | | | |
|----------------------------|--------------------------------|----------------------------|--------------------------------------|
| <input type="checkbox"/> A | Nama method berbeda | <input type="checkbox"/> B | Return type boleh sama boleh berbeda |
| <input type="checkbox"/> C | Return type harus berbeda | <input type="checkbox"/> D | Nama method harus sama |
| <input type="checkbox"/> E | Daftar parameter harus berbeda | | |

15. Fungsi new() dalam Java adalah

- | | |
|--|---|
| <div>A</div> Instansiasi obyek | <div>B</div> Keperluan pewarisan |
| <div>C</div> Keperluan pewarisan jamak | <div>D</div> Penanda program yang akan dijalankan oleh Java |

16.

```
public class mahasiswa{
    String Nim;
    String Nama;
    String Alamat;

    public void registrasi(){
        System.out.println("Nim : "+Nim);
        System.out.println("Nama : "+Nama);
        System.out.println("Alamat : "+Alamat);
    }
}
```

Diketahui terdapat class mahasiswa seperti gambar di atas. Buatlah deklarasi sebuah objek dengan nama objek mhs! (hanya menginstansiasi sebuah objek)

(a) _____ (b) _____

- | | |
|--|---|
| <div>A</div> Object mhs = Class mahasiswa(); | <div>B</div> mahasiswa mhs = new mahasiswa(); |
| <div>C</div> mahasiswa newMhs = new mahasiswa(); | <div>D</div> Object mhs = new mahasiswa(); |

17.

```
1 class Bicycle {
2     public int gear;
3     public int speed;
4
5     public Bicycle(int gear, int speed) {
6         this.gear = gear;
7         this.speed = speed;
8     }
9
10    public void applyBrake(int decrement) {
11        speed -= decrement;
12    }
13
14    public void speedUp(int increment) {
15        speed += increment;
16    }
17
18    public String toString() {
19        return("No of gears are " + gear + "\n" + "speed of bicycle is " + speed);
20    }
21 }
22
23 // derived class
24 class MountainBike extends Bicycle {
25     public int seatHeight;
26
27     public MountainBike(int gear, int speed, int startHeight) {
28         super(gear, speed);
29         seatHeight = startHeight;
30     }
31
32     public void setHeight(int newValue) {
33         seatHeight = newValue;
34     }
35
36     @Override
37     public String toString() {
38         return (super.toString()+"\nseat height is "+seatHeight);
39     }
40 }
41
42 // driver class
43 public class Test {
44     public static void main(String args[]) {
45         MountainBike mb = new MountainBike(3, 100, 25);
46         System.out.println(mb.toString());
47     }
48 }
```

Perhatikan kode pada gambar.

Ketika dijalankan, nilai yang ditampilkan untuk "speed of bicycle" adalah...

- | | |
|------------------|------------------|
| <div>A</div> 3 | <div>B</div> 100 |
| <div>C</div> 125 | <div>D</div> 25 |

18.

```

2  class Vehicle {
3      int maxSpeed = 120;
4  }
5
6  class Car extends Vehicle {
7      int maxSpeed = 180;
8      void display() {
9          System.out.println("Maximum Speed: " + super.maxSpeed);
10     }
11 }
12
13 /* Driver program to test */
14 class Test {
15     public static void main(String[] args) {
16         Car small = new Car();
17         small.display();
18     }
19 }

```

Perhatikan kode pada gambar.

Pada output program, berapa nilai maksimum speed yang ditampilkan?

- | | | | |
|-------------------------|-------------|-------------------------|-------------|
| <input type="radio"/> A | 180 | <input type="radio"/> B | 120 |
| <input type="radio"/> C | 120 dan 180 | <input type="radio"/> D | semua salah |

19. Manakah di bawah ini yang bukan konteks penggunaan keyword super dalam Java?

- | | | | |
|-------------------------|----------|-------------------------|-------------|
| <input type="radio"/> A | variabel | <input type="radio"/> B | method |
| <input type="radio"/> C | kelas | <input type="radio"/> D | konstruktor |

20. Di bawah ini keuntungan abstraksi data kecuali ...

- | | | | |
|-------------------------|-------------------------|-------------------------|---|
| <input type="radio"/> A | menurunkan reusability | <input type="radio"/> B | mengurangi kompleksitas melihat sesuatu |
| <input type="radio"/> C | mencegah duplikasi kode | <input type="radio"/> D | meningkatkan keamanan |

21. Manakah pernyataan yang salah di bawah ini?

- | | | | |
|-------------------------|--|-------------------------|---|
| <input type="radio"/> A | Subclass boleh memiliki method lebih banyak dari superclass | <input type="radio"/> B | Method di superclass bisa diatur sehingga tidak bisa di-override |
| <input type="radio"/> C | Superclass dapat menspesifikasikan method untuk subclass-nya | <input type="radio"/> D | Untuk nama method yang sama, isi method di superclass dan subclass harus sama agar tidak terjadi error saat kompilasi |

22. Perhatikan kode di bawah ini:

```

class Parent {
    final void show() {}
}
class Child extends Parent {
    void show() {}
}

```

Jika kode di atas dijalankan, apa yang terjadi?

- | | | | |
|-------------------------|----------------------------|-------------------------|----------------------------|
| <input type="radio"/> A | Terjadi error | <input type="radio"/> B | Terjadi polimorfisme |
| <input type="radio"/> C | Terjadi proses inheritance | <input type="radio"/> D | Terjadi proses enkapsulasi |

23. Dalam pemrograman berorientasi objek terdapat 2 method, tuliskan pengertian dari method..

- | | | | |
|----------------------------|--|----------------------------|---|
| <input type="checkbox"/> A | Method adalah bahasa pemrograman berorientasi objek | <input type="checkbox"/> B | Method adalah bahasa yang yang tidak memiliki sebuah objek |
| <input type="checkbox"/> C | Method Adalah Bahasa yang tidak dapat dibaca oleh computer | <input type="checkbox"/> D | Method adalah prosedur/fungsi yang dimiliki oleh sebuah objek |

24. Hak akses yang dimiliki oleh atribut adalah ...

- | | | | |
|----------------------------|-------------------------------------|----------------------------|--------------------------------|
| <input type="checkbox"/> A | portable, public, dan intermediate | <input type="checkbox"/> B | private, public, dan protected |
| <input type="checkbox"/> C | private, portable, dan protected | <input type="checkbox"/> D | private, public, dan portable |
| <input type="checkbox"/> E | intermediate, public, dan protected | | |

25. Sebutkan tiga prinsip utama dalam Pemrograman Berorientasi Obyek ...

- | | | | |
|----------------------------|--|----------------------------|-----------------------------------|
| <input type="checkbox"/> A | Public, Protected, Polymorphism | <input type="checkbox"/> B | Public, Protected, Private |
| <input type="checkbox"/> C | Encapsulation, Polymorphism, Extend | <input type="checkbox"/> D | Polymorphism , Inheritance, Class |
| <input type="checkbox"/> E | Inheritance, Polymorphism, Encapsulation | | |

26. Pernyataan yang tepat mengenai pemrograman berorientasi objek adalah...

- | | | | |
|----------------------------|--|----------------------------|--|
| <input type="checkbox"/> A | Pemograman berorientasi objek adalah pemrograman yang fokus kepada penciptaan objek-obje | <input type="checkbox"/> B | Pemograman berorientasi objek adalah pemrograman yang menutupi kelemahan procedural progamming |
| <input type="checkbox"/> C | Pemograman berorientasi objek adalah pemrograman modern yang fokus memecahkan masalah | <input type="checkbox"/> D | Pemograman berorientasi obejek adalah pemrograman yang fokus kepada fungsi-fungsi objek-objek |
| <input type="checkbox"/> E | Pemograman berorientasi objek adalah pemrograman yang di kembangkan dari procedural progamming | | |

27. Salah satu sifat Java yang memiliki arti "*banyak bentuk*" disebut?

- | | | | |
|----------------------------|-------------|----------------------------|--------------|
| <input type="checkbox"/> A | Inheritance | <input type="checkbox"/> B | Data Hiding |
| <input type="checkbox"/> C | Modifier | <input type="checkbox"/> D | Polymorphism |

28. "*public int getTinggi(){}*" adalah contoh pendeklarasian modifier di?

- | | | | |
|----------------------------|--------|----------------------------|-----------|
| <input type="checkbox"/> A | Method | <input type="checkbox"/> B | Objek |
| <input type="checkbox"/> C | Class | <input type="checkbox"/> D | Attribute |

29. Dalam konsep pemrograman berorientasi objek terdapat istilah Polymorphism, maksudnya ...

- | | |
|--|---|
| <p><input type="checkbox"/> A suatu mekanisme untuk menyembunyikan atau memproteksi suatu proses dari kemungkinan interpretasi atau penyalahgunaan dari luar sistem</p> <p><input type="checkbox"/> C digunakannya suatu interface yang sama untuk memerintah suatu objek agar melakukan suatu aksi atau tindakan yang mungkin hasil akhir yang serupa, tetapi melalui proses yang berbeda</p> | <p><input type="checkbox"/> B suatu proses dimana suatu class diturunkan dari class lainnya sehingga ia mendapatkan ciri atau sifat dari class tersebut</p> <p><input type="checkbox"/> D Benar Semua</p> |
|--|---|

30. How many objects can be made from a class?

- | | |
|---|--|
| <p><input type="checkbox"/> A one</p> <p><input type="checkbox"/> C None, you make classes from objects</p> | <p><input type="checkbox"/> B All of the above</p> <p><input type="checkbox"/> D As many as you want</p> |
|---|--|

31.

```
class kucing:
    def __init__(self, nama, perilaku):
        self.nama = nama
        self.perilaku = perilaku

    def perkenalan(self):
        print(f'Perkenalkan nama kucingku {self.nama}, dia kucing yang {self.perilaku}')

meow = kucing('Meo', 'suka berlari')
kitty = kucing(perilaku = 'suka tidur', nama = 'Kitty')
meow.perkenalan()
kitty.perkenalan()
print()
```

Pilih salah satu yang termasuk Atribut!

- | | |
|--|---|
| <p><input type="checkbox"/> A Meow, Kitty</p> <p><input type="checkbox"/> C Self</p> | <p><input type="checkbox"/> B Perkenalan</p> <p><input type="checkbox"/> D Nama, Perilaku</p> |
|--|---|

32. Kelas turunan yang ada pada *Inheritance* dapat memiliki konstruktor tetapi memiliki perilaku yang sedikit berbeda dengan konstruktor yang terdapat pada kelas induk, apa perilaku yang berbeda tersebut?

- | | |
|---|---|
| <p><input type="checkbox"/> A Apabila kelas turunan memiliki konstruktor sendiri, maka ia akan menimpa konstruktor dari kelas induk sehingga konstruktor kelas induk tidak akan pernah dieksekusi.</p> <p><input type="checkbox"/> C Apabila kelas turunan memiliki konstruktor sendiri, maka ia akan mengeksekusi konstruktor dari kelas induk terlebih dahulu, baru menjalankan konstruktor dari kelas turunan.</p> | <p><input type="checkbox"/> B Apabila kelas turunan memiliki konstruktor sendiri, maka ia akan mengeksekusi konstruktor miliknya sendiri dan kelas induk.</p> <p><input type="checkbox"/> D Apabila kelas turunan memiliki konstruktor sendiri, maka ia akan menjalankan konstruktor dari kelas induk sehingga konstruktor pada kelas turunan tidak akan pernah dieksekusi.</p> |
|---|---|

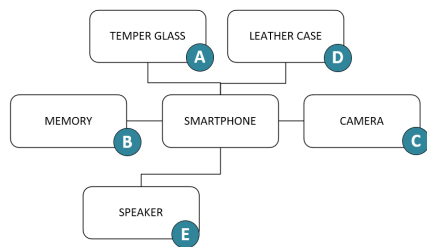
33.



Seekor Kuda jantan hitam dan gagah berlari dengan sangat cepat.
Manakah yang menunjukkan konsep Behavior?

- | | | | |
|----------------------------|------------------------|----------------------------|----------------|
| <input type="checkbox"/> A | Berlari cepat | <input type="checkbox"/> B | Kuda Jantan |
| <input type="checkbox"/> C | Seekor kuda yang gagah | <input type="checkbox"/> D | Berwarna hitam |

34.



Dari gambar disamping mana saja yang seharusnya menggunakan relasi COMPOSITION?
(3 Jawaban)

- | | | | |
|----------------------------|---|----------------------------|---|
| <input type="checkbox"/> A | A | <input type="checkbox"/> B | D |
| <input type="checkbox"/> C | B | <input type="checkbox"/> D | C |
| <input type="checkbox"/> E | E | | |

35.



Berdasarkan gambar disamping menunjukkan bahwa pelanggan bisa tidak meminjam mobil

- | | | | |
|----------------------------|-------|----------------------------|-------|
| <input type="checkbox"/> A | BENAR | <input type="checkbox"/> B | SALAH |
|----------------------------|-------|----------------------------|-------|

36. Yang termasuk bagian dari OOP adalah, kecuali..

- | | | | |
|----------------------------|---------|----------------------------|----------|
| <input type="checkbox"/> A | Sub | <input type="checkbox"/> B | Class |
| <input type="checkbox"/> C | Atribut | <input type="checkbox"/> D | Property |

37. Tindakan/aksi yang bisa dilakukan di dalam *class* disebut ...

- | | | | |
|----------------------------|----------|----------------------------|---------|
| <input type="checkbox"/> A | Behavior | <input type="checkbox"/> B | Object |
| <input type="checkbox"/> C | Property | <input type="checkbox"/> D | Atribut |

38. Fungsi atau prosedur yang dibuat oleh programmer dalam suatu kelas disebut...

- | | | | |
|----------------------------|-----------|----------------------------|--------|
| <input type="checkbox"/> A | Attribute | <input type="checkbox"/> B | Method |
| <input type="checkbox"/> C | Object | <input type="checkbox"/> D | Class |

39. _____ merupakan kelas yang mewariskan sifat dan kelakuan kepada kelas lain.

- | | | | |
|----------------------------|-------------|----------------------------|-------------|
| <input type="checkbox"/> A | Sub Class | <input type="checkbox"/> B | Inheritance |
| <input type="checkbox"/> C | Super Class | <input type="checkbox"/> D | Class |