

Given:

```
1. public class Barn {
2.     public static void main(String[] args) {
3.         new Barn().go("hi", 1);
4.         new Barn().go("hi", "world", 2);
5.     }
6.     public void go(String... y, int x) {
7.         System.out.print(y[y.length - 1] + " ");
8.     }
9. }
```

What is the result?

- A. hi hi
- B. hi world
- C. world world
- D. Compilation fails.
- E. An exception is thrown at runtime.

Answer: D

Q>

What is the result?

```
11. public class Person {
12.     String name = "No name";
13.     public Person(String nm) { name = nm; }
14. }
15.
16. public class Employee extends Person {
17.     String empID = "0000";
18.     public Employee(String id) { empID = id; }
19. }
20.
21. public class EmployeeTest {
22.     public static void main(String[] args){
23.         Employee e = new Employee("4321");
24.         System.out.println(e.empID);
25.     }
26. }
```

Choose the answer

- A. 4321
- B. 0000
- C. An exception is thrown at runtime.
- D. Compilation fails because of an error in line 18.

Question

```
1. public class Mud {
2.     //insert code here
3.     System.out.println("hi");
4. }
5. }
```

And the following five fragments:

```
public static void main(String...a) {
public static void main(String.* a) {
public static void main(String... a) {
public static void main(String[]... a) {
public static void main(String...[] a) {
```

How many of the code fragments, inserted independently at line 2, compile?

- A. 0

- B. 1
- C. 2
- D. 3
- E. 4
- F. 5

Answer: D

#### QUESTION

Given:

```

1. class Atom {
2.     Atom() {super(); System.out.print("atom "); }
3. }
4. class Rock extends Atom {
5.     Rock(String type) { super(); System.out.print(type); }
6. }
7. public class Mountain extends Rock {
8.     Mountain() {
9.         super("granite ");
10.        new Rock("granite ");
11.    }
12.    public static void main(String[] a) { new Mountain(); }
13.}

```

What is the result?

- A. Compilation fails.
- B. atom granite
- C. granite granite
- D. atom granite granite
- E. An exception is thrown at runtime.
- F. atom granite atom granite

Answer:F(constructor chaining) atom granite atom granite

Given:

```

11. abstract class Vehicle { public int speed() { return 0; }
12. class Car extends Vehicle { public int speed() { return 60; }
13. class RaceCar extends Car { public int speed() { return 150; } ...

21. RaceCar racer = new RaceCar();
22. Car car = new RaceCar();
23. Vehicle vehicle = new RaceCar();
24. System.out.println(racer.speed() + ", " + car.speed() + ", " +
vehicle.speed());

```

What is the result?

- A. 0, 0, 0
- B. 150, 60, 0
- C. Compilation fails.
- D. 150, 150, 150
- E. An exception is thrown at runtime.

Vehicle

```
|=> int speed() => 0
```

```
|
```

Car

```
|=> int speed() => 60
```

```
|
```

```

|
RaceCar (object)
|=> int speed() => 150

```

Answer: D(becoz of overriding jvm will use the runtime object)150,150,150

Question

```

5. class Building { }
6.     public class Barn extends Building {
7.         public static void main(String[] args) {
8.             Building build1 = new Building();
9.             Barn barn1 = new Barn();
10.            Barn barn2 = (Barn) build1;  Compilation successful, but Runtime Exception generated (ClassCastException)
11.            Object obj1 = (Object) build1;
12.            String str1 = (String) build1; //RE: ClassCastException
13.            Building build2 = (Building) barn1;
14.        }
15. }

```

Which is true?

- A. If line 10 is removed, the compilation succeeds.
- B. If line 11 is removed, the compilation succeeds.
- C. If line 12 is removed, the compilation succeeds.
- D. If line 13 is removed, the compilation succeeds.
- E. More than one line must be removed for compilation to succeed.

```

Object ==> obj1
|
Building====> build1
|
Barn=====> barn1

```

Answer: C

Given:

```

21. class Money {
22.     private String country = "Canada";
23.     public String getC() { return country; }
24. }
25. class Yen extends Money {
26.     public String getC() { return super.country; }
27. }
28. public class Euro extends Money {
29.     public String getC(int x) { return super.getC(); }
30.     public static void main(String[] args) {
31.         System.out.print(new Yen().getC() + " " + new
Euro().getC());
32.     }
33. }

```

What is the result?

- A. Canada
- B. null Canada
- C. Canada null
- D. Canada Canada
- E. Compilation fails due to an error on line 26.
- F. Compilation fails due to an error on line 29.

Answer: E

Q>

Given:

```
1. public class Boxer1{
2.     Integer i;
3.     int x;
4.     public Boxer1(int y) {
5.         x = i+y;
6.         System.out.println(x);
7.     }
8.     public static void main(String[] args) {
9.         new Boxer1(new Integer(4));
10.    }
11.}
```

What is the result?

- A. The value "4" is printed at the command line.
- B. Compilation fails because of an error in line 5.
- C. Compilation fails because of an error in line 9.
- D. A NullPointerException occurs at runtime.
- E. A NumberFormatException occurs at runtime.
- F. An IllegalStateException occurs at runtime.

y = 4

x = null + 4(jvm do operation on null => NullPointerException)

answer: D

Note:

Integer i1= new Integer("ten");//NumberFormatException