MCQ

1. Question: 1

Given:  
11. public interface A111 {  
12. String s = “yo”;  
13. public void method1();  
14. }  
17. interface B { }  
20. interface C extends A111, B {  
21. public void method1();  
22. public void method1(int x);  
23. } What is the result?

* 1. 

A. Compilation succeeds.

* 1. 

B. Compilation fails due to multiple errors.

* 1. 

C. Compilation fails due to an error only on line 20.

* 1. 

D. Compilation fails due to an error only on line 21.

* 1. 

E. Compilation fails due to an error only on line 22.

1. Question: 2

class Car {  
public void start() {  
System.out.println(“Car started…”);  
}

public Car m1() {

return new Car();

}

}

class BMW extends Car {  
public void start() {  
System.out.println(“BMW started…”);  
}

public BMW m1() {

return new BMW();

}

}

class Audi extends Car {  
public void start() {  
System.out.println(“Audi started…”);  
}

public Car m1() {

return new Audi();

}

}

public class TestMCQ {  
public static void main(String[] args) {  
Car c1 = new Audi();  
c1.m1().start();  
}  
}

* 1. 

a.Car started…

* 1. 

b.BMW started…

* 1. 

c.Audi started…

* 1. 

d. Compile time error

1. Question: 3

What is the result of compiling and running this program?

class Mammal{  
void eat(Mammal m){  
System.out.println(“Mammal eats food”);  
}  
}

class Cattle extends Mammal{  
void eat(Cattle c){  
System.out.println(“Cattle eats hay”);  
}  
}

class Horse extends Cattle{  
void eat(Horse h){  
System.out.println(“Horse eats hay”);  
}  
}  
public class Test{  
public static void main(String[] args){  
Mammal h = new Horse();  
Cattle c = new Horse();  
c.eat(h);  
}  
}

* 1. 

A. prints “Mammal eats food”

* 1. 

B. prints “Cattle eats hay”

* 1. 

C. prints “Horse eats hay”

* 1. 

D. Class cast Exception at runtime.

* 1. 

E. None of these

1. Question: 4

What will be the output?  
class A{  
int i = 10;  
public void printValue(){  
System.out.print(“Value-A”);  
}  
}

class B extends A{  
int i = 12;  
public void printValue(){  
System.out.print(“Value-B”);  
}  
}

public class Test{  
public static void main(String args[]){  
A a = new B();  
a.printValue();  
System.out.print(a.i);  
}  
}

* 1. 

A. Value-B 11

* 1. 

B. Value-B 10

* 1. 

C. Value-A 10

* 1. 

D. Value-A 11

* 1. 

E. None of these

1. Question: 5

method cannot be inherited to subclass.

* 1. 

A. super

* 1. 

B. static

* 1. 

C. final

* 1. 

D. private

* 1. 

E. None of these

1. Question: 6

class Sample {  
public void meth() {  
System.out.println(“Sample meth”);  
}  
}  
interface I{  
public void meth();  
}  
class Example extends Sample implements I {  
public void m2() {  
System.out.println(“m2”);  
}  
}  
public class TestMcq {  
public static void main(String[] args) {  
Example e = new Example();  
e.meth();  
e.m2();  
}  
}

* 1. 

A. Sample meth

* 1. 

B. Compile time error

* 1. 

C. Runtime Error

* 1. 

D. Sample meth  
m2

1. Question: 7

Which of these field declarations are legal within the body of an interface?  
Select the all correct answers.  
(a) public static int answer = 42;  
(b) int answer;  
(c) final static int answer = 42;  
(d) public int answer = 42;  
(e) private final static int answer = 42;

* 1. 

A. a, b, c

* 1. 

B. a, c

* 1. 

C. a, c, d

* 1. 

D. a, d, e

* 1. 

E. a, b, d

1. Question: 8

class Parent  
{  
int a = 15;  
int display()  
{  
int b = a\*a;  
return b;  
}  
}  
class Child extends Parent  
{  
float display()  
{  
float d = a/2;  
return d;  
}  
}  
public class Main  
{  
public static void main(String args[])  
{  
Child ob = new Child();  
float num = ob.display();  
System.out.println(num);  
}  
}

* 1. 

a.compile Time Error

* 1. 

b. 7.5

* 1. 

c. 0

* 1. 

d. 7.0

1. Question: 9

class A {  
void m1(String x) {  
System.out.println(“One”);  
}  
}  
class B extends A {  
public void m1(String x) {  
System.out.println(“Two”);  
super.m1(null);  
}  
}  
public class Test {  
public static void main(String[] args){  
B obj = new B();  
obj.m1(null);  
}  
}

* 1. 

a. Output is : Two, One

* 1. 

b. Output is : One, Two

* 1. 

c. Compile time error

* 1. 

d Runtime error

1. Question: 10

class Base{  
int value = 0;  
Base(){  
addValue();  
}  
void addValue(){  
value += 10;  
}  
int getValue(){  
return value;  
}  
}

class Derived extends Base{

Derived(){

addValue();

}

void addValue(){

int value=10;

super.value += 20;

}

}

public class TestMCQ{

public static void main(String[] args){

Base b = new Derived();

System.out.println(b.getValue());

}

}

* 1. 

A. 30

* 1. 

B. 10

* 1. 

C. 40

* 1. 

D. 20

* 1. 

E. Compilation error [duplicate varriable]