iCET
Connecting Intelligence

Institute of Computer Engineering Technology

iCET Certified Master

ASSIGNMENT

Assignement	WEEK 06 - Object Oriented Programming
Name	Abstraction
Ass. Date	25th February 2024

01. What are the reasons for line1, and line2 are compiled and line 3 is compile error of the following program?

```
class Customer{
    int code;
    Customer(int code){this.code=code;}
}
class Demo{
    public static void main(String args[]){
        Customer c1=new Customer(1001);
        c1.hashCode(); //Line 1
        c1.toString(); //Line 2
        c1.myMethod(); //Line 3
    }
}
```

02. What are the reasons why line5 is compiled and line 8 is compile error of the following program?

```
class Customer{
    private int code;
    private String name;
    public Customer(){
        super(); //Line 5
    }
    public Customer(int code, String name){
        super(code); //Line 8
        this.code=code;
        this.name=name;
    }
}
```

03. Complete the following program to obtain outputs for the line 8 as “Customer code-1001”

```
class Customer{
    int code;
    Customer(int code){this.code=code;}
}
class Demo{
    public static void main(String args[]){
        Customer c1=new Customer(1001);
        System.out.println(c1); //Line 8
    }
}
```

04. Insert codes to the class "Customer" to get the correct outputs for the following program.

```
class Customer{
        private int code;
        private String name;
        Customer(int code, String name){
                this.code=code;
                this.name=name;
        }
}
class Demo{
        public static void main(String args[]){
                Customer c1=new Customer(1001,"Danapala");
                Customer c2=new Customer(1001,"Danapala");
                Customer c3=new Customer(1002,"Gunapala");
                System.out.println(c1.hashCode()); //1001
                System.out.println(c2.hashCode()); //1001
                System.out.println(c3.hashCode()); //1002
        }
}
```

05. Which of the following lines are legal? Explain your answer.

```
class Super{}
class Sub extends Super{}
class Demo{
        public static void main(String args[]){
                Super sup;
                Sub sub;
                Super []supArray;
                Object ob;
                ob=new Super(); //Line 1
                ob=new Sub(); //Line 2
                ob=new Object(); //Line 3
                ob=new Object[10]; //Line 4
                sub=new Sub(); //Line 5
                sup=new Sub(); //Line 6
                sub=new Super(); //Line 7
                sup=new Super[10]; //Line 8
                sub=new Sub[10]; //Line 9
                supArray=new Sub[10]; //Line 10
                supArray=new Super[10]; //Line 11
                ob=new Super[10]; //Line 12
        }
}
```

06. Which of the following lines are legal? Explain your answer.

```
import javax.swing.*;
class A{}
class B extends A{}
class C extends B{}
class D extends B{}
class Demo{
        public static void main(String args[]){
                A[] ar={new A(),new B(),new C()}; //Line 1
                B[] br={new A(), new B(), new C()}; //Line 2
                C[] cr={new C(),new D(), new B()}; //Line 3
                Object[] ob={new A(), new D(), //Line 4
                new String(),new JFrame()};
        }
}
```

07. Complete the class “Customer” to get the output as follow.

```
class Customer{
        private int code;
        private String name;
        Customer(int code, String name){
                this.name=name;
                this.code=code;
        }
}
class Demo{
        public static void main(String args[]){
                Customer c1=new Customer(1001, "Danapala");
                Customer c2=new Customer(1002, "Gunapala");
                System.out.println(c1); //[1001-Danapala]
                System.out.println(c2); //[1002-Gunapala]
        }
}
```

08. Given:

```
class Vehicle{
      String getName() { return "Vehicle"; }
      Vehicle getType() { return this; }
}
class Car extends Vehicle{
      // insert code hereLine 6
}
class Toyota extends Car{ }
```

Which statement(s), inserted at line 6, will compile?

A. Vehicle getType() { return this; }
B. String getType() { return "this"; }
C. Car getType() { return this; }
D. Toyota getType() { return new Toyota(); }

09. If a base class has a method defined as void method() { }. Which of the following are legal prototypes in a derived class of this class? Select all correct answers.

A. void method() { }
B. int method() { return 0;}
C. void method(int i) { }
D. private void method() { }

10. Which of the following statements are true?

A. method cannot be overloaded to be less public in a child class
B. To be overridden a method must have the same name and parameter types
C. To be overridden a method must have the same name, parameter, and return types
D. An overridden method must have the same name,

11. Given:

```
 class Super{
       //Insert Code here Line 2
 }
 class Sub extends Super{
       void myMethod(){};
 }
```

Which of the following code fragments could be inserted at line 2 and still allow the code to compile?

A. static void myMethod(){}

B. final void myMethod(){}

C. private void myMethod(){}

D. private static void myMethod(){}

E. private final void myMethod(){}

F. static void myMethod(int i){}

G. public void myMethod(){}

H. protected void myMethod(){}

12. Given:

```
1. class Plant {
2.       String getName() { return "plant"; }
3.        Plant getType() { return this; }
4. }
5. class Flower extends Plant {
6.         // insert code here
7. }
8. class Tulip extends Flower { }
```

Which statement(s), inserted at line 6, will compile?

A. Flower getType() { return this; }

B. String getType() { return "this"; }

C. Plant getType() { return this; }

D. Tulip getType() { return new Tulip(); }

13. Which of the following statements are true?

A. A final method cannot be overridden.

B. All methods declared in a final class are implicitly final.

C. The methods declared in a final class must be explicitly declared final or a compile-time error occurs.

D. It is a compile-time error if a private method is declared final.

E. A machine code generator can inline the body of a final method.

F. None of the above.

14. Given:

```
1. class Programmer {
2.         Programmer debug() { return this; }
3. }
4. class SCJP extends Programmer {
5.         // insert code here
6. }
```

Which, inserted at line 5, will compile?

A. Programmer debug() { return this; }

B. SCJP debug() { return this; }

C. Object debug() { return this; }

D. int debug() { return 1; }

E. int debug(int x) { return 1; }

F. Object debug(int x) { return this; }

15. Given:

```
class G {
        String s1 = "G.s1";
        void printS1(){
                System.out.print("G.printS1," + s1);
        }
        G() { printS1();}
}
class H extends G {
        String s1 = "H.s1";
        void printS1(){
                System.out.print("H.printS1," + s1);
        }
}
class Demo{
        public static void main(String[] args) {
                H h = new H();
        }
}
```

What is the result of attempting to compile and run the program?

A. Prints: G.printS1,G.s1

B. Prints: G.printS1,H.s1

C. Prints: G.printS1,null

D. Prints: H.printS1,G.s1

E. Prints: H.printS1,H.s1

F. Prints: H.printS1,null

16. Given

```
class E{
        void m(){
                System.out.print("A"+" ");
        }
        static void m1(){
                System.out.print("B"+" ");
        }
}
class F extends E{
        void m(){
                System.out.print("AAA"+" ");
        }
        static void m1(){
                System.out.print("BBB"+" ");
        }
        public static void main(String args[]){
                E e=new F();
                e.m();
                e.m1();
        }
}
```

What is the result of attempting to compile and run the program?

A. Prints: AB
B. Prints: AAAB
C. Prints: ABBB
D. Prints: AAABBB

17. Given

```
class Super{
        static int i=10; //Line 1
        int j=20; //Line 2
        void m1(){} //Line 3
        static void m2(){} //Line 4
}
class Sub extends Super{
        int i=5; //Line 5
        static int j=10; //Line 6
        static void m1(){} //Line 7
        void m2(){} //Line 8
}
```

A. Line 1
B. Line 2
C. Line 3
D. Line 4
E. Line 5
F. Line 6
G. Line 7
H. Line 8

18. Which of the statements below are true?
    A. Abstract classes cannot be used to instantiate objects, because they're incomplete.
    B. An abstract class must contain at least one abstract method.
    C. Constructors and static methods can be declared abstract.
    D. Abstract methods do not provide implementations.
    E. Abstract classes sometimes constitute several levels of a hierarchy.
    F. Classes that can be used to instantiate objects are called concrete classes. They provide implementations of every method they declare.
    G. A class that contains any abstract methods must be declared as an abstract class. Each concrete subclass must provide implementations of each of the superclass's abstract methods.

19. Which of the following are true statements?
    A. Any named class can be declared abstract.
    B. An incompletely implemented class must be declared abstract.
    C. An abstract class can be instantiated.
    D. An abstract class is implicitly final.
    E. An abstract class must declare at least one abstract method.
    F. An abstract class cannot extend a concrete class.

20. Which of the statements below are true?
    A. A superclass reference variable can be used to invoke only methods declared in the superclass.
    B. Operator 'instanceof' determines if an object hasthe 'has a' relationship with a specific type.
    C. The 'is-a' relationship applies only between the subclass and its super classes, not vice versa.
    D. Most method calls are resolved at execution time, based on the type of the object being manipulated. This process is known as dynamic binding or late binding.

21. Which of the statements below are true?

    A. An interface specifies what operations are allowed but not how they’re performed.

    B. A Java interface describes a set of methods that can be called on an object.

    C. An interface cannot declare variables or constants.

    D. All objects of a class that implement multiple interfaces have the ‘is-a’ relationship with each implemented interface type.

22. Given Code:

```
abstract class Base{
        abstract public void myfunc();
        public void another(){
                System.out.println("Another method");
        }
}
class Abs extends Base{
        public static void main(String argv[]){
                Abs a = new Abs();
                a.amethod();
        }
        public void myfunc(){
                System.out.println("My func");
        }
        public void amethod(){
                myfunc();
        }
}
```

    What will happen when you attempt to compile and run this code?

    A. The code will compile and run, printing out the words "My Func"

    B. The compiler will complain that the Base class has non-abstract methods

    C. The code will compile but complain at run time that the Base class has non-abstract methods

    D. The compiler will complain that the method ‘myfunc’ in the base class has nobody, nobody at all to lose it

23. Which of the following methods can be legally inserted at line 2.

```
abstract class Customer{
        //Line 2
}
```

A. void search(String name){}
B. abstract void search(String name){}
C. abstract void search(String name);
D. private abstract void search(String name);
E. static abstract void search(String name);
F. private static abstract void search(String name);
G. public abstract void search(String name);

24. Compare and contrast abstract classes and interfaces. Why would you use an abstract class? Why would you use an interface?

25. Given:

```
interface A {
        void m1(); // 1
        public void m2(); // 2
        protected void m3(); // 3
        private void m4(); // 4
}
```

What is the result of attempting to compile the code?
A. Compiler error at line 1.
B. Compiler error at line 2.
C. Compiler error at line 3
D. Compiler error at line 4.

26. Which of the following statements are true?
A. All of the variables in an interface are implicitly static.
B. All of the variables in an interface are implicitly final.
C. All of the methods in an interface are implicitly abstract.
D. A method in an interface can access class-level variables.

27. Which of the following are legal declarations for non-nested classes and interfaces?
    A. final abstract class Test
    B. public static interface Test{}
    C. final public class Test{}
    D. protected abstract class Test{}
    E. protected interface Test{}
    F. abstract public class Test{}

28. Which of the following statements are true?
    A. An interface can only contain methods and not variables.
    B. Interfaces cannot have constructors.
    C. Class may extend only one other class and implement only one interface.
    D. Interfaces are the Java approach to addressing its lack of multiple inheritance, but require.
    E. Implementing classes to create the functionality of the Interfaces.