Question 1

Correct

Mark 1.00 out of 1.00

Flag question

Which of the following are valid lambda expression ?

Select one or more:

- [x] (int a, int b)->a+b; ✔
- [ ] int a, int b->a+b;
- [x] ( )->System.out.println("Welcome"); ✔
- [x] (int a)->a+2; ✔

Question 2

Correct

Mark 1.00 out of 1.00

Flag question

Predict the output of the below code

```
interface ITest{
    public float calculate(int a,int b);
}
public class Test{
    public static void main(String args[])
    {
        ITest test=(int a,int b)->{return a+b;};
        System.out.println(test.calculate(5, 4));
    }
}
```

Select one:

- ( ) Compilation error because the statements in lamba expression should not be enclosed in curly braces {}
- ( ) Compilation error because the lamba expression should not contain the data type.
- ( ) Compilation error because the lamba expression should not contain a return statement
- (•) Prints 9 ✔

Question 3

Correct

Mark 1.00 out of 1.00

Flag question

A class can be declared as _____ if it should not be sub classed.

Select one:

- ( ) public
- ( ) protected
- ( ) private
- (•) final ✔

Question 4

Correct

Mark 1.00 out of 1.00

Flag question

If a method in an interface is implemented in it, then that method should be either _____ or _____.

Select one:

- (•) static, default ✔
- ( ) abstract, static
- ( ) public , abstra
- ( ) abstract, default

Question 5

Correct

Mark 4.00 out of 4.00

Flag question

B A

```
class A {
   String name="A";
   public String getName() {
      return name;
   }
   String greeting() {
      return "class A";
   }
}
class B extends A {
   String name="B";
   String greeting() {
      return "class B";
   }
}
public class Test  {
  public static void main(String arg[]) {
     A a=new A();
     A b=new B();
     System.out.println(a.greeting()+" has name "+a.getName());
     System.out.println(b.greeting()+" has name "+b.getName());
  }
}
```

Place the names "A" and "B" in the following output.

class A ✔ has name A ✔

class B ✔ has name A ✔

Question 6

Correct

Mark 1.00 out of 1.00

Flag question

A default method in an interface can be either private or public or protected. State True or False.

Select one:

- True
- False ✔

Question 7

Correct

Mark 1.00 out of 1.00

Flag question

Given:

```
1. public class Employee {
2. String name;
3. double baseSalary;
4. Employee(String name, double baseSalary) {
5. this.name = name;
6. this.baseSalary = baseSalary;
7. }
8. }
```

And:

```
11. public class Salesperson extends Employee {
12. double commission;
13. public Salesperson(String name, double baseSalary,
14. double commission) {
15. // insert code here
16. }
17. }
```

Which code, inserted at line 17, completes the Salesperson constructor?

Select one:

- this.commission = commission;
  super();
- super();
  this.commission = commission;
- this.commission = commission;
  super(name, baseSalary);
- this.commission = commission;
- super(name, baseSalary);
  this.commission = commission; ✔
- super();
  commission = commission;

Question 8

Correct

Mark 1.00 out of 1.00

Flag question

Observe the below code

```
interface A {
    public void printNumber(int a);
    public default void display()
    {
        System.out.println("Hello");
    }
}

class Test{
    public static void main(String args[])
    {
        A a1= (num) -> System.out.println(num);
        a1.printNumber(20);
    }
}
```

What will be the output of the above code?

Select one:

- Runtime error because the interface A should contain only one abstract method
- Hello
  20
- Compilation error because the display method in interface A should be declared as static
- 20 ✔
- Compilation error because the interface A cannot have a default method

Question 9

Correct

Mark 1.00 out of 1.00

Flag question

Which would declare a compilable abstract class?

Select one:

- public abstract class Shape { public Square draw() { } } ✔
- public class Shape { public abstract Square draw(); }
- public abstract class Shape { public Square draw(); }
- public class Shape abstract { public abstract Square draw(); }

Question 10

Correct

Mark 1.00 out of 1.00

Flag question

If a class inheriting an abstract class does not provide definition for all abstract methods in the parent class, then it will be known as _____.

Select one:

- A simple class
- A concrete class
- Static class
- abstract ✔

Question 11

Correct

Mark 1.00 out of 1.00

Flag question

Predict the output:

```
public abstract class Abs {
        public Abs(){
                System.out.println("Constructor from Abstract class");
        }
}
public class Test extends Abs {
        public static void main(String args[]){
                        Abs obj=new Test();
        }
}
```

Select one:

- Constructor from Abstract class ✔
- Program will execute successfully but not display anything
- Compile time error: An abstract class cannot have a constructor
- Compile time error: An abstract class cannot be instantiated

Question 12

Correct

Mark 1.00 out of 1.00

Flag question

If a method in a super class is overridden by the sub class, then the overridden method can be invoked using _____ keyword.

Select one:

- this
- extends
- super ✔
- class

Question 13

Correct

Mark 1.00 out of 1.00

Flag question

Which of the following are valid functional interface ?

Select one or more:

- ```
  @FunctionalInterface
  interface Test
  {
      public void calculate(int a,int b,char oper);
      public default void display(){
              System.out.println("default method");
      }
  }
  ```
  ✔
- ```
  @FunctionalInterface
  interface Test
  {
      int a=10;
      public void display();
  }
  ```
  ✔
- ```
  @FunctionalInterface
  interface Test
  {
      public void display();
  }
  ```
  ✔
- ```
  @FunctionalInterface
  interface Test
  {
      public void display();

      public char getInput();
  }
  ```

Question 14

Correct

Mark 1.00 out of 1.00

Flag question

Observe the below interface.

```
interface B{
    public float display();
}
```

Which of the following are valid lambda expression for the above interface?

Select one or more:

- [x] B b1=( )->{ float a=10.5f; return a; }; ✔
- [ ] B b1=( ) -> float a=10.5f; return a;
- [ ] B b1=( )->{ return a; };
- [x] int a=15;
  B b1=( )->{ return a; } ; ✔

Question 15

Correct

Mark 1.00 out of 1.00

Flag question

```
public abstract class Shape  {
    private int x; private int y;
    public abstract void draw();
    public void setAnchor(int x, int y) {
            this.x = x;
            this.y = y;
    }
}
```

Which two classes use the Shape class correctly? (Choose two.)

Select one or more:

- [x] public class Circle extends Shape {
  private int radius;
  public void draw() {/* code here */}
  } ✔
- [x] public abstract class Circle extends Shape {
  private int radius;
  } ✔
- [ ] public class Circle extends Shape {
  private int radius;
  public void draw();
  }
- [ ] public class Circle implements Shape {
  private int radius;
  }

Question 16

Correct

Mark 1.00 out of 1.00

Flag question

_____ can be achieved through inheritance.

Select one:

- ( ) code reusability
- ( ) run time polymorphism
- (•) both run time polymorphism & code reusability ✔
- ( ) none of the options

Question 17

Correct

Mark 1.00 out of 1.00

Flag question

Which of the following represents the correct lambda expression for the Functional method :

int findMin(int a,int b) ?

Select one:

```
(int a,int b) -> {
 int min = a>b ? a : b;
 return min; }
```
✔

```
(a, b) -> (
 int min = a>b ? a : b;
 return min; )
```

```
(int a,int b) -> {
 min = a>b ? a : b;
 return min; }
```

```
(int a,int b) ->
 int min = a>b ? a : b;
 return min;
```

Question 18

Correct

Mark 1.00 out of 1.00

Flag question

Assume Book is a parent class and Magazine class is the child of Book class.

Match the following:

Book b = new Magazine(); Upcasting ✔

Magazine m = (Magazine) b; Downcasting ✔

Question 19

Correct

Mark 1.00 out of 1.00

Flag question

Predict the output of the below code :

```
interface A{
    public void display();
    public void printNumber(int a);
}
class Test{
    public static void main(String args[])
    {
        A a1={ ( ) -> System.out.println("Hai");
                (a) -> System.out.println(a);
              };
        a1.printNumber(20);
    }
}
```

Select one:

- Runtime error
- Executes but prints nothing
- Compilation error ✔
- Prints 20