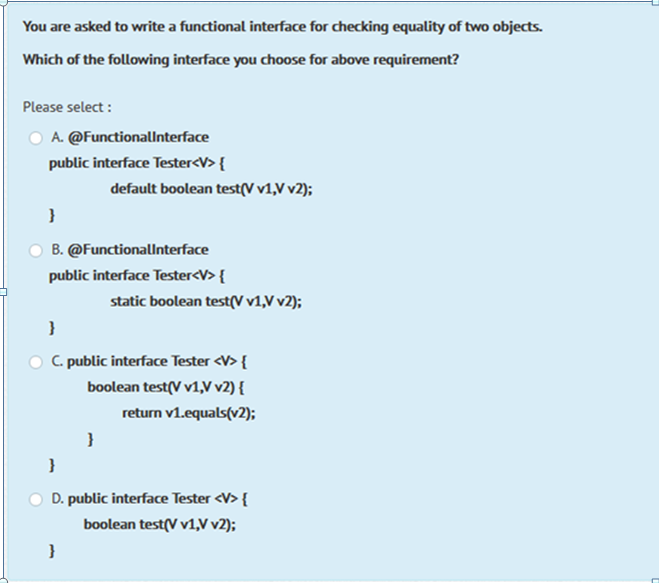
**Pregunta 1**

****

**Pregunta 2**

Given the following:

interface Runnable{

void run();

}

Which of the following will create instance of Runnable type (choose 2)?

A. Runnable run = () -> {System.out.println("Run");};

B. Runnable run =void -> System.outprintln("Run");

C. Runnable run = () -> System.out.println("Run");

D. Runnable run = -> System.out.println("Run"};

E. Runnable run = () -> {return System.out.println("Run");}

**Pregunta 3**

Que ocurrirá al ejecutar el siguiente programa?

**interface** InterA{

**void** m();

**default** **void** mr(){

System.***out***.println("mr de InterA");

}

String toString(); //linea 1

}

**interface** InterB **extends** InterA{

**default** **void** mr(){ //linea 2

System.***out***.println("mr de InterB");

}

}

**public** **class** TestLambda {

**public** **static** **void** main(String[] args) {

InterB ia=()->System.***out***.println("hola"); //linea 3

ia.m();

ia.mr();

}

}

1. Error de compilación en linea 1
2. Error de compilación en linea 2
3. Error de compilación en línea 3
4. Mostrará "Hola mr de InterA"
5. Mostrará "Hola mr de InterB"

**Pregunta 4**

Dada la siguiente instrucción:

**x->Integer.toString(x);**

¿Cuál de los siguientes métodos satisface la expresión lambda anterior?

1. public abstract Integer metodo(String s);
2. public abstract String metodo(Integer s);
3. public abstract void metodo(Integer s);
4. public abstract Integer metodo();
5. public abstract Integer metodo(Integer s);