

Coincidencia de patrones con instanceof



Operador instanceof

- Se utiliza para comprobar si un objeto es de un tipo específico:

```
Object ob=new ...  
if(ob instanceof String){  
  
}
```

- Se puede utilizar con clases e interfaces (cualquier interfaz)
- En el caso de clases, si no hay relación de herencia entre el tipo del objeto y la clase indicada como segundo operador, error de compilación:

```
String s="my cad";  
if(s instanceof Integer){ //error de compilación  
  
}
```

Nueva funcionalidad

➤ Desde Java 16 se puede utilizar *instanceof* para asignar el objeto a una variable del tipo específico, sin realizar un *cast*.

Antes de Java 16

```
Object obj=new String("Mi cadena");  
if(obj instanceof String) {  
    String s=(String)obj;  
    System.out.println("Longitud: "+s.length());  
}
```

Desde Java 16

```
Object obj=new String("Mi cadena");  
if(obj instanceof String s) {  
    System.out.println("Longitud: "+s.length());  
}
```

se convierte y se asigna
directamente a la variable

Revisión conceptos



Dado el siguiente bloque de código, ¿Cuál será el resultado de su ejecución?:

```
String cad=null;
Object ob=cad;
if(ob instanceof String s){
    System.out.println(s.length());
}
```

- a. Error de compilación
- b. Se muestra 0
- c. Excepción NullPointerException
- d. Ejecuta sin errores pero no muestra nada

Respuesta

La respuesta es la **d**. Al aplicar instanceof con un valor *null*, el resultado es directamente false, por lo que no se cumple la condición y no se muestra nada