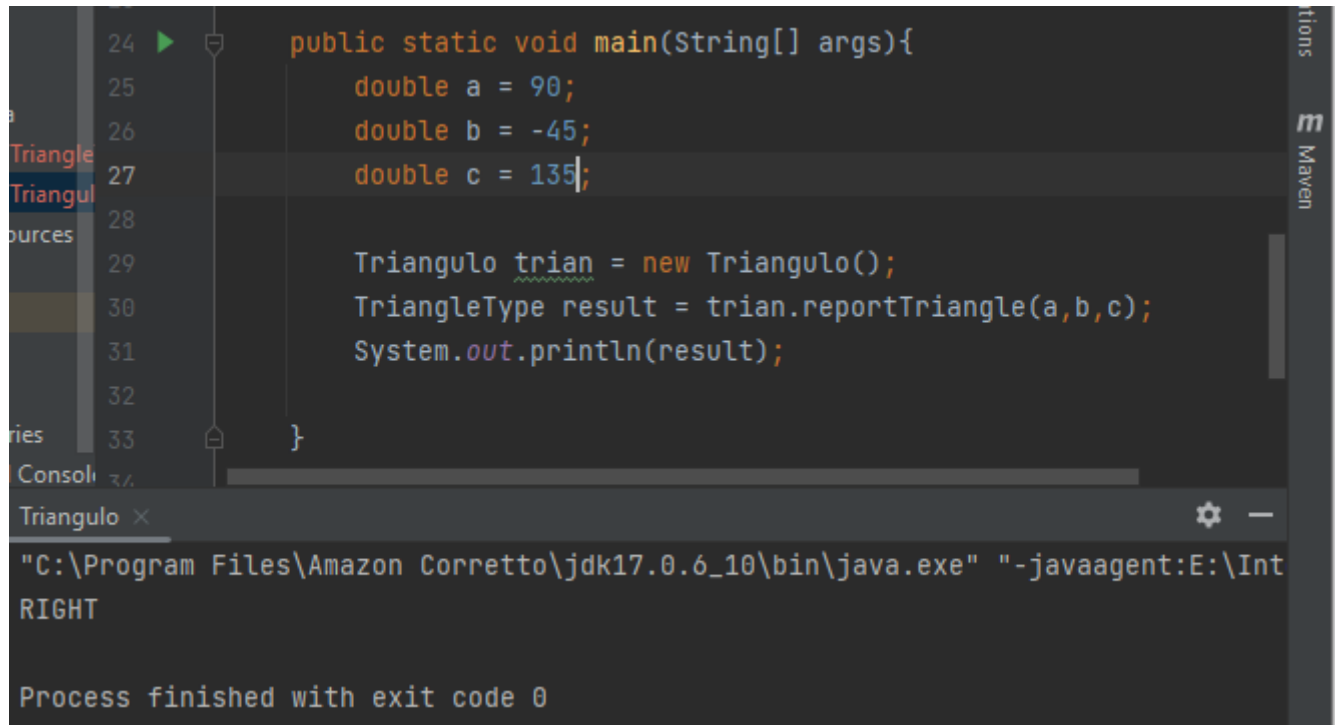


## DISEÑO POR CONTRATO

- **Regla de precondition asumida.**

El diseño por contrato adopta el diseño de precondition. El método asume que tu precondition siempre se cumple.



```
24 public static void main(String[] args){
25     double a = 90;
26     double b = -45;
27     double c = 135;
28
29     Triangulo trian = new Triangulo();
30     TriangleType result = trian.reportTriangle(a,b,c);
31     System.out.println(result);
32
33 }
```

Console

Triangulo x

"C:\Program Files\Amazon Corretto\jdk17.0.6\_10\bin\java.exe" "-javaagent:E:\Int  
RIGHT

Process finished with exit code 0

¿Cuáles son los resultados para : (90, 45, 45) = RIGHT, (120, 40, 20) = OBTUSE  
(50, 60, 70) = ACUTE , ¿Qué sucede con (90, -45, 135) = RIGHT?.

¿Quién es el responsable de este fallo, el proveedor o el cliente?. El responsable es el Cliente por no cumplir con las precondiciones.

Corrige este error.

```
if(a<0 || b<0 || c<0 || (a+b+c)!=180 ){
    return TriangleType.ERROR;
}
```

- **Reglas de violación de pre/postcondiciones**

La violación de una precondition o postcondition en tiempo de ejecución indica la existencia de un error.

La regla de violación de la precondition establece que una violación de la precondition es la manifestación de un error en el cliente, independientemente de si la postcondition se cumple o no.

La regla de violación de la postcondition establece que una violación de la postcondición es la manifestación de un error en el proveedor porque el proveedor no cumplió con su contrato.

Pregunta: En el ejemplo anterior indica una violación de precondition.

Se viola la precondition de  $b > 0$ , se digito  $b = -45$ .

```

* @param letra PRE: Letra : {'a'-'z' , 'A'-'Z'}
* @return POST: true si letra {a, e, i ,o, u, A, E, I, O, U}
*/
boolean isVowel(char letra){
    String vowels = "aeiouy&@";
    char ch = Character.toLowerCase(letra);
    return vowels.indexOf(ch) >= 0;
}

```

¿Qué sucede cuando `letra = '@'` ?.

```

char letra = '@';
Vocal vocal = new Vocal();
boolean resultado2 = vocal.isVowel(letra);
System.out.println(resultado2);

```

main ×

"C:\Program Files\Amazon Corretto\jdk17.0.6\_10\bin\java.exe"  
true

Sale True porque así lo definimos en "vowels", pero viola la precondition.

- Regla de precondition razonable
- Regla de disponibilidad de precondition.

## CAMBIO DE CONTRATO

Considera el siguiente método **getCell()** en el programa TicTacToe.

Pregunta: ¿Cómo cambia la precondition y la postcondition en el ejemplo anterior?

La precondition no cambia, La postcondition cambia , me retorna el valor de la celda si cumple con la precondition y si no cumple me retorna -1.