



LINK TO THE PATH

```
enum Direction { North, East, South, West }

public class Main {
    public static final String label(Direction d) {
        switch (d) {
            case North : return "north";
            case East : return "east";
            case South : return "south";
            //case West : return "west";
        }
        throw new IllegalArgumentException("not a valid direction");
    }

    public static void main(String args[]) {
        System.out.println(label(Direction.East));
        System.out.println(label(Direction.West));
    }
}
```



Exception in thread 'main' java.lang.IllegalArgumentException: not a valid direction



RuntimError

Le campilateur ne sait pas décompiler si nous avons traité tous les cas!!!

MAGNET 2-DUALS - TRANSLES

Indice sur le problème



LINK TO THE PATH

UTILISATION DES ENUM EN JAVA ... MAIS EN MODIFIANT LE CODE



Indice sur le problème



```
enum Direction { North, East, South, West }

public class Main {
    public static final String label(Direction d) {
        switch (d) {
            case North : return "north";
            case East : return "east";
            case South : return "south";
            //case West : return "west";
        }
        throw new IllegalArgumentException("not a valid direction");
    }

    public static void main(String args[]) {
        System.out.println(label(Direction.East));
        System.out.println(label(Direction.West));
    }
}
```

Runtime ERROR

Exception in thread "main" java.lang.IllegalArgumentException: not a valid direction

Le compilateur ne sait pas détecter si nous avons traité tous les cas !!!

LINK TO THE PATH

IMPLÉMENTONS NOTRE ENUM EN JAVA ... IL PARAÎT QUE JAVA 14 A DU PATTERN MATCHING



```
interface Direction{};
class North implements Direction{};
class East implements Direction{};
class South implements Direction{};
class West implements Direction{};

public class Main {
    public static final String label(Direction d) {
        if(d instanceof North dn){
            return "north";
        }else if(d instanceof East de){
            return "east";
        }else if(d instanceof South ds){
            return "south";
        }else if(d instanceof West dw){
            return « west";
        }
    }

    public static void main(String args[]) {
        System.out.println(label(new East()));
        System.out.println(label(new West()));
    }
}
```