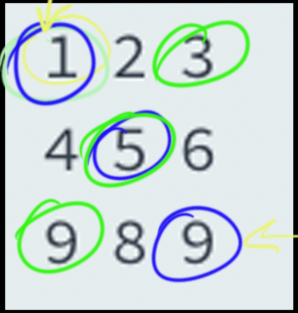


1	2	3
4	5	6
9	8	9

$1+5+9=15$

$3+5+9=17$

$|15-17|=2$

data.get(0).get(0)

data.get(2).get(2)

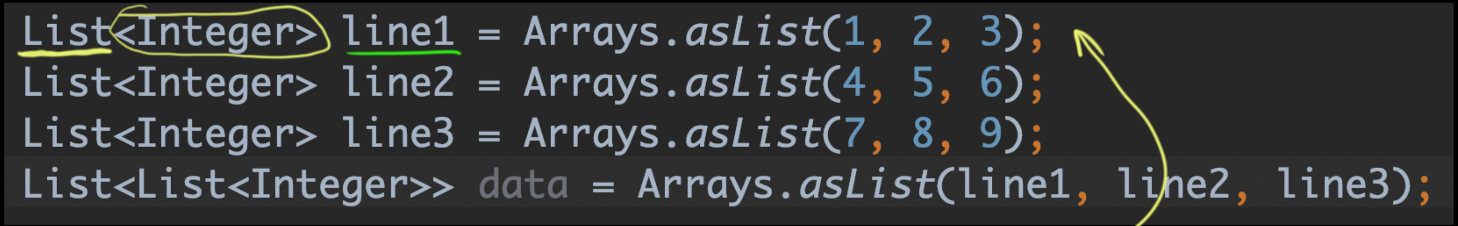
List<List<Int>>

List<Int>

Int

1	x	x
x	5	x
x	x	9

x	x	3
x	5	x
9	x	x



```java
List<Integer> line1 = Arrays.asList(1, 2, 3);
List<Integer> line2 = Arrays.asList(4, 5, 6);
List<Integer> line3 = Arrays.asList(7, 8, 9);
List<List<Integer>> data = Arrays.asList(line1, line2, line3);
```

int[] line1 = {1, 2, 3} <

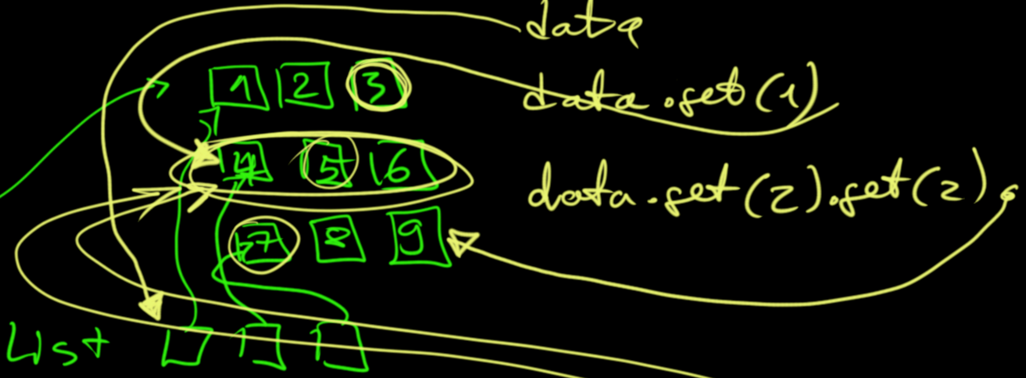
Person[] ≠ List<Person>

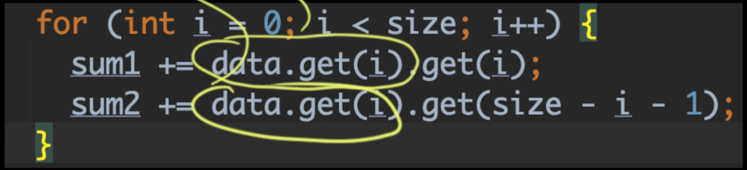
List<Integer>

line1

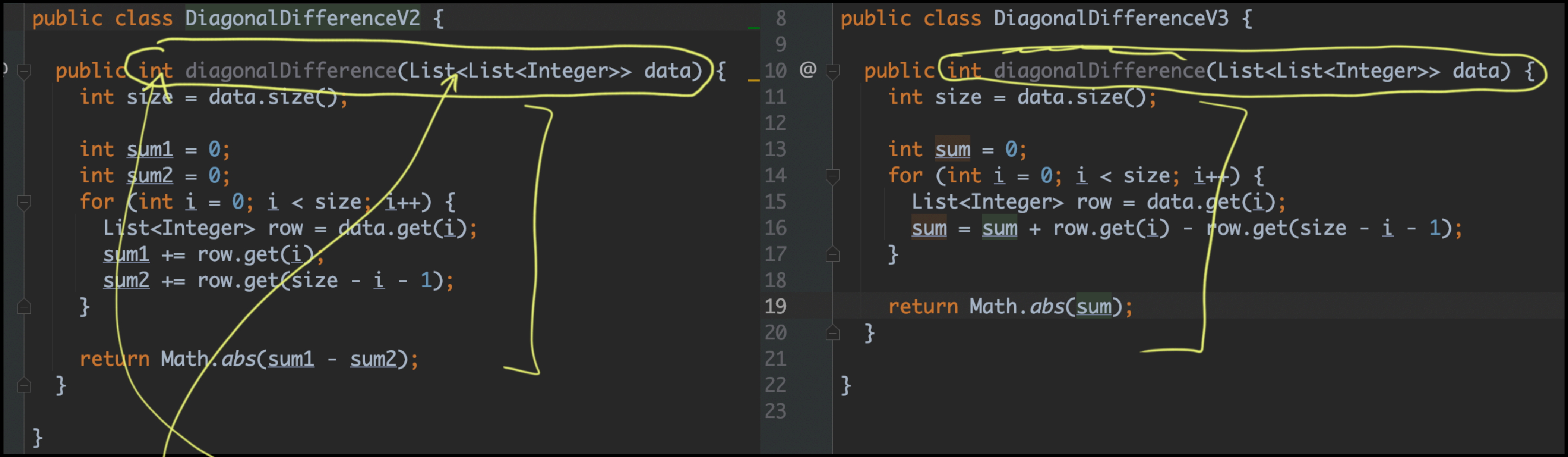
line2

data List<List<Integer>>





```java
for (int i = 0; i < size; i++) {
  sum1 += data.get(i).get(i);
  sum2 += data.get(i).get(size - i - 1);
}
```



```java
public class DiagonalDifferenceV2 {

    public int diagonalDifference(List<List<Integer>> data) {
        int size = data.size();

        int sum1 = 0;
        int sum2 = 0;
        for (int i = 0; i < size; i++) {
            List<Integer> row = data.get(i);
            sum1 += row.get(i);
            sum2 += row.get(size - i - 1);
        }

        return Math.abs(sum1 - sum2);
    }

}
```

```java
public class DiagonalDifferenceV3 {

    public int diagonalDifference(List<List<Integer>> data) {
        int size = data.size();

        int sum = 0;
        for (int i = 0; i < size; i++) {
            List<Integer> row = data.get(i);
            sum = sum + row.get(i) - row.get(size - i - 1);
        }

        return Math.abs(sum);
    }

}
```

f : A ⇒ B

name

input type

output type

add (Double, Double) -> Double

add (Double, Double) -> Int

different

we cant have both in one class

overl.

add (Double, Double) -> Double

add (Int, Int) -> Double