3Sum

Try to solve the 3Sum problem.

We'll cover the following ⌃

- Statement
- Examples
- Understand the problem
- Figure it out!
- Try it yourself

Statement

Given an array of integers, `nums`, and an integer value, `target`, determine if there are any three integers in `nums` whose sum is equal to the `target`, that is, `nums[i] + nums[j] + nums[k] == target`. Return TRUE if three such integers exist in the array. Otherwise, return FALSE.

> **Note:** A valid triplet consists of elements with *distinct* indexes. This means, for the triplet `nums[i]`, `nums[j]`, and `nums[k]`, $i \neq j$, $i \neq k$ and $j \neq k$.

Constraints:

- $3 \leq$ `nums.length` ≤ 500
- $-10^3 \leq$ `nums[i]` $\leq 10^3$
- $-10^3 \leq$ `target` $\leq 10^3$

Examples

Sample example 1

Input

| target | 20 |

| nums | 3 | 7 | 1 | 2 | 8 | 4 | 5 |

Output

| result | True |

Understand the problem

Let's take a moment to make sure you've correctly understood the problem. The quiz below helps you check if you're solving the correct problem:

Sum of Three Values

1   What should be the output if the following set of inputs is provided?

nums = [2, 3, 1]

target = 6

A) True

B) False

›

Figure it out!

We have a game for you to play. Rearrange the logical building blocks to develop a clearer understanding of how to solve this problem.

Sequence - Vertical ⓘ

Drag and drop the cards to rearrange them in the correct sequence.

0	Sort the input array in ascending order.
1	Iterate over the entire sorted array to find the triplet whose sum is equal to the target.
2	Adjust the calculated sum value, until it becomes equal to the target value, by conditionally moving the pointers, `low` and `high` .

?

T$_T$

☼

3

Return TRUE if the required sum is found. Otherwise, return FALSE.

4

In each iteration, make a triplet by storing the current array element and the other two elements using two pointers (`low` and `high`), and calculate their sum.

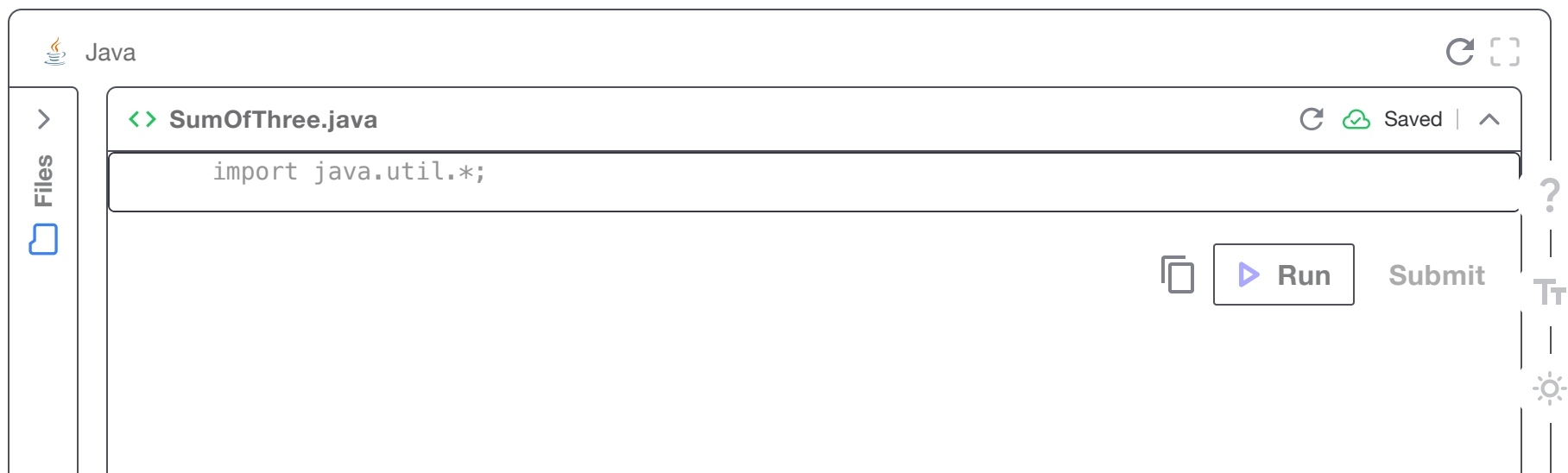Reset   Show Solution   Submit

Try it yourself

Implement your solution in `SumOfThree.java` in the following coding playground. The supporting code template provided in `TwoPointers.java` is meant to assist in developing your solution to the problem.

Java   ⟳ ⛶

<> **SumOfThree.java**   ⟳ ☁ Saved | ∧

```java
import java.util.*;
```

⧉   ▷ Run   Submit

Test Cases   Results   Code Feedback BETA

3Sum

Mark As Completed   Next →