.vscode\Knapsack.java

```java
import java.util.Arrays;
import java.util.Comparator;

class Item {
    int value;
    int weight;

    Item(int value, int weight) {
        this.value = value;
        this.weight = weight;
    }
}

public class FractionalKnapsack {

    // Function to get the maximum value in the knapsack
    public static double getMaxValue(Item[] items, int capacity) {
        // Sort items by their value-to-weight ratio in descending order
        Arrays.sort(items, new Comparator<Item>() {
            public int compare(Item a, Item b) {
                double r1 = (double) a.value / a.weight;
                double r2 = (double) b.value / b.weight;
                return Double.compare(r2, r1);
            }
        });

        double totalValue = 0.0; // Total value of items in the knapsack

        for (Item item : items) {
            if (capacity <= 0) {
                break; // If the knapsack is full, break the loop
            }

            // If the item's weight is less than the remaining capacity
            if (item.weight <= capacity) {
                totalValue += item.value; // Take the whole item
                capacity -= item.weight; // Decrease the remaining capacity
            } else {
                // Take the fractional part of the item
                totalValue += item.value * ((double) capacity / item.weight);
                capacity = 0; // The knapsack is now full
            }
        }

        return totalValue; // Return the maximum value obtained
    }

    public static void main(String[] args) {
```

```
        Item[] items = {
            new Item(60, 10),
            new Item(100, 20),
            new Item(120, 30)
        };

        int capacity = 50; // Capacity of the knapsack

        double maxValue = getMaxValue(items, capacity);
        System.out.printf("Maximum value in the knapsack = %.2f\n", maxValue);
    }
}


Maximum value in the knapsack = 240.00

```