

Name(s) _____ Period _____ Date _____

Extra Practice - Project Planning



Check for Understanding

The following method is intended to return an `int` containing the element at position `index` in the array `numbers`. For example, if `numbers` contains `[1, 2, 3, 4, 5]`, then `getValue(numbers, 3)` should return 4.

```
/* missing precondition */
public int getValue(int[] numbers, int index) {
    return numbers[index];
}
```

Which of the following is the most appropriate precondition for the method so that it does not cause an error?

- A. `/* Precondition: index > numbers.length */`
- B. `/* Precondition: index >= numbers.length */`
- C. `/* Precondition: 0 <= index <= numbers.length - 1 */`
- D. `/* Precondition: 0 < index < numbers.length - 1 */`
- E. `/* Precondition: 0 <= index <= numbers.length */`

AP Exam Prep

Given the method shown below, provide an appropriate precondition and postcondition.

```
public double getSalePrice(Dessert aDessert, int orderQuantity) {
    double salePrice = aDessert.getPrice();

    if (orderQuantity >= 12) {
        salePrice -= 0.75;
    }
    else {
        salePrice -= 0.50;
    }

    return salePrice;
}
```

Extra Practice

Do This: In the `PainterPlus` class, we wrote the method `paintFullLine()` shown below. Add Javadocs documentation for this method that explains the purpose of the method, any parameters and returns, and the precondition and postcondition for the method.

```
public void paintFullLine(String color) {
    while (canMove()) {
        paint(color);
        move();
    }
    paint(color);
}
```