

629971

Suppose you are defining a Java `ProductItem` class to store information about the inventory in a store, and you want to give the class an instance variable for the number of those items currently in stock. Choose the best datatype for modeling this number:

- a. `double`
- b. `float`
- *c. `int`
- d. a user-defined `NumberInStock` class
- e. `String`
- f. `"`
- g. `"`
- h. `"`
- i. `"`
- j. `"`

General Feedback:

A and B are wrong because you want to model an integer, not a floating point number.

E is wrong because you probably want to use this number for adding and subtracting, not just for display.

D is less good than C, because the built-in operations on ints will most likely be sufficient for the purposes of this variable.

633604

What will the following code print, assuming that `N` is a positive integer?

```
int count=0;
for (int i=0; i<N; i++) {
    if (i % 2 == 0) {
        count++;
    }
}
System.out.println(count);
```

- a. `N`
- b. `N/2`
- c. `N/2+1`
- *d. `(N+1)/2`
- e. 0 (i.e. the loop body never executes)
- f. `"`
- g. `"`
- h. `"`

- i. "
- j. "

General Feedback:

If N is even, for example N=12, $N/2$ works. However, suppose N is odd, for example N=13. Here the result should be 7, which is $(N+1)/2$. However, for N=12, $(N+1)/2$ also works. $(N+1)/2$ works for all positive integer.

634906

Two algorithms accomplish the same task on a collection of N items. Algorithm A performs $(N/2)^3$ operations. Algorithm B performs N^2 operations. Under what conditions does algorithm A offer strictly better performance?

- *a. $N \leq 7$
- b. $N \leq 8$
- c. $N > 8$
- d. For all N
- e. For no N
- f. "
- g. "
- h. "
- i. "
- j. "

General Feedback:

For $N \leq 7$, $(N/2)^3 < N^2$.