

Solutions

Each answer shows the steps, then the **result**.

2.1

- an `ArrayList` holds **objects** only; the type must be the wrapper, `Integer`

2.2

- **unboxing**: the `Double` from `get` becomes a `double`

2.3

- `12 + 3 = 15`
- string concatenation: `"123"`

2.4

- it overflows to `Integer.MAX_VALUE`, $2^{31} - 1$

3.1 C

- `get` returns an `Integer`, stored in an `int`

3.2 C

- `total += n` unboxes `null`

3.3 B

- 0 and -1 are larger than every value of an all-negative array

3.4

(a)

```
ArrayList<Integer> values = new ArrayList<Integer>();
for (String s : typed)
{
    values.add(Integer.parseInt(s));
}
```

- creates an empty `ArrayList<Integer>`
- traverses every `String` in `typed`
- adds `Integer.parseInt(s)` for each

(b)

- in `values.add(...)`: the `int` from `parseInt` is boxed to an `Integer`

(c)

```
int smallest = Integer.MAX_VALUE;
for (int v : values)
{
    if (v < smallest) smallest = v;
}
```

- starts at `Integer.MAX_VALUE`
- traverses `values`, unboxing each
- replaces `smallest` when a value is smaller