

Answer Key

Line	Answer
1–3	No output. Three accounts are created: Ada #1 (\$200), Ben #2 (\$0), Cy #3 (\$50).
4	CE: there is no constructor with no parameters. The line is removed, so no account is created and the count stays at 3.
5	3
6	Account #3 (Cy): \$50.00
7–8	No output. Ben has \$75. The negative deposit is ignored.
9	75.0
10	CE: balance is private.
11	false (Ada has only \$200)
12	true (Ada now has \$80, Cy has \$170)
13	Account #1 (Ada): \$80.00 / Account #3 (Cy): \$170.00
14	CE: there is no setBalance method.
15	No output. Cy's \$170 moves to Ben, so Cy has \$0 and Ben has \$245.
16	245.0 (245.0 + 0.0; it adds numbers, not strings)
17–18	No output. e is an alias for b, so Ben now has \$200.
19	200.0
20	true
21	false (different account numbers)
22	CE: MIN_BALANCE is final.
23–24	No output. The rate is now 50%, and Ben grows to \$300.
25	Account #2 (Ben): \$300.00
26	CE: getBalance is an instance method, so it can't be called on the class.
27	CE: computeInterest is private.
28	No output. Ada's account now belongs to Ava.
29	Account #1 (Ava): \$80.00 (the owner changed; the balance is still \$80)
30	Ben has \$300.0 (a double concatenated with a String prints as 300.0, not \$300.00)
31	No output. Gus is account #4.
32	4
33	No output. h is null.
34	RE: NullPointerException. The program stops here.
35	Never runs, so nothing is printed.

Part B

1. Change line 4 to new BankAccount("Dee") (or pass a name and a balance). Or add a no-argument constructor to BankAccount.
2. System.out.println(a.getBalance()); An instance method needs an object to be called on.
3. e = b copies the reference, not the object. e and b point to the same BankAccount.
4. Line 4 never compiled, so that constructor never ran. Only three accounts existed before Gus, so Gus is #4.
5. Every account, including ones created later. interestRate is a static field shared by the whole class, and static methods are called with the class name. (Only Ben gets interest in this program, because addInterest is only called on b.)
6. Nothing changes. amount is Ada's whole balance. Withdrawing it takes her to \$0, and depositing it brings her back. The two parameters are aliases for the same object.