Answer: The contain method of List call the equal method of Employee to check if they are equal. **The equal method was not overridden correctly**. In fact, it was not overridden at all. The equal method implemented in the employee class was wrong as shown below. Therefore, it was never called, but instead the equal method of base **Object** class was called, and this method returns true if and only if e1 and e1 refer to the same object. Since the comparison was between two different instance of Employee present in two different list, the *listsAreEqual* failed

***Wrong:***

**public** **boolean** equals(Employee e) {

**return** e.name.equals(name) && e.salary == salary;

}

@Override

**public** **boolean** equals(Object ob) {

**if**(ob == **null**) **return** **false**;

**if**(!(ob **instanceof** Employee)) **return** **false**;

Employee emp = (Employee)ob;

**return** emp.name.equals(name) && emp.salary == salary;

}