Instantiable class for Reservation objects:

public class Reservation {

private int resNumber;

private String resCustName;

private boolean resFreqFlyer;

//parameterized constructor

Reservation(int resNum, String resName, boolean freqFlyer){

setResNumber(resNum);

setResCustName(resName);

setResFreqFlyer(freqFlyer);

}

//setters

public void setResNumber(int num){

resNumber = num;

}

public void setResCustName(String name){

resCustName = name;

}

public void setResFreqFlyer(boolean flyer){

resFreqFlyer = flyer;

}

//getters

public int getResNumber(){

return resNumber;

}

public String getResCustName(){

return resCustName;

}

public boolean getResFreqFlyer(){

return resFreqFlyer;

}

}

Test class used to create Reservation array:

public class TestReservation{

public static void main(String[] args) {

Reservation[] reservations; //statement 1

reservations = new Reservation[4]; //statement 2

reservations[3] = new Reservation(1001, "Mickey Mouse", True); //statement 3

reservations[2].setResNumber = 2000; //statement 4

reservations[1] = new Reservation(); //statement 5

reservations[4] = new Reservation(1002, "Minnie Mouse", False); //statement 6

}

}

Identify what memory looks like after each statement, and/or identify exceptions. Use the back of the sheet if necessary.

Memory after statement 1:

Memory address allocated to variable named reservations

Reservations address 1000 contains a null value

|  |
| --- |
| null |

Memory after statement 2:

Memory address 1000 now contains an array of null Reservation objects

Address 1000

|  |
| --- |
| Null |
| Null |
| Null |
| Null |

Memory after statement 3:

Memory is allocated to a new Reservation object, that address is stored in the reservations array at index[3]

Address 1000 Address 2000

|  |
| --- |
| Null |
| Null |
| Null |
| 2000 |

|  |
| --- |
| 1001 |
| Mickey Mouse |
| true |

Memory after statement 4:

I see 2 issues with this statement.

1. setResNumber is a method that needs 2000 to be passed as an argument. Not set equal like a variable.

2. If it were coded correctly index 2 in the reservations array is null. It is not pointing to a Reservation object to set a variable.

Memory after statement 5:

Memory is allocated for new Reservation object using default parameters. Memory location is stored in reservations array index[1]

Address 1000 Address 2000 Address 3000

|  |
| --- |
| 1001 |
| Mickey Mouse |
| true |

|  |
| --- |
| Null |
| 3000 |
| Null |
| 2000 |

|  |
| --- |
| 0 |
| null |
| false |

Memory after statement 6:

Exception index[4] is out of range for reservations array