Lecture 10 - In Class Exercise

Goal: A deeper dive into ISP

1 Set Intersection

Instructions: Work with your neighbors in groups of 2.

```
public static Set intersection (Set s1, Set s2)
/**
 * @param  s1, s2 : sets to compute intersection of
 * @return a (non null) Set equal to the intersection of Sets s1 and s2
 * @throws NullPointerException if s1 or s2 is null
*/

Characteristic: Type of s1
- s1 = null
- s1 = {}
- s1 has at least one element

Characteristic: Relation between s1 and s2
- s1 and s2 represent the same set
- s1 is a subset of s2
- s2 is a subset of s1
- s1 and s2 do not have any elements in common
```

Based on the code above, answer the following questions:

1. Does the partition for the characteristic "Type of s1" satisfy the completeness property? If not, give a value for s1 that does not fit in any block.
2. Does the partition for the characteristic "Type of s1" satisfy the disjointness property? If not, give a value for s1 that fits in more than one block.
3. If necessary, fix "Type of s1".
4. Repeat the prior 3 steps for the characteristic "Relation between s1 and s2".
5. If the "Base Choice" criterion were applied to the two partitions (exactly as written), how many test requirements would result?
6. If the "Base Choice" criterion were applied to the repaired partitions, how many test requirements would result? Write out these test requirements.
7. Are all of these feasible? If not, what should happen with the infeasible requirements?
8. Refine the test requirements into tests.
9. If the "Pair Wise" criterion were applied to the repaired partitions, how many test requirements would result? Write out these test requirements. How many would be feasible?