1. What are the differences between Lists, Sets, and Maps in Java?

   They are interfaces. They have different data structures.

   List allows duplicates and null values and it saves insertion of orders so you can retrieve data using index on other hand, set does not allow duplicates and it allows null value only once, it does not save insertion order so you can't retrieve using index. Map has key-value pairs and keys does not allows duplicates, keys allows null value only once, value allows null values.

   Difference Between List, Set and Map in Java - Program Talk

2. List at least two different implementations for each collection (List, Set, and Map). When would you use one of the implementations over the other?

   List has common two implementations: ArrayList and LinkedList. ArrayList is fast at accessing elements and slow at adding and removing elements than LinkedList.

   Set has three common implementations:HashSet, LinkedHashSet and TreeSet. HashSet is fast but elements are not in order. LinkedHashSet is slow on adding and removing elements but it keeps elements in order. TreeSet is slow but keeps elements sorted.

   Map has three common implementations: HashMap, LinkedHashMap, TreeMap. HashMap is fast but not in order elements, LinkedHashMap saves the order of elements, TreeMap keeps order of the keys, but it is slow when making insertion or inclusion testing.

   Implementations | Implementation Patterns: Collections | InformIT

3. Write a line of code that shows how you would instantiate an ArrayList of String.

   ```
   List<String> example = new ArrayList<String>();
   ```

4. Write a line of code that shows how you would instantiate a HashSet of StringBuilder.

   ```
   Set<String> example2 = new HashSet<>();
   ```

5. Write a line of code that shows how you would instantiate a HashMap of String, String.

   ```
   Map<String, String> example3 = new HashMap<>();
   ```

6. What is your favorite thing you learned this week?

I learned Collections and how they work.