CS61B sp24 week1

tips

1. **You can have more than one class in a single Java file, but only one of those classes can be public.**
   The public class name must match the file name, and the other classes must be package-private or have a lower level of visibility (default, protected, or private).
2. Java could only read string as command line argument
   `public static void main(String[] arguments) {}`
   main function could only accept `String[] arguments`, if want int/ float, we need *converting* or *parsing* it like this

```
public class MainClass {
    public static void main(String[] arguments) {
        if (arguments.length < 2) {
            System.out.println("Please provide an integer and a float as arguments.");
            return;
        }

        try {
            // Parse the first argument as an integer
            int intValue = Integer.parseInt(arguments[0]);
            // Parse the second argument as a float
            float floatValue = Float.parseFloat(arguments[1]);

            System.out.println("Integer value: " + intValue);
            System.out.println("Float value: " + floatValue);
        } catch (NumberFormatException e) {
            System.out.println("Invalid input. Please ensure you provide a valid integer and float.");
        }
    }
}
```

3. The diff between `List.of` and `Arrays.asList` methods in Java
   - `List.of` The list returned is immutable. This means that after the list is created, you cannot add, remove, or change elements in the list.
   - `Arrays.asList` The list returned is mutable in terms of its elements but has a fixed size. You cannot add or remove elements, but you can change existing elements. like use `set` method to change the element
   - eg.

```
public class ArraysAsListExample {

    public static void main(String[] args) {

        List<Integer> numbers = Arrays.asList(1, 2, 3, 4, 5);

        System.out.println(numbers); // Output: [1, 2, 3, 4, 5]

        numbers.set(0, 10); // Works fine

        System.out.println(numbers); // Output: [10, 2, 3, 4, 5]



        // Uncommenting the following lines will throw UnsupportedOperationException

        // numbers.add(6);

        // numbers.remove(1);

    }
```

```
}
```

3. for both method and variable there are 2 types of them : instance and static
    - If the method is going to be invoked by an instance of the class , then it should be non-static.
    - static variable
        - **Static Variable**: A static variable belongs to the class and is shared among all instances.
        - **Shared State**: Changes to a static variable are reflected across all instances of the class.
    - when we calling a static method, we have to invoke it using the class name
4. defining a class including declaration, instantiation and assignment
    - which also include invocation of methods