

1. This question involves the `Account` class, which is used to represent user accounts for a website. The `Account` class contains a helper method, `isAvailable`, which determines whether a username is available. You will write a constructor and a method in the `Account` class.

```
public class Account
{
    private String username; // To be initialized in part (a)

    /**
     * Creates a username based on the parameter requestedName. If the
     * username is unavailable, appends successive integers, beginning
     * with 1, to requestedName until an available username is found,
     * as described in part (a).
     */
    public Account(String requestedName)
    { /* to be implemented in part (a) */ }

    /**
     * Returns true if the parameter str is an available username;
     * returns false otherwise.
     */
    public static boolean isAvailable(String str)
    { /* implementation not shown */ }

    /**
     * Returns a shortened version of username with each hyphen ("-")
     * and the character before it removed, as described in part (b)
     * Preconditions: username does not start or end with a hyphen.
     *                  username does not contain consecutive hyphens.
     *                  username.length() >= 2
     * Postcondition: username is unchanged.
     */
    public String getShortenedName()
    { /* to be implemented in part (b) */ }

    /* There may be instance variables, constructors, and methods
       that are not shown. */
}
```

The following information applies to part A.

In part A, you will write the `Account` constructor, which creates a username based on the parameter `requestedName`.

A helper method, `isAvailable`, has been provided to determine whether a username is available. The `isAvailable` method returns `true` if its parameter is an available username and returns `false` otherwise.

If `requestedName` is an available username, the `Account` constructor will assign `requestedName` to the instance variable `username`. If not, the constructor will try different variations of `requestedName` until an available username is found and assigned to the instance variable `username`. The variations consist of `requestedName` followed by an integer, as in the following example.

- If `requestedName` is "Luis-Cruz" and this username is not available, then the constructor will check the availability of "Luis-Cruz1", "Luis-Cruz2", "Luis-Cruz3", and so on, until an available username is found. The first available username found is assigned to `username`.
- If `requestedName` is "PSmith" and this username is available, then "PSmith" is assigned to `username`.

The following information applies to part B.

In part B, you will write the `getShortenedName` method, which returns a shortened version of the instance variable `username` with each hyphen ("-") and the character immediately before it removed. If no hyphens appear in `username`, the value of `username` is returned.

For example, if `username` is "Amy-Marie-Lin", `getShortenedName` should return "AmMariLin".

As another example, if `username` is "SammyB3", `getShortenedName` should return "SammyB3".

Part A

Complete the `Account` constructor. You must use `isAvailable` appropriately in order to receive full credit.

```
/**
 * Creates a username based on the parameter requestedName. If the
 * username is unavailable, appends successive integers, beginning
 * with 1, to requestedName until an available username is found,
 * as described in part (a).
 */
public Account(String requestedName)
```

Part B

Complete method `getShortenedName`.

```
/**
 * Returns a shortened version of username with each hyphen ("-")
 * and the character before it removed, as described in part (b)
 * Preconditions: username does not start or end with a hyphen.
 *                  username does not contain consecutive hyphens.
 *                  username.length() >= 2
 * Postcondition: username is unchanged.
 */
public String getShortenedName()
```