Lab-7

Implement the Collection Framework and Lambda expression concept based on your domain.

```
import java.util.Arrays;

import java.util.List;



class Builder {

private String botName;

private String discordToken;



Builder(String botName, String discordToken) {

if (discordToken.isEmpty()) {

System.out.println("Token not found");

}

this.botName = botName;

this.discordToken = discordToken;

}



public void showBotDetails() {

System.out.println("Discord Bot with " + botName + " created in your
account");

}



public void createGuild(String guildName) {
```

```
System.out.println("Created a guild " + guildName + " associated to the
bot");

}



public void getBotPermission(int permission) {

boolean isAdmin = true;

boolean isMod = true;

boolean BAN_MEMBERS = true;

boolean KICK_MEMBERS = true;

if (permission == 0) {

System.out.println("The Bot created has 0 permissions");

}

if (isAdmin && isMod && BAN_MEMBERS && KICK_MEMBERS) {

System.out.println("The Bot is set with full permissions");

}

}



public void showBotDetails(String clientName) {

System.out.println("Discord Bot with " + botName + " created for client "
+ clientName);

}

}



class SlashCommands extends Builder {

private List<String> members;
```

```
SlashCommands() {

super("botName", "discordToken");

this.members = Arrays.asList(

"Elumeveguy", "Emojorekes", "Eroxihisom", "Ulelabutuk", "Ayibiciqeb",

"Oguyocuxas", "Uxibabujen", "Epiwimaguk", "Idenefibiy", "Amarebamat");

}


public void getMembers() {

System.out.println("Getting members of the text channel");

members.forEach(member -> System.out.println("Channel member: " +
member));

}


public void slashCommand(String user) {

System.out.println("Ban User ");

System.out.println("User " + user + " is banned");

}


public void filterMembersStartingWith(String prefix) {

System.out.println("Members starting with '" + prefix + "':");

members.stream()

.filter(member -> member.startsWith(prefix))

.forEach(System.out::println);

}

}
```

```
class Client extends SlashCommands {

private String name;

private String discordServer;


Client(String name, String discordServer) {

super();

this.name = name;

this.discordServer = discordServer;

}


public void getBotDetails() {

System.out.println("Your account name is " + name);

System.out.println("Your Discord server name is " + discordServer);

}

}


class DiscordBot {

public static void main(String[] args) {

Builder build = new Builder("discordbot1", "skffdjdfjfjddd11");

build.showBotDetails();

build.createGuild("Mointer");

build.getBotPermission(0);
```

```
SlashCommands cl = new SlashCommands();

cl.getMembers();

cl.slashCommand("userToBan");

cl.filterMembersStartingWith("E");



// Client

Client client = new Client("ram", "ram's server");

build.showBotDetails("ram");

client.getBotDetails();

}

}
```