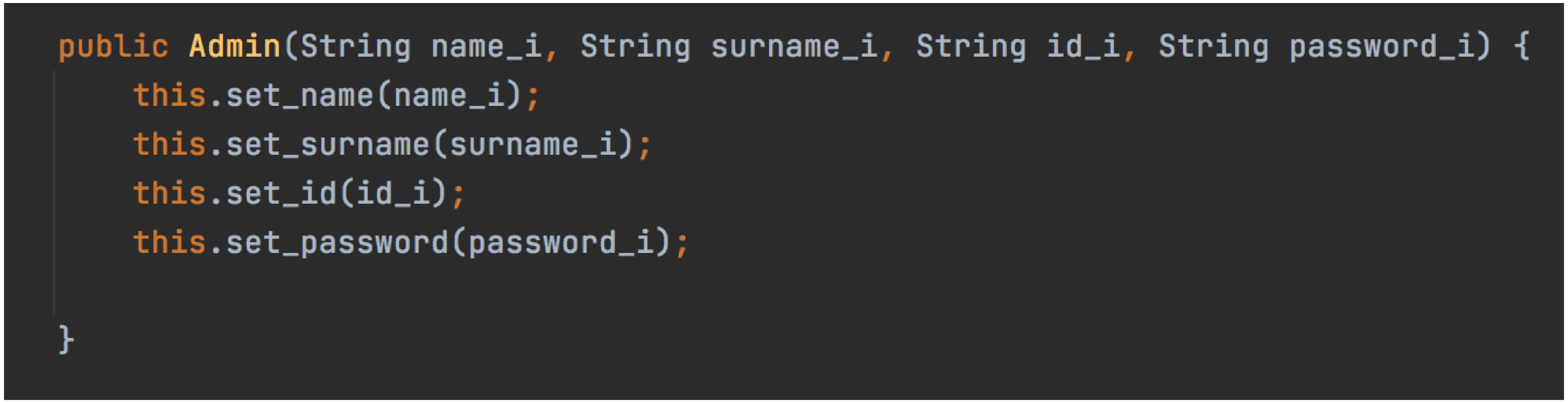
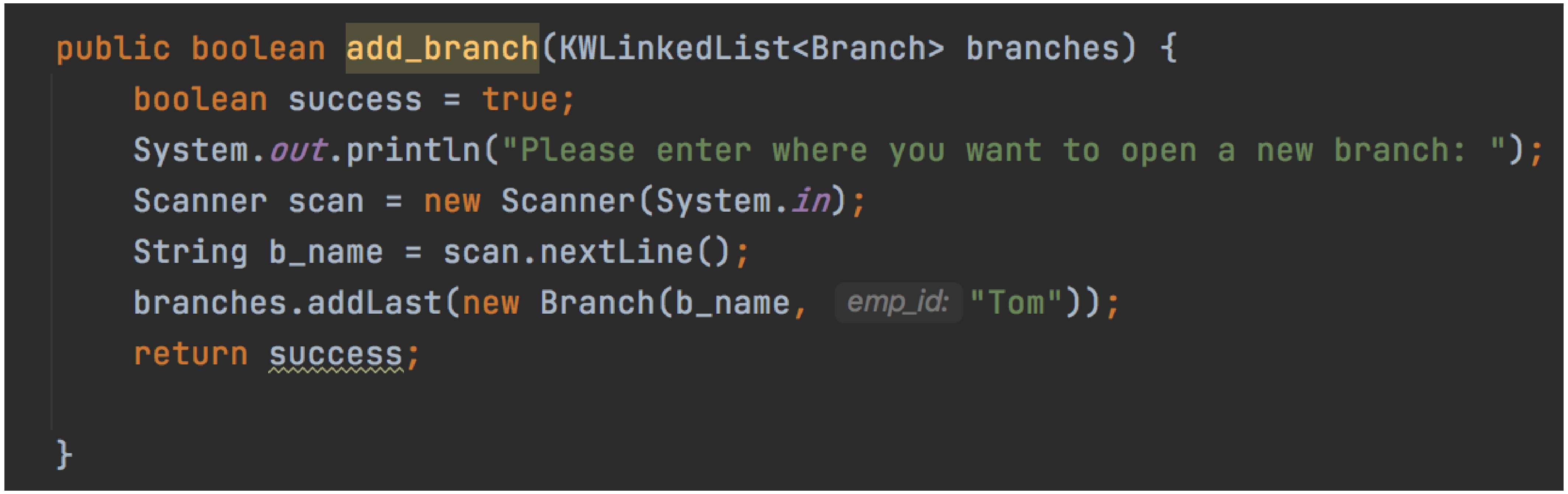
CSE 222 HOMEWORK 3 PART 2

1.



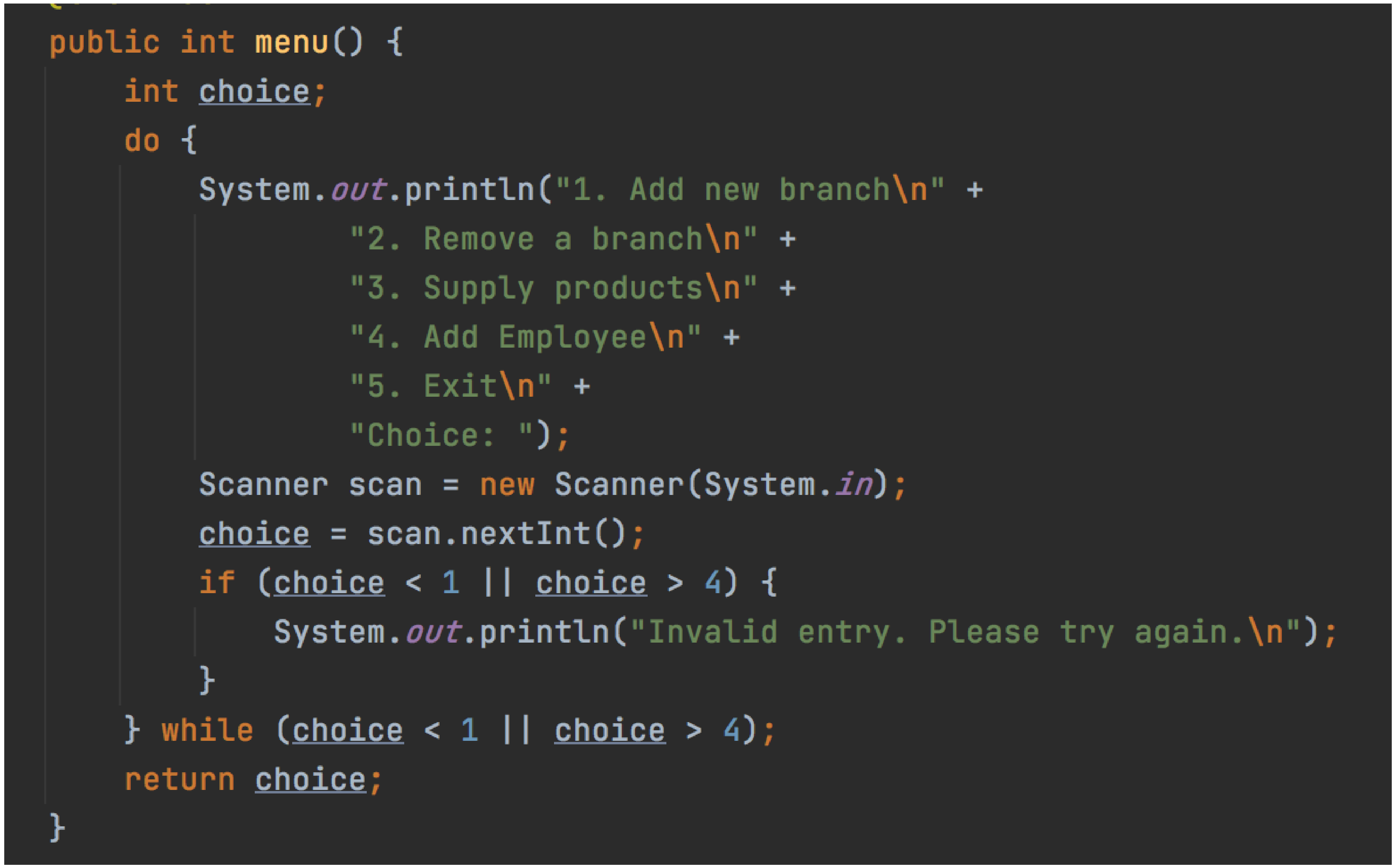
```java
public Admin(String name_i, String surname_i, String id_i, String password_i) {
    this.set_name(name_i);
    this.set_surname(surname_i);
    this.set_id(id_i);
    this.set_password(password_i);

}
```

2.



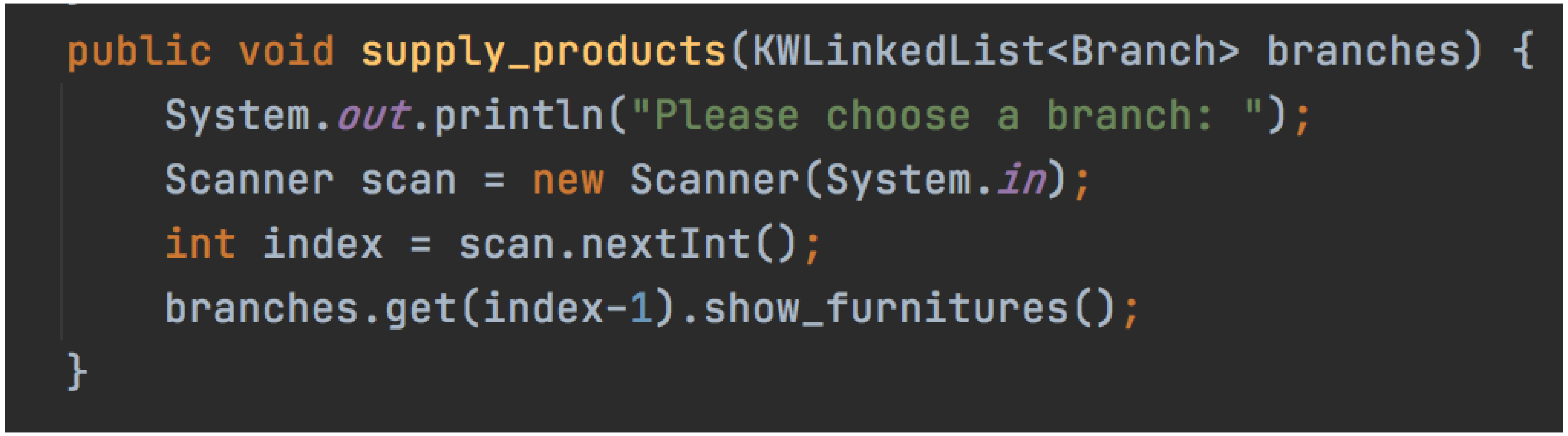
```java
public boolean add_branch(KWLinkedList<Branch> branches) {
    boolean success = true;
    System.out.println("Please enter where you want to open a new branch: ");
    Scanner scan = new Scanner(System.in);
    String b_name = scan.nextLine();
    branches.addLast(new Branch(b_name,  emp_id: "Tom"));
    return success;

}
```

3.



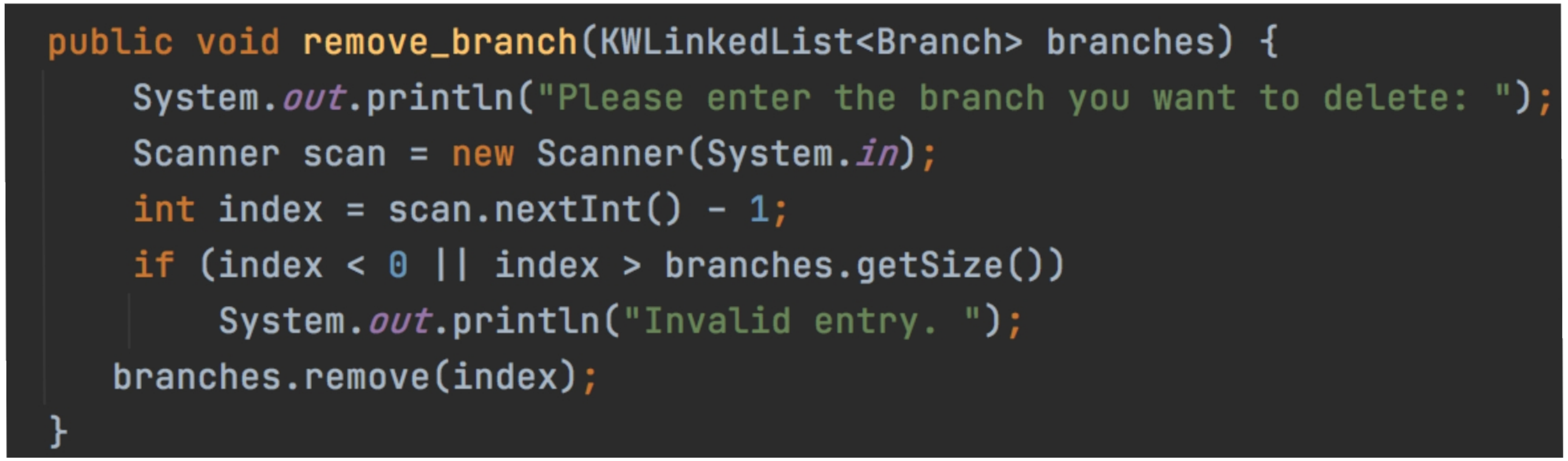
```
public int menu() {
    int choice;
    do {
        System.out.println("1. Add new branch\n" +
                "2. Remove a branch\n" +
                "3. Supply products\n" +
                "4. Add Employee\n" +
                "5. Exit\n" +
                "Choice: ");
        Scanner scan = new Scanner(System.in);
        choice = scan.nextInt();
        if (choice < 1 || choice > 4) {
            System.out.println("Invalid entry. Please try again.\n");
        }
    } while (choice < 1 || choice > 4);
    return choice;
}
```

4.



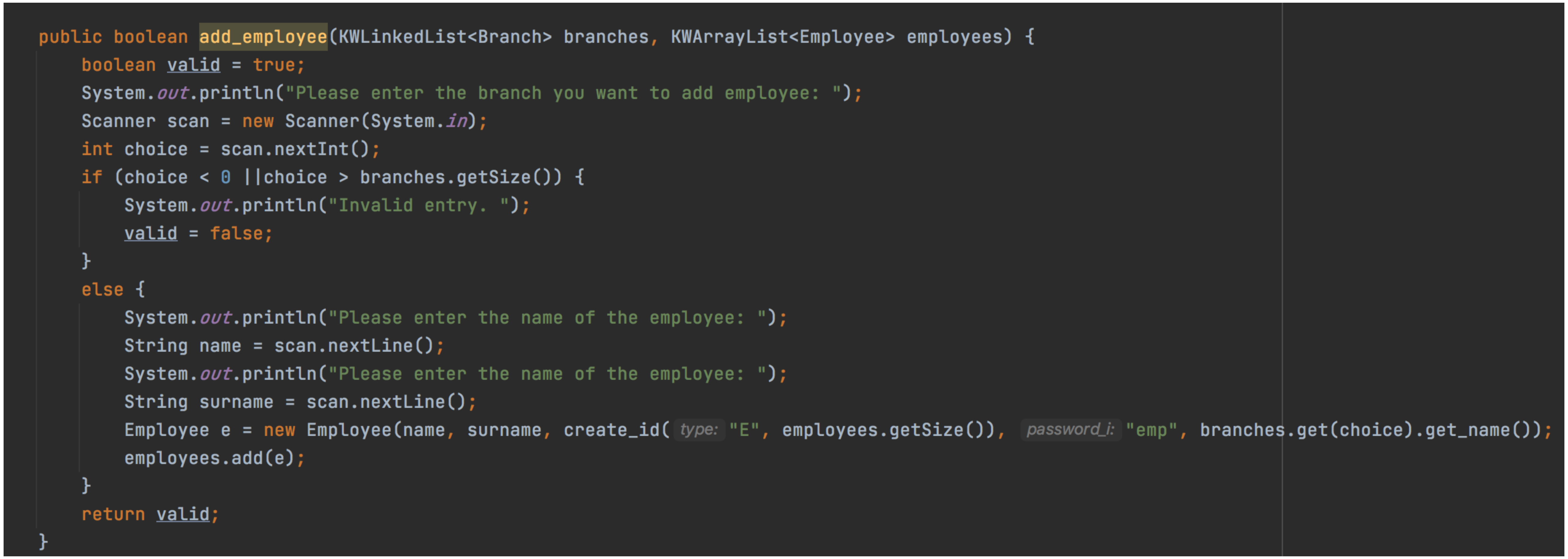
```
public void supply_products(KWLinkedList<Branch> branches) {
    System.out.println("Please choose a branch: ");
    Scanner scan = new Scanner(System.in);
    int index = scan.nextInt();
    branches.get(index-1).show_furnitures();
}
```

5.



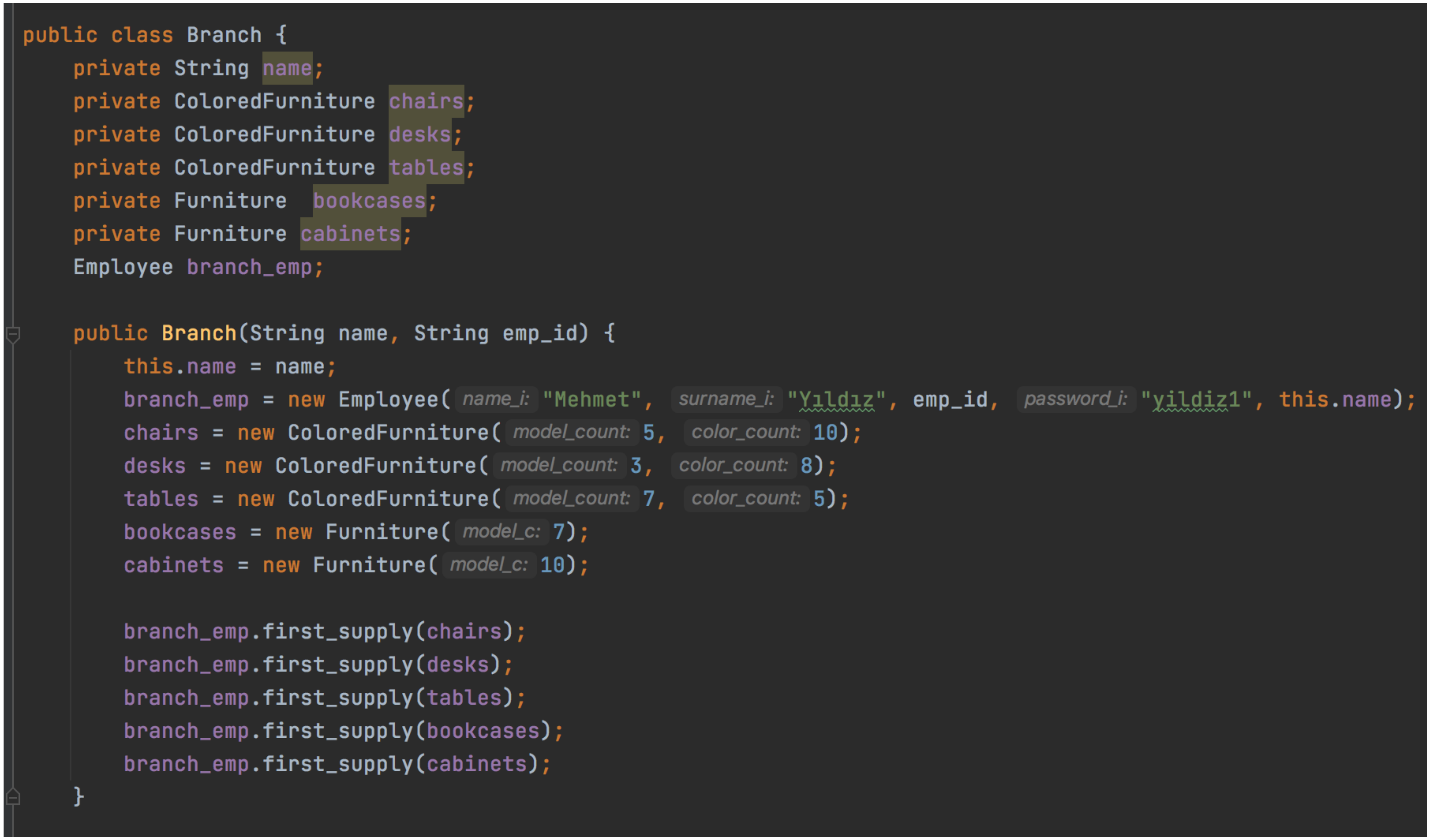
```java
public void remove_branch(KWLinkedList<Branch> branches) {
        System.out.println("Please enter the branch you want to delete: ");
        Scanner scan = new Scanner(System.in);
        int index = scan.nextInt() - 1;
        if (index < 0 || index > branches.getSize())
            System.out.println("Invalid entry. ");
      branches.remove(index);
}
```

6.



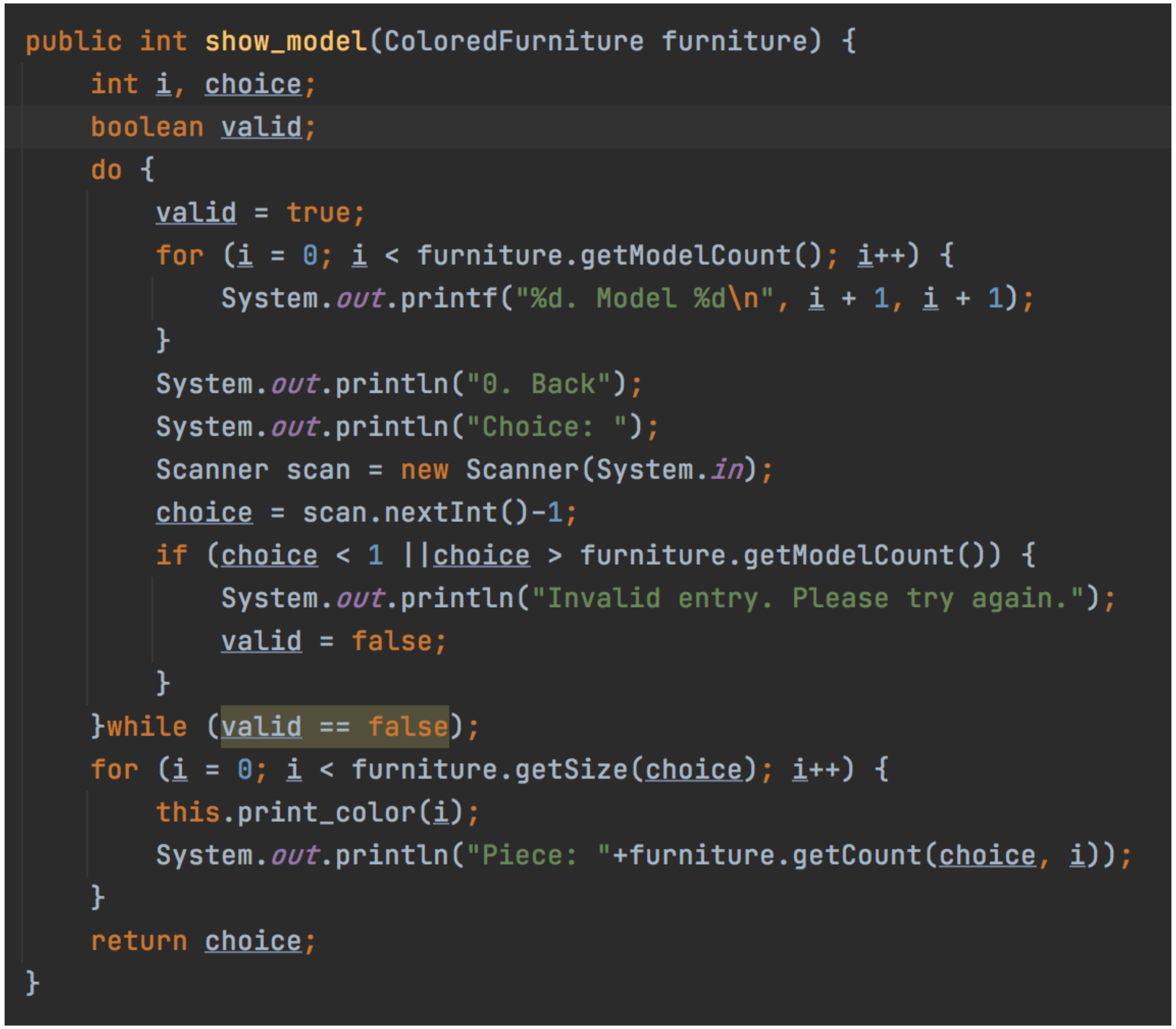
```java
public boolean add_employee(KWLinkedList<Branch> branches, KWArrayList<Employee> employees) {
    boolean valid = true;
    System.out.println("Please enter the branch you want to add employee: ");
    Scanner scan = new Scanner(System.in);
    int choice = scan.nextInt();
    if (choice < 0 ||choice > branches.getSize()) {
        System.out.println("Invalid entry. ");
        valid = false;
    }
    else {
        System.out.println("Please enter the name of the employee: ");
        String name = scan.nextLine();
        System.out.println("Please enter the name of the employee: ");
        String surname = scan.nextLine();
        Employee e = new Employee(name, surname, create_id( type: "E", employees.getSize()),  password_i: "emp", branches.get(choice).get_name());
        employees.add(e);
    }
    return valid;
}
```

7.



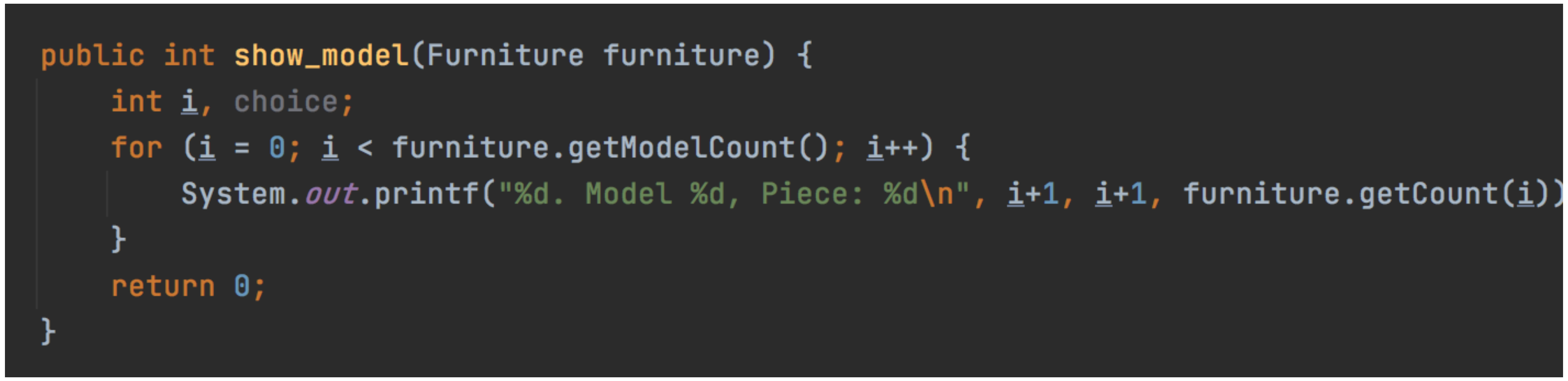
```java
public class Branch {
    private String name;
    private ColoredFurniture chairs;
    private ColoredFurniture desks;
    private ColoredFurniture tables;
    private Furniture  bookcases;
    private Furniture cabinets;
    Employee branch_emp;

    public Branch(String name, String emp_id) {
        this.name = name;
        branch_emp = new Employee( name_i: "Mehmet",  surname_i: "Yıldız", emp_id,  password_i: "yildiz1", this.name);
        chairs = new ColoredFurniture( model_count: 5,  color_count: 10);
        desks = new ColoredFurniture( model_count: 3,  color_count: 8);
        tables = new ColoredFurniture( model_count: 7,  color_count: 5);
        bookcases = new Furniture( model_c: 7);
        cabinets = new Furniture( model_c: 10);

        branch_emp.first_supply(chairs);
        branch_emp.first_supply(desks);
        branch_emp.first_supply(tables);
        branch_emp.first_supply(bookcases);
        branch_emp.first_supply(cabinets);
    }
```

8.



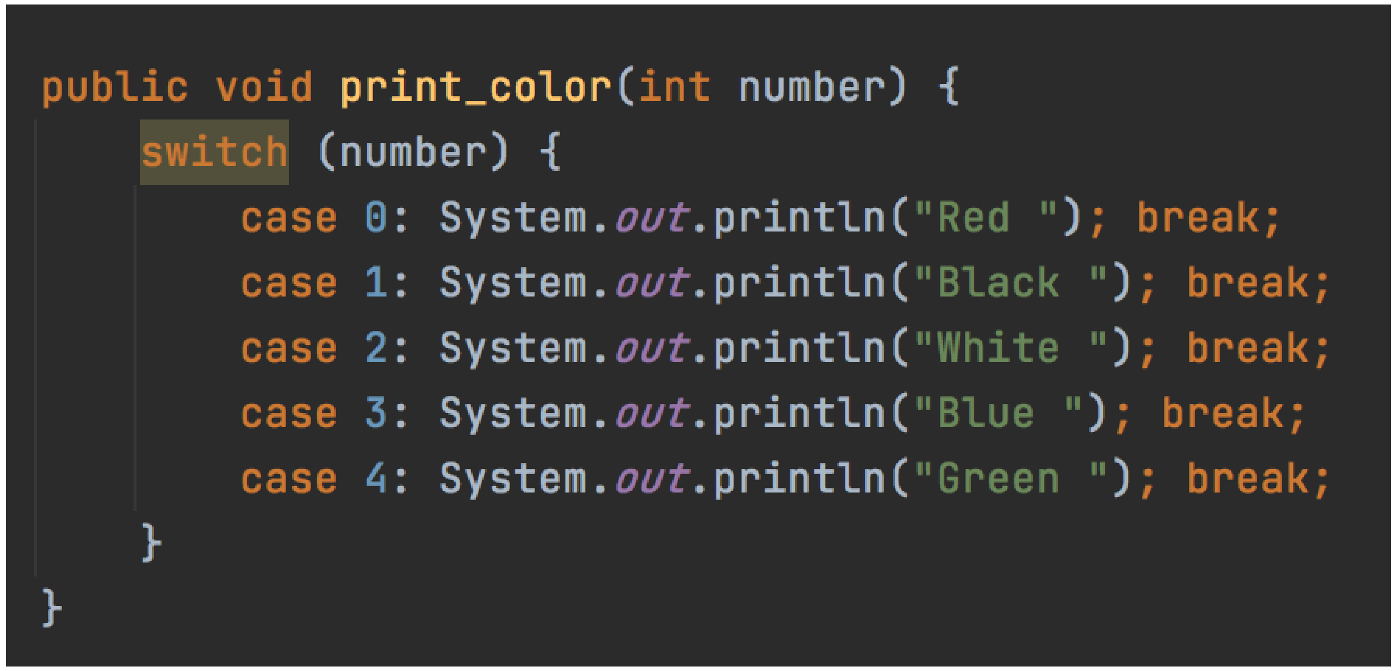
```java
public int show_model(ColoredFurniture furniture) {
    int i, choice;
    boolean valid;
    do {
        valid = true;
        for (i = 0; i < furniture.getModelCount(); i++) {
            System.out.printf("%d. Model %d\n", i + 1, i + 1);
        }
        System.out.println("0. Back");
        System.out.println("Choice: ");
        Scanner scan = new Scanner(System.in);
        choice = scan.nextInt()-1;
        if (choice < 1 ||choice > furniture.getModelCount()) {
            System.out.println("Invalid entry. Please try again.");
            valid = false;
        }
    }while (valid == false);
    for (i = 0; i < furniture.getSize(choice); i++) {
        this.print_color(i);
        System.out.println("Piece: "+furniture.getCount(choice, i));
    }
    return choice;
}
```

9.



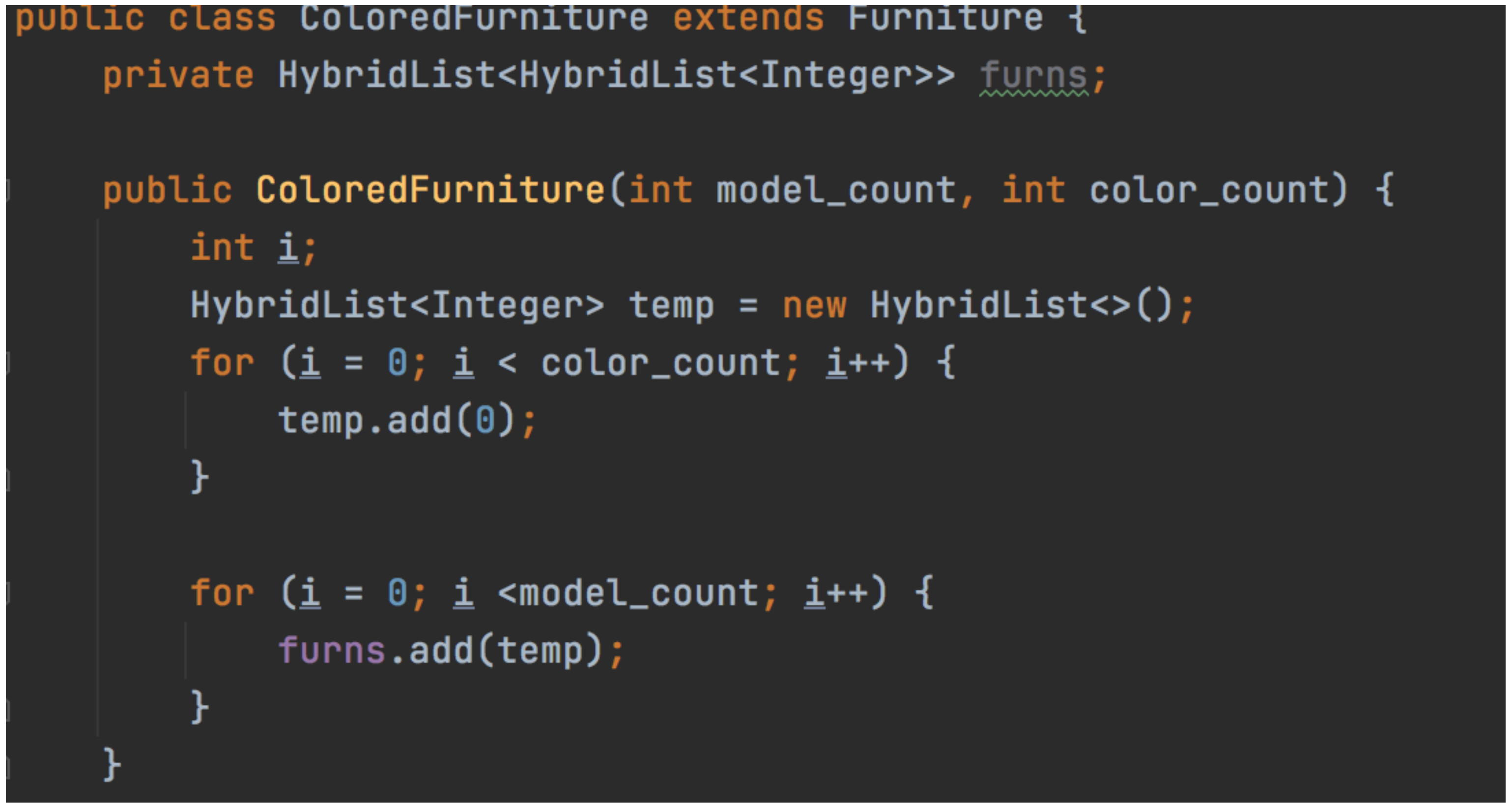
```java
public int show_model(Furniture furniture) {
    int i, choice;
    for (i = 0; i < furniture.getModelCount(); i++) {
        System.out.printf("%d. Model %d, Piece: %d\n", i+1, i+1, furniture.getCount(i))
    }
    return 0;
}
```

10.



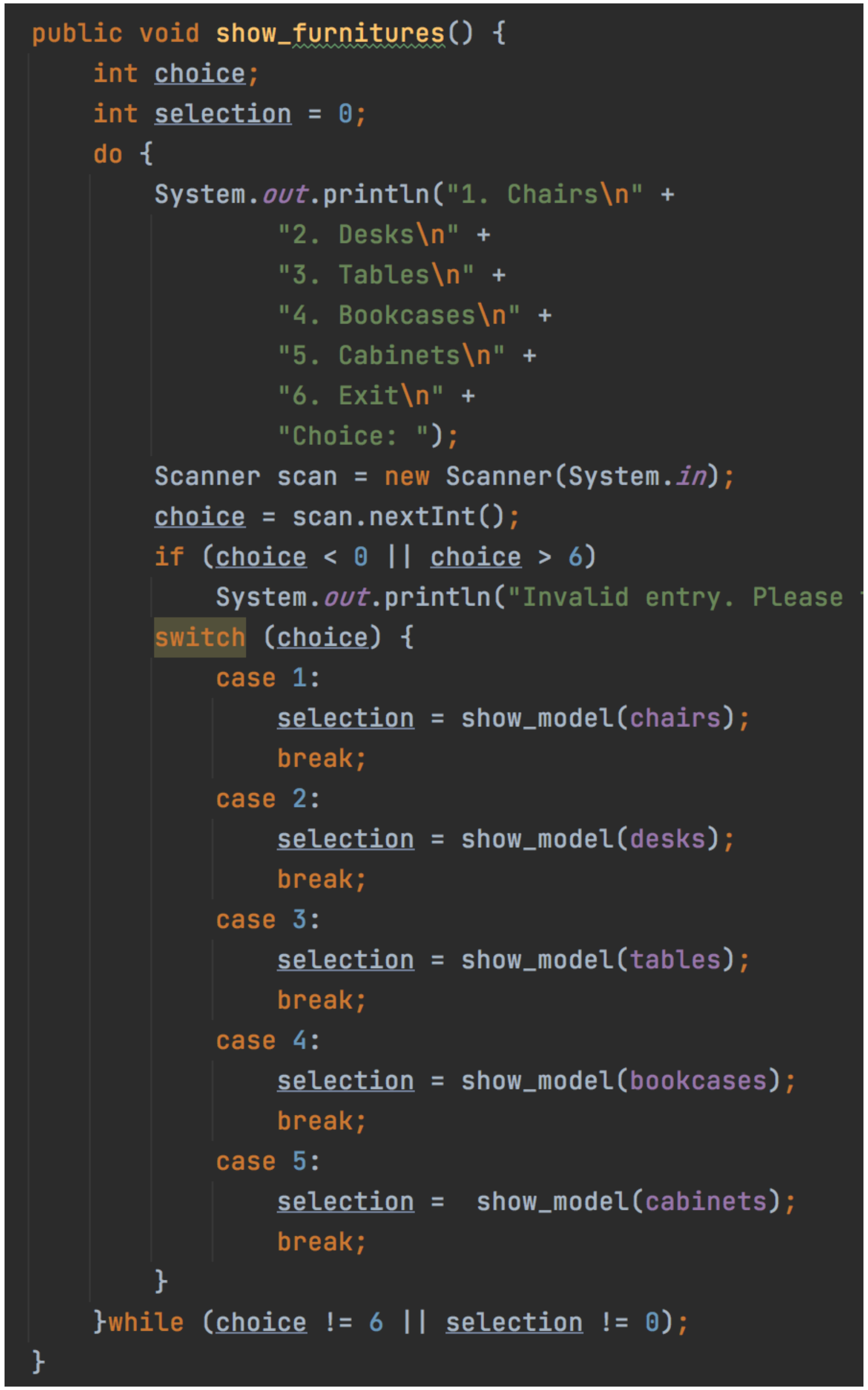
```java
public void print_color(int number) {
    switch (number) {
        case 0: System.out.println("Red "); break;
        case 1: System.out.println("Black "); break;
        case 2: System.out.println("White "); break;
        case 3: System.out.println("Blue "); break;
        case 4: System.out.println("Green "); break;
    }
}
```

11.

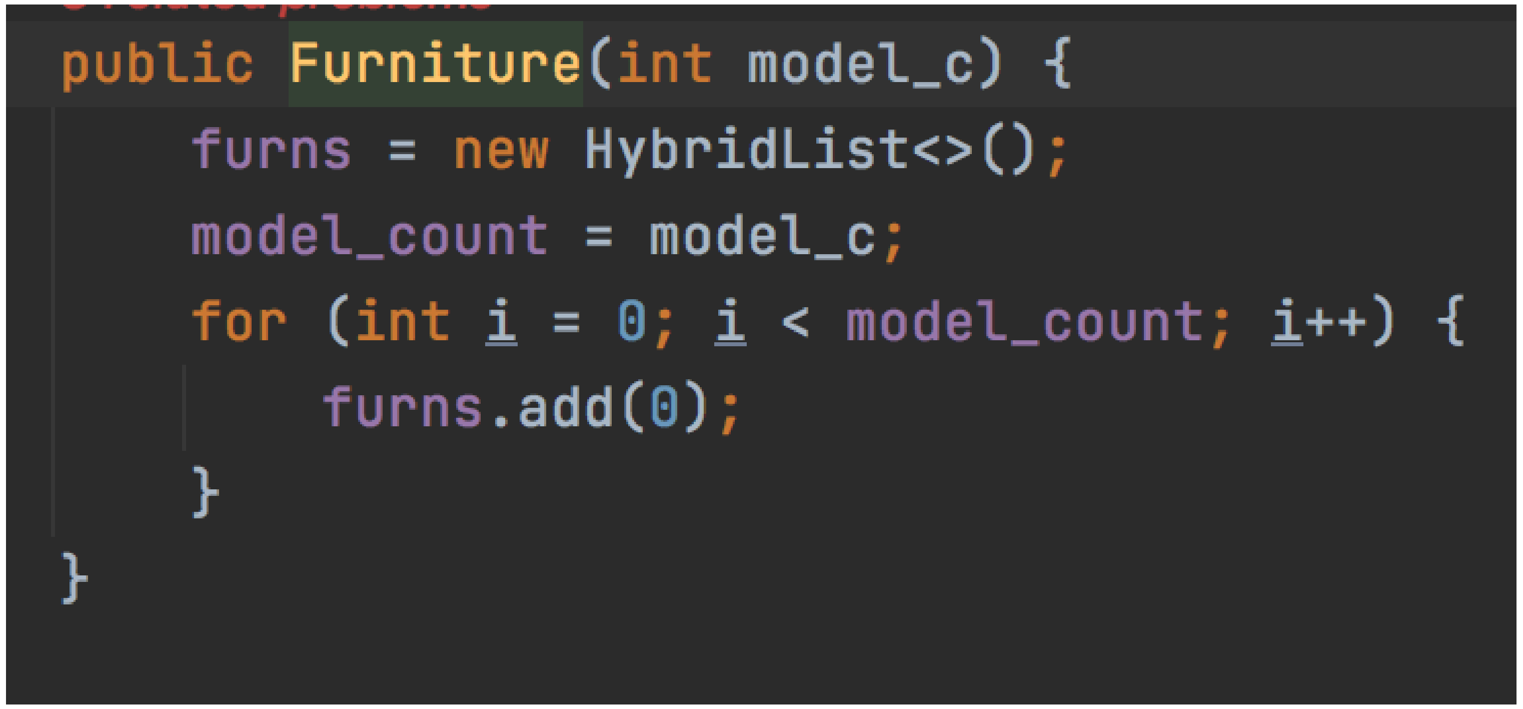


```java
public class ColoredFurniture extends Furniture {
    private HybridList<HybridList<Integer>> furns;

    public ColoredFurniture(int model_count, int color_count) {
        int i;
        HybridList<Integer> temp = new HybridList<>();
        for (i = 0; i < color_count; i++) {
            temp.add(0);
        }

        for (i = 0; i <model_count; i++) {
            furns.add(temp);
        }
    }
```

12.

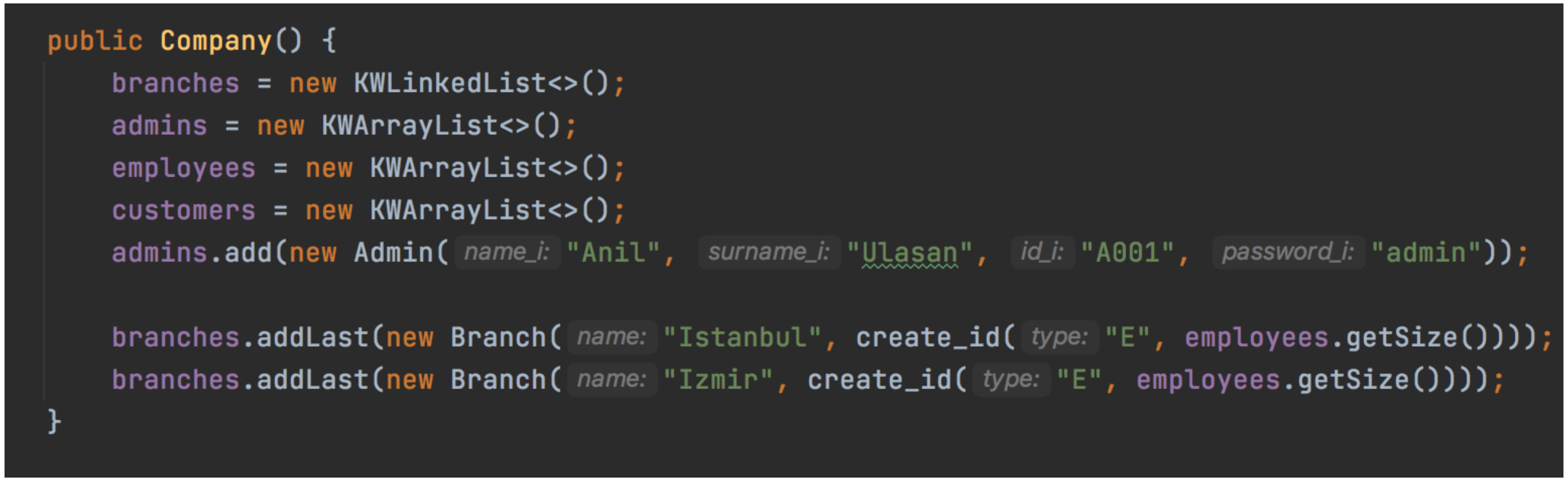

```java
public void show_furnitures() {
    int choice;
    int selection = 0;
    do {
        System.out.println("1. Chairs\n" +
                "2. Desks\n" +
                "3. Tables\n" +
                "4. Bookcases\n" +
                "5. Cabinets\n" +
                "6. Exit\n" +
                "Choice: ");
        Scanner scan = new Scanner(System.in);
        choice = scan.nextInt();
        if (choice < 0 || choice > 6)
            System.out.println("Invalid entry. Please
        switch (choice) {
            case 1:
                selection = show_model(chairs);
                break;
            case 2:
                selection = show_model(desks);
                break;
            case 3:
                selection = show_model(tables);
                break;
            case 4:
                selection = show_model(bookcases);
                break;
            case 5:
                selection =  show_model(cabinets);
                break;
        }
    }while (choice != 6 || selection != 0);
}
```

13.



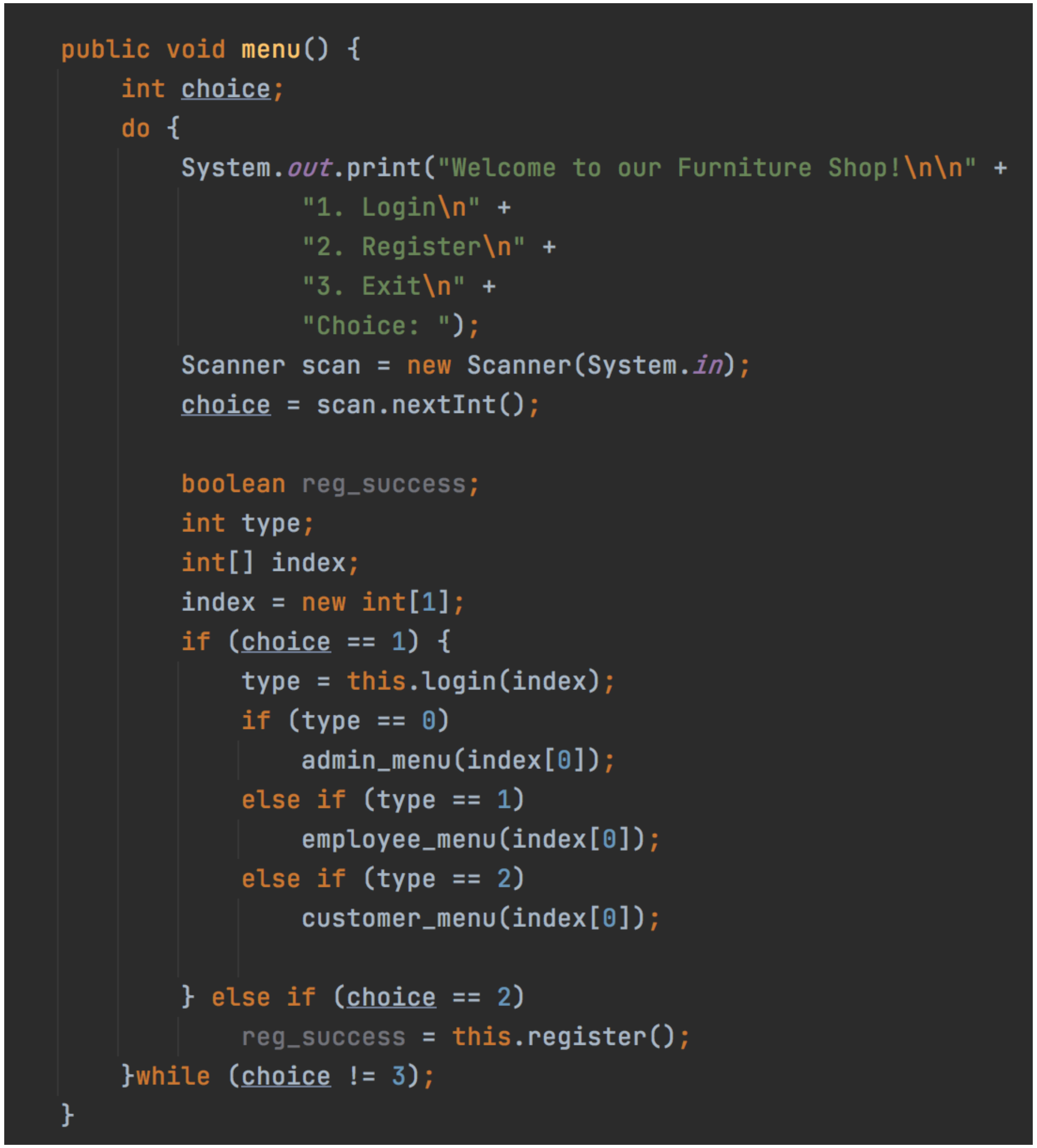
```java
public Furniture(int model_c) {
    furns = new HybridList<>();
    model_count = model_c;
    for (int i = 0; i < model_count; i++) {
        furns.add(0);
    }
}
```

14.



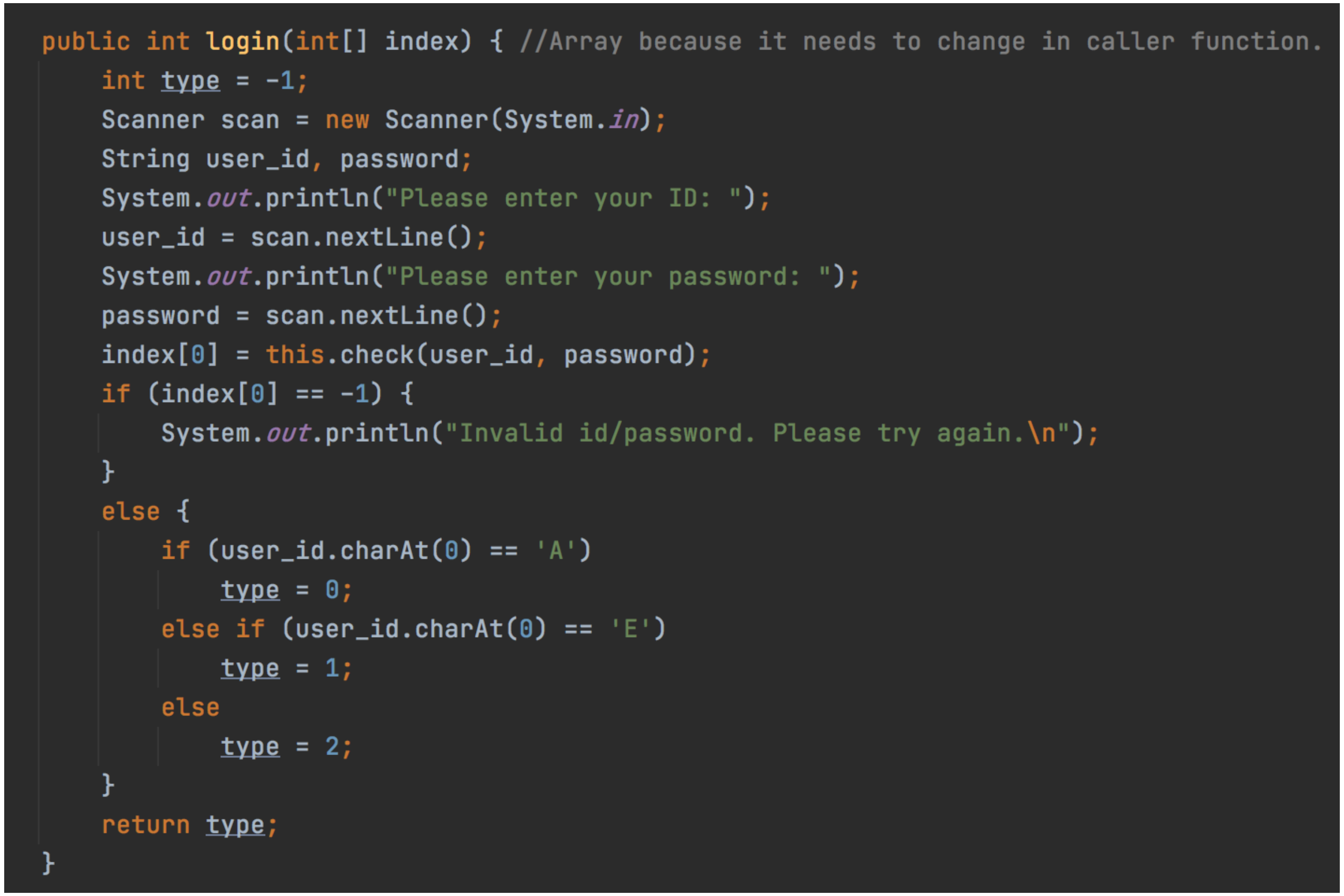
```java
public Company() {
    branches = new KWLinkedList<>();
    admins = new KWArrayList<>();
    employees = new KWArrayList<>();
    customers = new KWArrayList<>();
    admins.add(new Admin( name_i: "Anil",  surname_i: "Ulasan",  id_i: "A001",  password_i: "admin"));

    branches.addLast(new Branch( name: "Istanbul", create_id( type: "E", employees.getSize())));
    branches.addLast(new Branch( name: "Izmir", create_id( type: "E", employees.getSize())));
}
```

15.



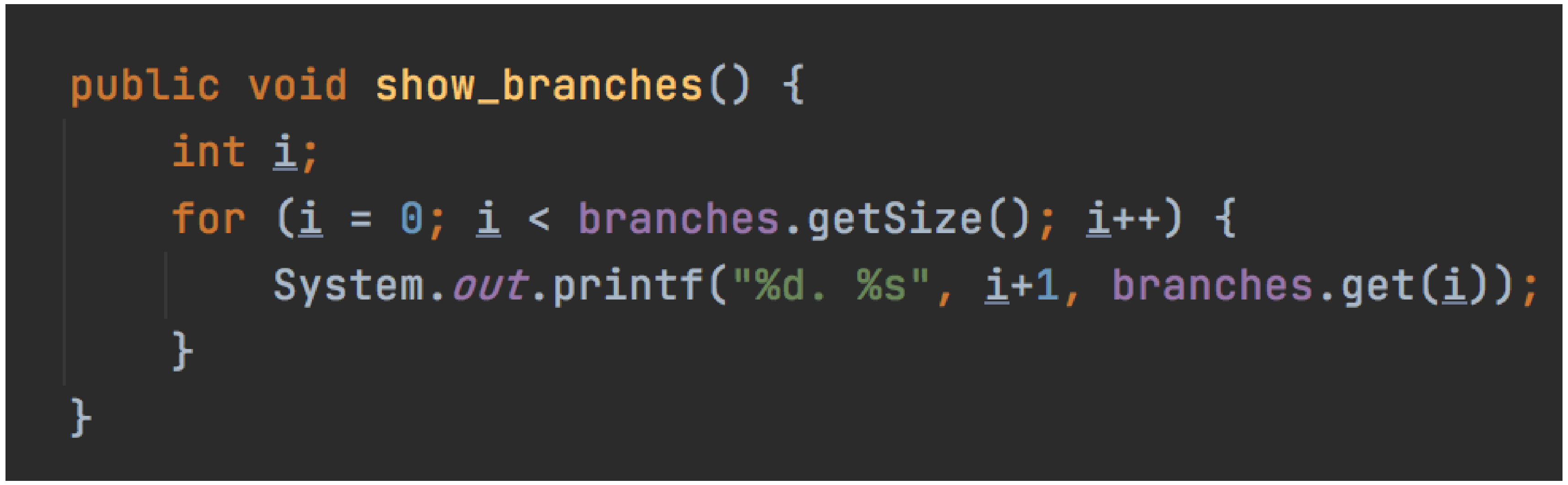
```java
public void menu() {
    int choice;
    do {
        System.out.print("Welcome to our Furniture Shop!\n\n" +
                "1. Login\n" +
                "2. Register\n" +
                "3. Exit\n" +
                "Choice: ");
        Scanner scan = new Scanner(System.in);
        choice = scan.nextInt();

        boolean reg_success;
        int type;
        int[] index;
        index = new int[1];
        if (choice == 1) {
            type = this.login(index);
            if (type == 0)
                admin_menu(index[0]);
            else if (type == 1)
                employee_menu(index[0]);
            else if (type == 2)
                customer_menu(index[0]);

        } else if (choice == 2)
            reg_success = this.register();
    }while (choice != 3);
}
```

16.



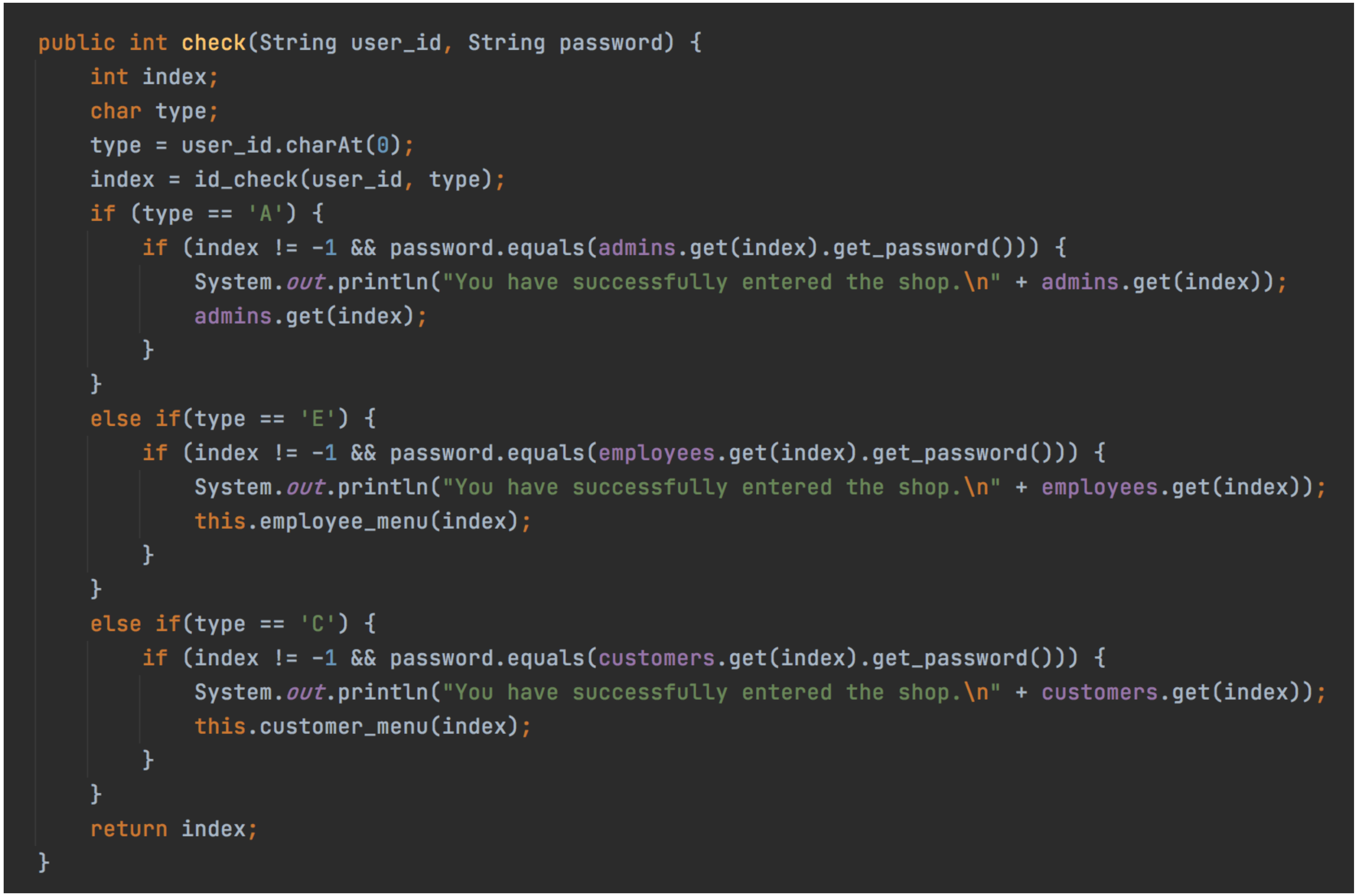
```java
public int login(int[] index) { //Array because it needs to change in caller function.
    int type = -1;
    Scanner scan = new Scanner(System.in);
    String user_id, password;
    System.out.println("Please enter your ID: ");
    user_id = scan.nextLine();
    System.out.println("Please enter your password: ");
    password = scan.nextLine();
    index[0] = this.check(user_id, password);
    if (index[0] == -1) {
        System.out.println("Invalid id/password. Please try again.\n");
    }
    else {
        if (user_id.charAt(0) == 'A')
            type = 0;
        else if (user_id.charAt(0) == 'E')
            type = 1;
        else
            type = 2;
    }
    return type;
}
```

17.



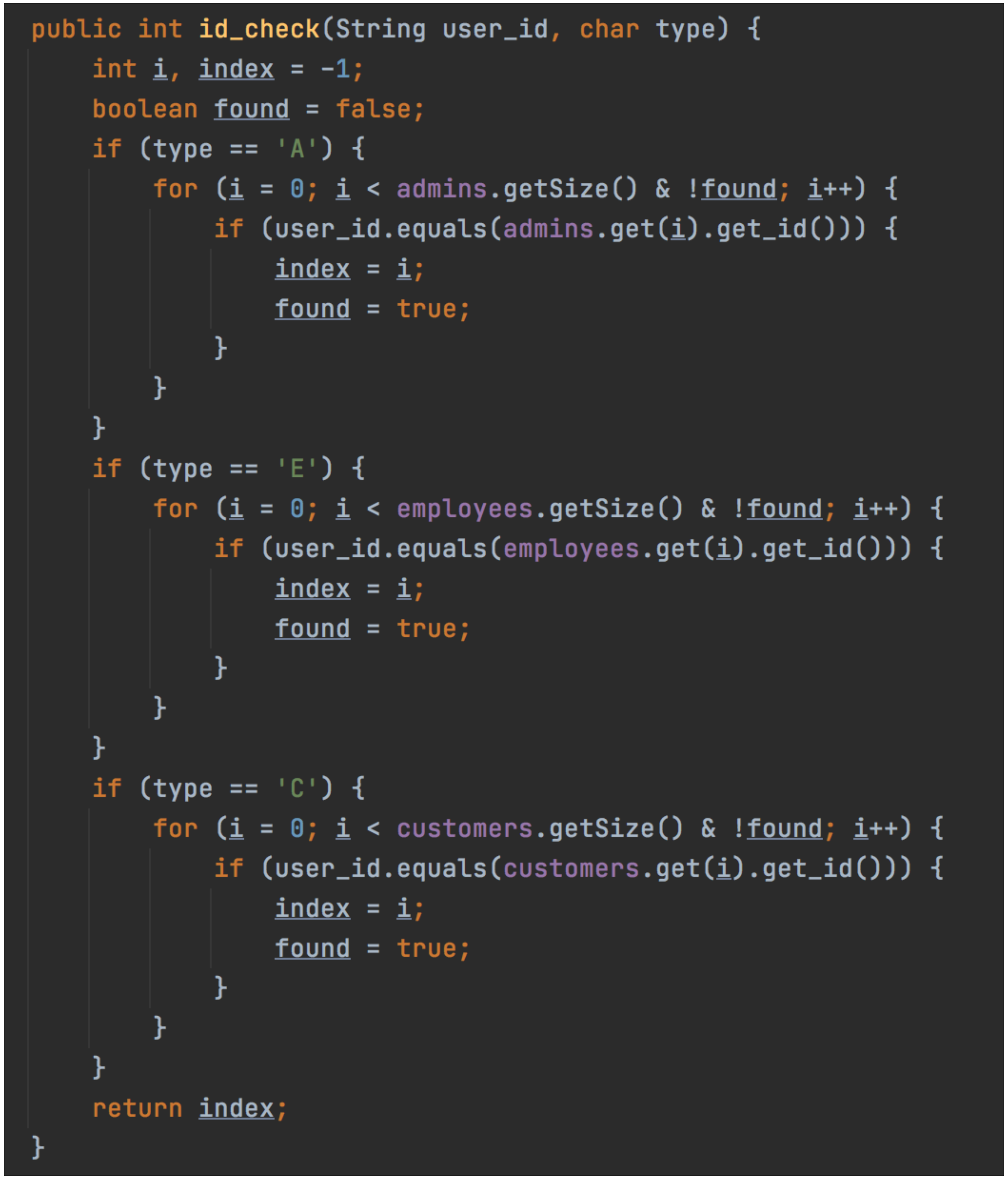
```java
public void show_branches() {
    int i;
    for (i = 0; i < branches.getSize(); i++) {
        System.out.printf("%d. %s", i+1, branches.get(i));
    }
}
```

18.



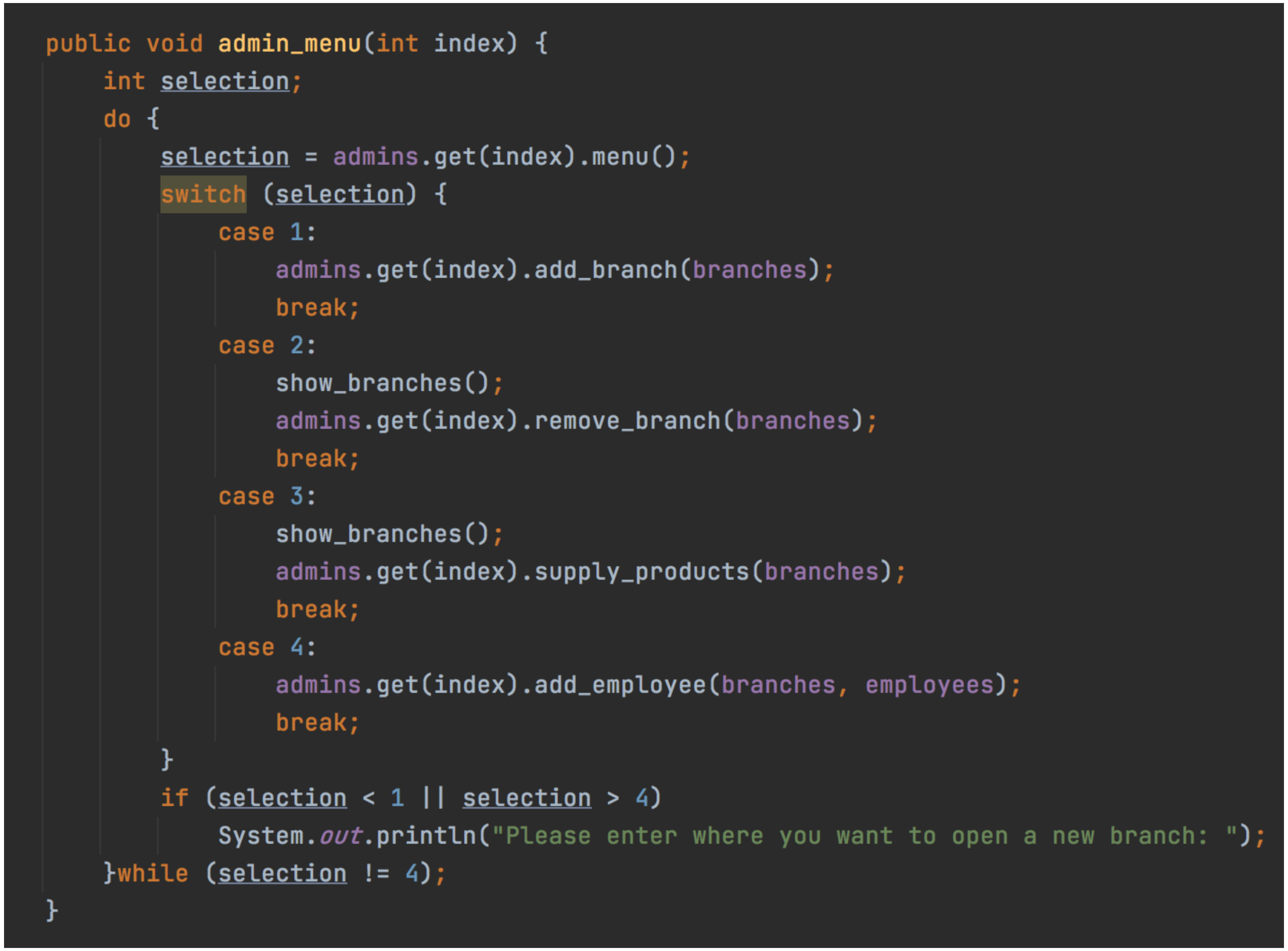
```java
public int check(String user_id, String password) {
    int index;
    char type;
    type = user_id.charAt(0);
    index = id_check(user_id, type);
    if (type == 'A') {
        if (index != -1 && password.equals(admins.get(index).get_password())) {
            System.out.println("You have successfully entered the shop.\n" + admins.get(index));
            admins.get(index);
        }
    }
    else if(type == 'E') {
        if (index != -1 && password.equals(employees.get(index).get_password())) {
            System.out.println("You have successfully entered the shop.\n" + employees.get(index));
            this.employee_menu(index);
        }
    }
    else if(type == 'C') {
        if (index != -1 && password.equals(customers.get(index).get_password())) {
            System.out.println("You have successfully entered the shop.\n" + customers.get(index));
            this.customer_menu(index);
        }
    }
    return index;
}
```

19.



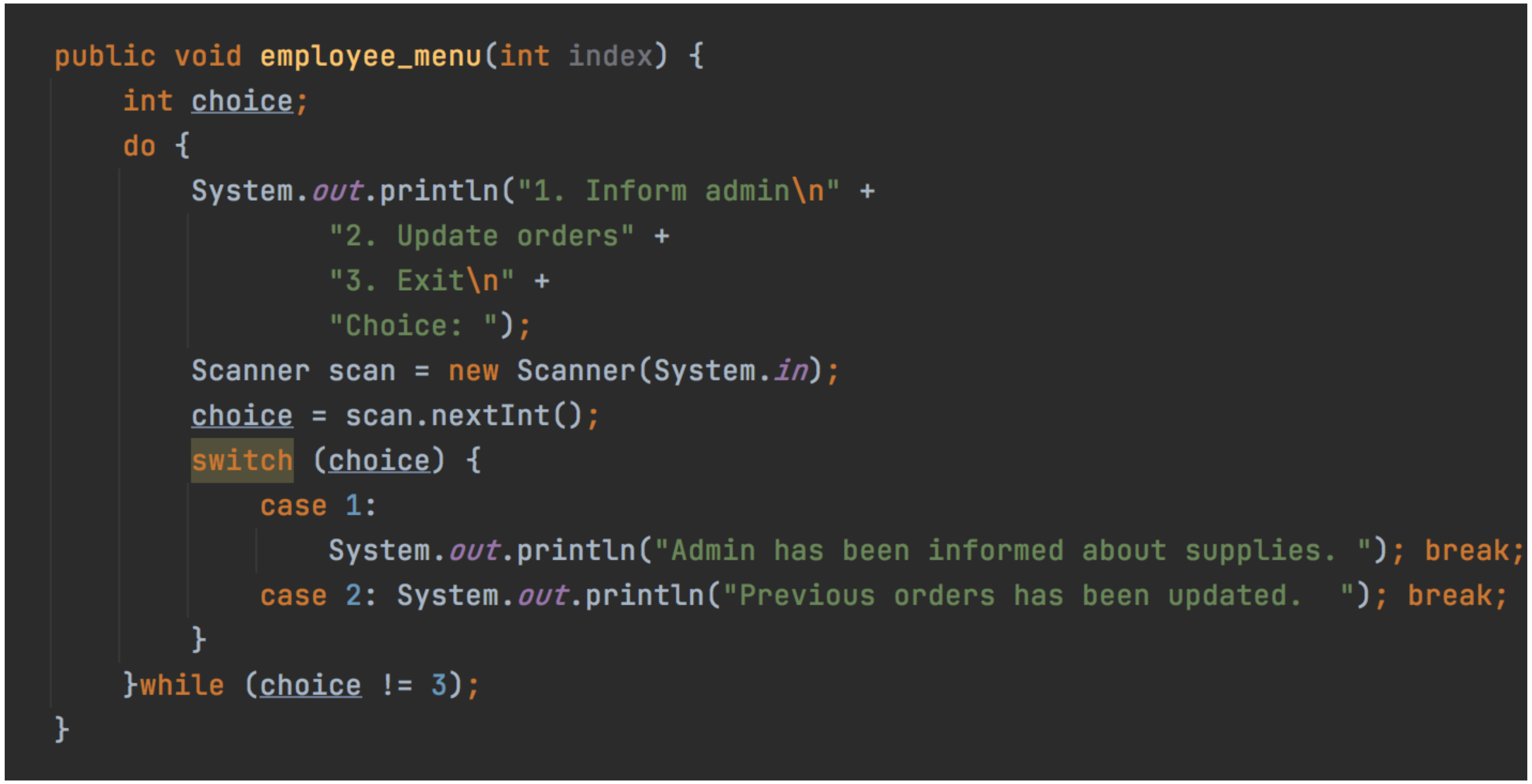
```java
public int id_check(String user_id, char type) {
    int i, index = -1;
    boolean found = false;
    if (type == 'A') {
        for (i = 0; i < admins.getSize() & !found; i++) {
            if (user_id.equals(admins.get(i).get_id())) {
                index = i;
                found = true;
            }
        }
    }
    if (type == 'E') {
        for (i = 0; i < employees.getSize() & !found; i++) {
            if (user_id.equals(employees.get(i).get_id())) {
                index = i;
                found = true;
            }
        }
    }
    if (type == 'C') {
        for (i = 0; i < customers.getSize() & !found; i++) {
            if (user_id.equals(customers.get(i).get_id())) {
                index = i;
                found = true;
            }
        }
    }
    return index;
}
```

20.



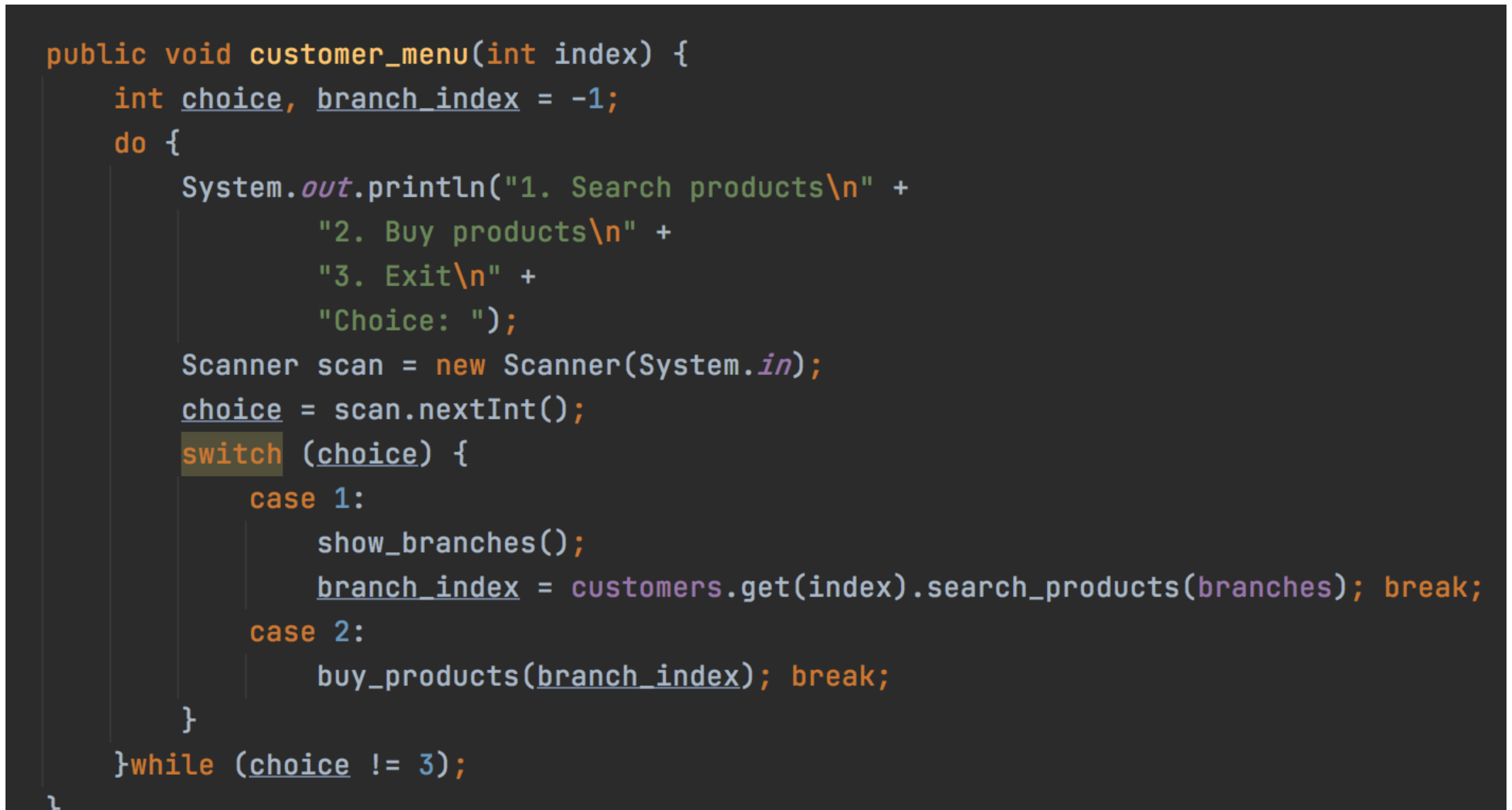
```java
public void admin_menu(int index) {
    int selection;
    do {
        selection = admins.get(index).menu();
        switch (selection) {
            case 1:
                admins.get(index).add_branch(branches);
                break;
            case 2:
                show_branches();
                admins.get(index).remove_branch(branches);
                break;
            case 3:
                show_branches();
                admins.get(index).supply_products(branches);
                break;
            case 4:
                admins.get(index).add_employee(branches, employees);
                break;
        }
        if (selection < 1 || selection > 4)
            System.out.println("Please enter where you want to open a new branch: ");
    }while (selection != 4);
}
```

21.



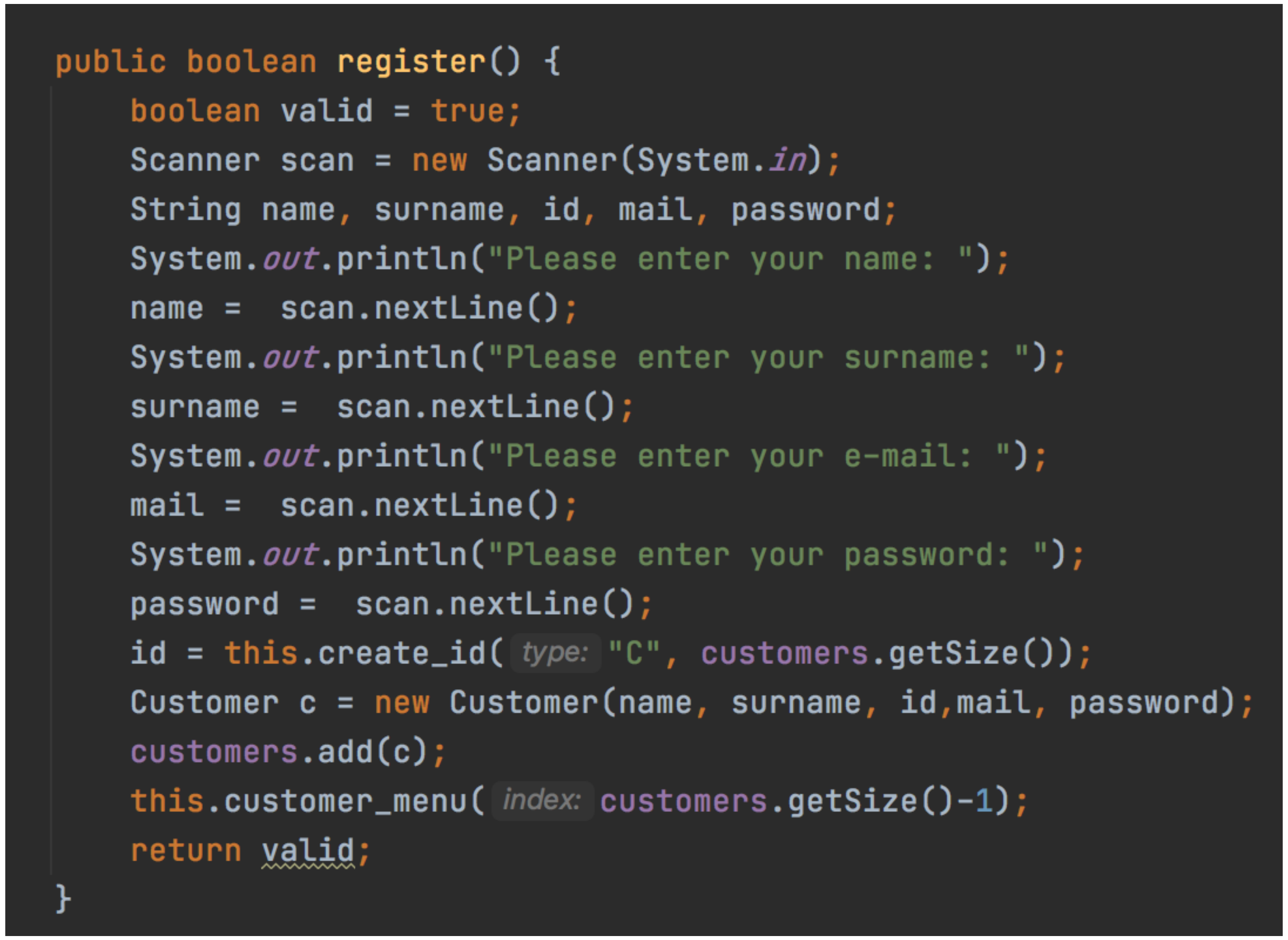
```java
public void employee_menu(int index) {
    int choice;
    do {
        System.out.println("1. Inform admin\n" +
                "2. Update orders" +
                "3. Exit\n" +
                "Choice: ");
        Scanner scan = new Scanner(System.in);
        choice = scan.nextInt();
        switch (choice) {
            case 1:
                System.out.println("Admin has been informed about supplies. "); break;
            case 2: System.out.println("Previous orders has been updated.  "); break;
        }
    }while (choice != 3);
}
```

22.



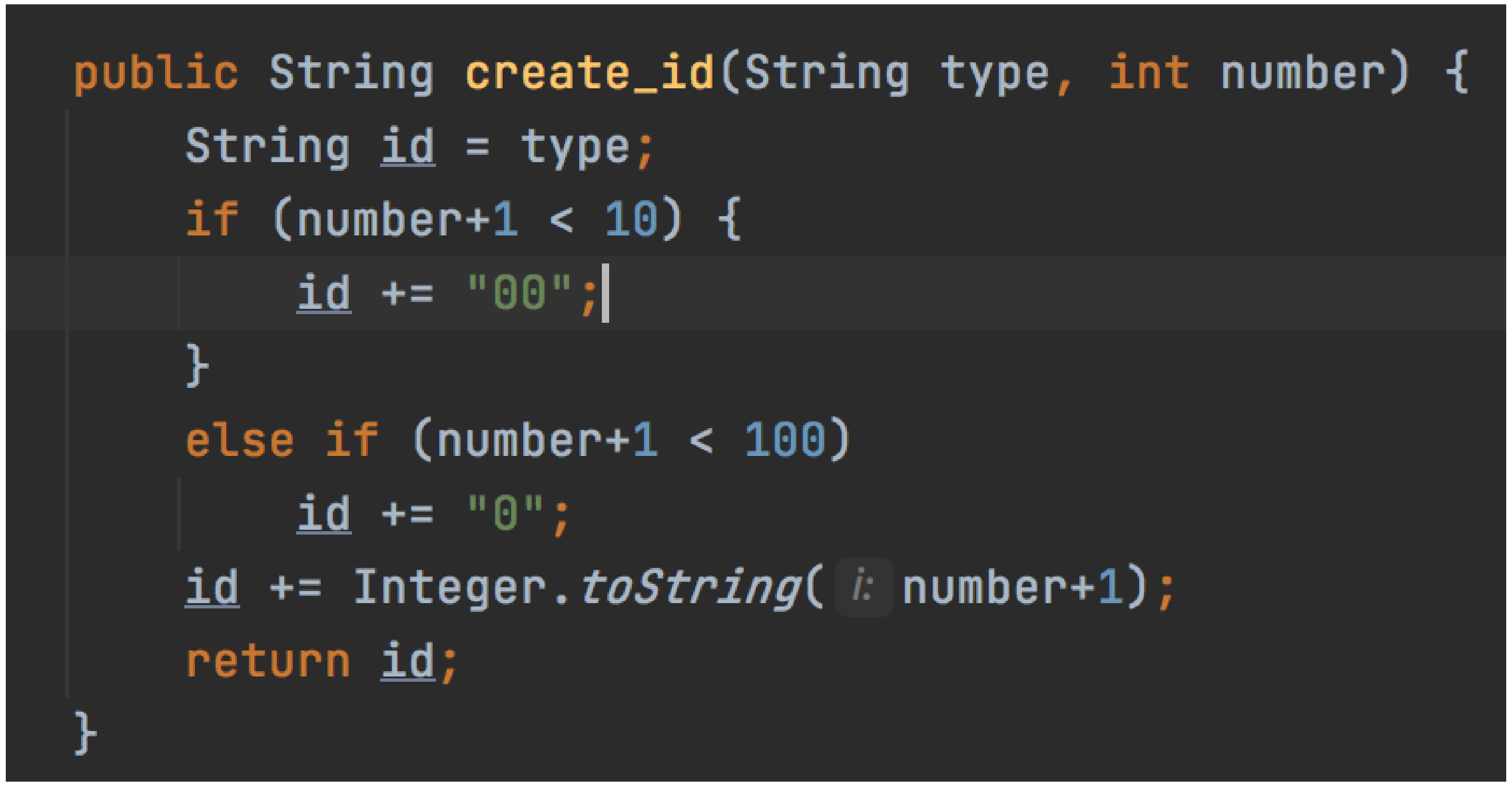
```java
public void customer_menu(int index) {
    int choice, branch_index = -1;
    do {
        System.out.println("1. Search products\n" +
                "2. Buy products\n" +
                "3. Exit\n" +
                "Choice: ");
        Scanner scan = new Scanner(System.in);
        choice = scan.nextInt();
        switch (choice) {
            case 1:
                show_branches();
                branch_index = customers.get(index).search_products(branches); break;
            case 2:
                buy_products(branch_index); break;
        }
    }while (choice != 3);
}
```

23.



```java
public boolean register() {
    boolean valid = true;
    Scanner scan = new Scanner(System.in);
    String name, surname, id, mail, password;
    System.out.println("Please enter your name: ");
    name =  scan.nextLine();
    System.out.println("Please enter your surname: ");
    surname =  scan.nextLine();
    System.out.println("Please enter your e-mail: ");
    mail =  scan.nextLine();
    System.out.println("Please enter your password: ");
    password =  scan.nextLine();
    id = this.create_id( type: "C", customers.getSize());
    Customer c = new Customer(name, surname, id,mail, password);
    customers.add(c);
    this.customer_menu( index: customers.getSize()-1);
    return valid;
}
```

24.



```java
public String create_id(String type, int number) {
    String id = type;
    if (number+1 < 10) {
        id += "00";
    }
    else if (number+1 < 100)
        id += "0";
    id += Integer.toString( i: number+1);
    return id;
}
```