ADMIN TIME COMPLEXITY

```java
public void see_all_product() {
    System.out.println(all_branchs);
}

public boolean product_is_in(int branch_id, String furniture, char model_id, int color_id) {

    return all_branchs.get(branch_id).data.is_in(furniture, model_id, color_id);
}
public void add_branch(String branch_name){
    try{
        all_branchs.add(new branch(branch_name), index: 0);
    }
    catch (Exception e){
        System.out.println(e);
    }
}


public void delete_branch(int index){
    try{
        all_branchs.remove(index);
    }
    catch (Exception e){
        System.out.println(e);
    }
}
```

O(n.m)

T(N)=O(n.m)

O(m^2.n)

T(N)=O(m^2.n)

O(1)

O(1)

T(N)=O(1)+O(1)=O(1)

O(n)

T(N)=O(n)

```java
public void add_employee(int branch_id,String employee_name){
    try{
        all_branchs.get(branch_id).employee_name.add( index: 0,employee_name);
    }
    catch (Exception e){
        System.out.println(e);
    }
}


public boolean delete_employee(int branch_id,String employee_name){
    int index=all_branchs.get(branch_id).employee_name.contains(employee_name);
    if(index!=-1){
        all_branchs.get(branch_id).employee_name.remove(index);
       return true;
    }
    return false;
}

public void print_employee(int branch_id){
    System.out.println(all_branchs.get(branch_id).employee_name);
}

public void show_branchs_name(){
    System.out.println("\t***BRANCHS NAME**");
    for(int i=0;i<all_branchs.getSize();i++){
        System.out.println((i+1)+")"+all_branchs.get(i).branch_name+"\n");
    }
}
```

T(N)=O(n.m) +O(1)=O(n.m)

T(N)=O(n.m.k)+O(n.m)=O(n.m.k)

T(N)=O(n).O(n.m)=O(n^2.m)

COSTUMER TIME COMPLEXITY

```java
public void see_all_product() {
    System.out.println(all_branchs);
}

public boolean product_is_in(int branch_id, String furniture, char model_id, int color_id) {

    return all_branchs.get(branch_id).data.is_in(furniture, model_id, color_id);
}
```

O(n.m)

T(N)=O(n.m)

O(m^2.n)

T(N)=O(m^2.n)

```java
public void login_system(String name, String surname, String email, String password) {
    if (costumer_data == null) {
        costumer_data=new ArrayList_ahmet<costumer_information>(); // O(1)
        try{
            System.out.println("asda");
            email_=email; // O(1)
            password_=password; // O(1)
            costumer_data.add( index 0, new costumer_information()); // O(1)
            costumer_data.get(0).name = name; // O(1)
            costumer_data.get(0).surname = surname; // O(1)
            costumer_data.get(0).email = email; // O(1)
            costumer_data.get(0).password = password; // O(1)
            costumer_data.get(0).id = costumer_data.getSize()-1; // O(1)
            System.out.printf("NAME=%s\nSURNAME=%s\nEMAIL=%s\nPASSWORD=%s", costumer_data.get(0).name, costumer_data.get(0).surname, // O(1)
                                                                              costumer_data.get(0).email, costumer_data.get(0).password);
            System.out.println("REGISTERED\n"); // O(1)
            return;
        }
        catch (Exception e){
            System.out.println(e); // O(1)
        }

    }
    try{
        email_ = new String(email); // O(1)
        password_ = new String(password); // O(1)
        int index=costumer_data.getSize(); // O(1)
        System.out.println("index="+index); // O(1)
        costumer_data.add(index, new costumer_information()); // O(1)
        costumer_data.get(index).name = name; // O(1)
        costumer_data.get(index).surname = surname; // O(1)
        costumer_data.get(index).email = email; // O(1)
        costumer_data.get(index).password = password; // O(1)
        costumer_data.get(index).id = index; // O(1)
        System.out.printf("NAME=%s\nSURNAME=%s\nEMAIL=%s\nPASSWORD=%s", costumer_data.get(index).name, costumer_data.get(index).surname, // O(1)
                                                                          costumer_data.get(index).email, costumer_data.get(index).password);
        System.out.println("REGISTERED\n"); // O(1)
    }
    catch (Exception e){
        System.out.println(e); // O(1)
    }
}
```

O(1)

O(1)

NOTE:Costumer_data type is arrayList so all get O(1)

T(N)=O(1)

```
public boolean login_system(String email, String password) {
    email_ = new String(email);                                   O(1)
    password_ = new String(password);                             O(1)
    if (costumer_data == null) {                                  O(1)
        System.out.println("PLEASE REGISTER");                    O(1)
        return false;                                             O(1)
    }                                                             O(1)
    for (int i = 0; i < costumer_data.getSize(); i++) {           getSize()=O(m)
        if (costumer_data.get(i).email.equals(email_) && costumer_data.get(i).password.equals(password_)) {   Equals=O(n)
            System.out.println("LOG IN");                         O(1)
            return true;                                          O(1)
        }                                                         O(n.m)
    }
    System.out.println("PLEASE REGISTER");                        O(1)
    return false;                                                 O(1)
}                                                                 T(N)=O(n.m)

public void buy_product_online(int branch_number, String which_furniture, int color, char model_id, String addres, String phone_number) {
    boolean flag = false;
    int costumer_id;
    for (costumer_id = 0; costumer_id < costumer_data.getSize(); costumer_id++) {   getSize()=O(m)
        if (costumer_data.get(costumer_id).email.equals(email_) && costumer_data.get(costumer_id).password.equals(password_)) {   Equals=O(n)
            flag = true;                                          O(1)

            break;                                                O(1)
        }                                                         O(n.m)

    }
    if (flag == true) {
        if(all_branchs.get(branch_number).data.remove_item(which_furniture, color, model_id) == true) {   all_branchs.get()=O(k)  data_remove_item()=O(n.x)   O(k+n.x)
            costumer_data.get(costumer_id).prev_view.add( index: 0, entry: "\nFURNITURE= "+which_furniture+"\t\tMODEL= "+model_id+"\tCOLOR= "+color_prev(color));   O(t)
            costumer_data.get(costumer_id).address=addres;        O(1)
            costumer_data.get(costumer_id).address=phone_number;  O(1)

        }                                                         O(k+nx+t)
    }                                                             T(N)=O(n.m+k+nx+t)
}
```

```java
public void show_prev_views(){
  int i;
    for(i=0;i<costumer_data.getSize();i++){
    if(costumer_data.get(i).email.equals(email_)&&costumer_data.get(i).password.equals(password_)){
        break;
    }
}
if(costumer_data.get(i).prev_view!=null){
    System.out.println(costumer_data.get(i).name+" "+" "+costumer_data.get(i).surname);
    System.out.println(costumer_data.get(i).prev_view);
}

else
    System.out.println("NOT INFORMATION");
}
```

getSize=n

Equals()=O(m)

O(1)

O(1)

O(1)

O(1)

O(n).O(m)=O(n.m)

T(N)=O(M.N)

EMPLOYEE TIME COMPLEXITY

```java
public void see_all_product() {
    System.out.println(all_branchs);
}

public boolean product_is_in(int branch_id, String furniture, char model_id, int color_id) {

    return all_branchs.get(branch_id).data.is_in(furniture, model_id, color_id);
}
public void add_product(int branch_id,String WhichFurniture,int color_id,char model_id){

    try {
        if (WhichFurniture.equals("DESK")) {
            all_branchs.get(branch_id).data.add_new_item(new desks(color_id, model_entry: model_id-65));
        }
        else if (WhichFurniture.equals("CHAIR")) {
            all_branchs.get(branch_id).data.add_new_item(new chairs(color_id, model_entry: model_id-65));
        }
        else if (WhichFurniture.equals("MEETINGTABLE")) {
            all_branchs.get(branch_id).data.add_new_item(new meeting_tables(color_id, model_entry: model_id-65));
        }
        else if (WhichFurniture.equals("CABINET")) {
            all_branchs.get(branch_id).data.add_new_item(new cabinet( model_entry: model_id-65));
        }
        else if (WhichFurniture.equals("BOOKCASE")) {
            all_branchs.get(branch_id).data.add_new_item(new bookcase( model_entry: model_id-65));
        }

    }
    catch (Exception e){
        System.out.println(e);
    }
}
public void remove_product(int branch_id,String furniture,int color_id,char model_id){
    all_branchs.get(branch_id).data.remove_item(furniture,color_id,model_id);
}
```

O(n.m)

T(N)=O(n.m)

O(m^2.n)

T(N)=O(m^2.n)

O(n.m)

data.add_new_item=O(nm)

O(n.m)

O(n.m)

O(n.m)

T(N)=O(n.m)

O(1)

O(n.m)

T(N)=O(n.m)

```java
public void buy_product_store(costumer costumer_,int id,int branch_id,String which_furniture,char model_id,int color_id)throws ahmets_exceptions{
    if(costumer_.costumer_data.getSize()>id){
        remove_product(branch_id,which_furniture,color_id,model_id);
        costumer_.costumer_data.get(id).prev_view.add( index: 0, entry: "\nFURNITURE= "+which_furniture+"\t\tMODEL= "+model_id+"\tCOLOR= "+color_prev(color_id));
    }
    else{
        System.out.println("INCORRECT ID PLEASE TRY AGAIN\n");
    }

}
```

O(1)

O(n.m)

O(k)

O(1)

T(N)=O(nm+k)