**WEEK8**

**public** **class** my{

**public** **static** **void** main(**String**[] args) {

**Circle** a = new Circle();

        System.out.println(a);

        a.Calculate\_Area(1);

        a.Print\_Area();

        a.Display\_Area();

**Square** s = new Square();

        System.out.println(s);

        s.Calculate\_Area(5);

        s.Print\_Area();

        s.Display\_Area();

**Rectangle** r = new Rectangle();

        System.out.println(r);

        r.Calculate\_Area(5, 10);

        r.Print\_Area();

        r.Display\_Area();

    }

}

**class** Circle **extends** Shape{

**private** **double** radius;

**private** **double** Area;

    Circle(){

        super();

    }

**public** **String** toString(){

        return "This is a Circle";

    }

**public** **void** Display\_Area(){

        System.out.println(this.Area);

    }

**void** Print\_Area(){

        System.out.println(this.Area);

    }

**public** **double** Calculate\_Area(**double** radius){

        Area = radius \* radius \* Math.PI;

        return Area;

    }

}

**class** Shape **extends** Object{

**private** **double** Area;

    Shape(){

        Area = 0;

    }

**public** **String** toString(){

        return "This is a Shape";

    }

**double** Calculate\_Area(){

        return Area;

    }

**public** **void** Display\_Area(){

        System.out.println(this.Area);

    }

**void** Print\_Area(){

        System.out.println(this.Area);

    }

}

**class** Square **extends** Shape{

**private** **double** Length;

**private** **double** Area;

    Square(){

        super();

    }

**public** **String** toString(){

        return "This is a Square";

    }

**public** **void** Display\_Area(){

        System.out.println(this.Area);

    }

**void** Print\_Area(){

        System.out.println(this.Area);

    }

**public** **double** Calculate\_Area(**double** Length){

        Area = Length \* Length;

        return Area;

    }

}

**class** Rectangle **extends** Shape{

**private** **double** Length, Breadth;

**private** **double** Area;

    Rectangle(){

        super();

    }

**public** **String** toString(){

        return "This is a Rectangle";

    }

**public** **void** Display\_Area(){

        System.out.println(this.Area);

    }

**void** Print\_Area(){

        System.out.println(this.Area);

    }

**public** **double** Calculate\_Area(**double** Length, **double** Breadth){

        Area = Length \* Breadth;

        return Area;

    }

}

Изображение выглядит как текст

Автоматически созданное описание

**WEEK5**

import java**.**util**.**Scanner;

import java**.**util**.**List;

import java**.**util**.**ArrayList;

**public** **class** Main{

**static** **boolean** isExist(**List** list, **Integer** x){

        for (**int** i = 0; i < list.size(); i++) {

            if(list.get(i) == x){

                return true;

            }

        }

        return false;

    }

**public** **static** **void** main(**String**[] args) {

**Scanner** sc = new Scanner(System.in);

**List** <**Integer**> ar = new **ArrayList**<>();

        while(sc.hasNext()){

**int** a = sc.nextInt();

            if(a == 0) break;

            else ar.add(a);

        }

**int** x = sc.nextInt();

        if(isExist (ar, x)){

            System.out.println("Found!");

        }

        else{

            System.out.println("Not found!");

        }

    }

}

Изображение выглядит как текст

Автоматически созданное описание

**WEEK12**

import java.util.\*;

public class Task {

public static class FavoriteClasses {

private favorite1;

private favorite2;

private favorite3;

FavoriteClasses(fav1, fav2, fav3){

this.favorite1 = fav1;

this.favorite2 = fav2;

this.favorite3 = fav3;

}

public void getFav1(){

return this.favorite1;

}

public void getFav2(){

return this.favorite2;

}

public void getFav3(){

return this.favorite3;

}

}

public static void main(String[] args){

List r=new ArrayList();

r.add(3.5);

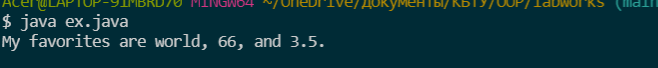
r.add(7.2);

FavoriteClasses a = new FavoriteClasses("world", 66, r.get(0));

System.out.println("My favorites are " + a.getFav1() + ", " + a.getFav2() + ", and " + a.getFav3() + ".");

}

}

****

**WEEK9**

package homework;

import java**.**io**.**IOException;

import java**.**io**.**FileWriter;

**public** **class** hw9{

**public** **static** **void** main(**String**[] args) {

**Users** u = new Users("Nursultan", "123123123123", "66666666666");

        u.er();

    }

}

**class** Users **extends** Errors

{

**private** **String** name;

**private** **String** IIN;

**private** **String** phone;

    Users(**String** n, **String** i, **String** p){

        name = n;

        IIN= i;

        phone = p;

    }

**public** **void** er(){

        if(IIN.length() != 12 || phone.length() != 11){

            try {

**FileWriter** myWriter = new FileWriter("filename.txt");

                myWriter.write(IIN + " " + phone);

                myWriter.close();

            } catch (**IOException** e) {

                System.out.println("ERROR");

                e.printStackTrace();

            }

            throw new ArithmeticException("IIN or phone are wrong");

        }

        else{

            System.out.println("Good job!");

        }

    }

}

**abstract** **class** Errors{

**public** **abstract** **void** er();

}

**Изображение выглядит как текст

Автоматически созданное описание**

**WEEK14**

import java**.**util**.**Scanner;

import java**.**util**.**ArrayList;

**public** **class** Week14 {

**static** **void** Task1(**String** arr[], **String** s){

**int** cnt = 0;

        for(**String** i: arr){

            if(i.equals(s)) {

                cnt++;

            }

        }

        System.out.println("word \"" + s + "\" occurs in array " + cnt + " times");

    }

**static** **boolean** Task2(**ArrayList**<**Integer**> arr, **int** a){

        for(**int** i: arr){

            if (a == i && a!=0) {

                return true;

            }

        }

        return false;

    }

**public** **static** **void** main(**String** args[]){

**Scanner** scan = new Scanner(System.in);

*// 1*

        System.out.println("TASK 1");

        System.out.println("Enter string:");

**String** words[] = {"uchiha", "itachi", "naruto", "sakura", "loves", "sasuke", "anime", "hate", "boruto"};

**String** word = scan.nextLine();

        Task1(words, word);

*// 2*

**int** n = 5;

**ArrayList**<**Integer**> a = new **ArrayList**<**Integer**>();

        System.out.println("TASK 2");

        System.out.println("Enter integers:");

        while(n!=0){

**Integer** num = scan.nextInt();

            a.add(num);

            n -= 1;

        }

        System.out.println("Enter target:");

**int** t;

        t = scan.nextInt();

        scan.close();

        if(Task2(a, t)){

            System.out.println("found");

        }else{

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

        }

    }

}

Изображение выглядит как текст

Автоматически созданное описание

**WEEK7**

**public** **class** week7 {

**public** **static** **void** main(**String**[] args) {

**Client** firstClient = new Client("Nurs", "8-(777)-777-77-77", "1");

        System.out.println(firstClient.getName());

        Client.setName(firstClient, "Nurislam");

        System.out.println(firstClient.getName());

        System.out.println(firstClient.getPhone());

        Client.setPhone(firstClient, "8-(777)-666-66-55");

        System.out.println(firstClient.getPhone());

        System.out.println(firstClient.getId());

**BankAccount** firstBankAccount = new BankAccount("6666 6666 6666 6666", "active", firstClient);

        System.out.println(firstBankAccount.getStatus());

        BankAccount.setStatus(firstBankAccount, "deleted");

        System.out.println(firstBankAccount.getStatus());

        System.out.println(firstBankAccount.getNumber());

        System.out.println(firstBankAccount.getClient().getName());

**Bank** firstBank = new Bank("Halyk Bank");

        System.out.println(firstBank.getName());

        Bank.setName(firstBank, "Forte Bank");

        System.out.println(firstBank.getName());

        System.out.println(firstBank.countAccount());

        firstBank.createBankAccount(firstBankAccount);

        System.out.println(firstBank.countAccount());

        firstBank.createClient(firstClient);

        System.out.println(firstBank.countClients());

        System.out.println(firstBank.makeReport());

    }

}

**class** Client {

**private** **String** name;

**private** **String** phone;

**public** **String** id;

    Client(**String** name, **String** phone, **String** id) {

        this.name = name;

        this.phone = phone;

        this.id = id;

    }

**static** **void** setName(**Client** c, **String** newName) {

        c.name = newName;

    }

**public** **String** getName() {

        return "Client: " + this.name;

    }

**public** **String** getId() {

        return "ID: " + this.id;

    }

**static** **void** setPhone(**Client** c, **String** newPhone) {

        c.phone = newPhone;

    }

**public** **String** getPhone() {

        return "Phone: " + this.phone;

    }

}

**class** BankAccount {

**private** **String** number;

**private** **String** status;

**public** **Client** owner;

    BankAccount(**String** number, **String** status, **Client** owner) {

        this.number = number;

        this.status = status;

        this.owner = owner;

    }

**static** **void** setStatus(**BankAccount** bankAccount, **String** newStatus) {

        bankAccount.status = newStatus;

    }

**public** **String** getStatus() {

        return "Status: " + this.status;

    }

**public** **String** getNumber() {

        return "Number: " + this.number;

    }

**public** **Client** getClient() {

        return this.owner;

    }

}

**class** Bank {

**private** **Client**[] clients;

**private** **BankAccount**[] accounts;

**private** **int** accountsCount;

**private** **int** clientsCount;

**public** **String** name;

    Bank(**String** name) {

        this.name = name;

        this.clients = new **Client**[100];

        this.accounts = new **BankAccount**[100];

    }

**static** **void** setName(**Bank** b, **String** newName) {

        b.name = newName;

    }

**public** **String** getName() {

        return "Bank: " + this.name;

    }

**public** **void** createBankAccount(**BankAccount** bankAccount) {

        if (this.accountsCount < 100 ) {

            this.accounts[accountsCount] = bankAccount;

            accountsCount++;

        }

    }

**public** **void** createClient(**Client** theClient) {

        if (this.clientsCount < 100 ) {

            this.clients[clientsCount] = theClient;

            clientsCount++;

        }

    }

**public** **String** makeReport(){

        return this.clients[this.clientsCount - 1].getId() + " " + this.clients[this.clientsCount - 1].getName() + " " + this.accounts[this.accountsCount - 1].getStatus();

    }

**public** **String** countAccount() {

        return "Count Account: " + this.accountsCount;

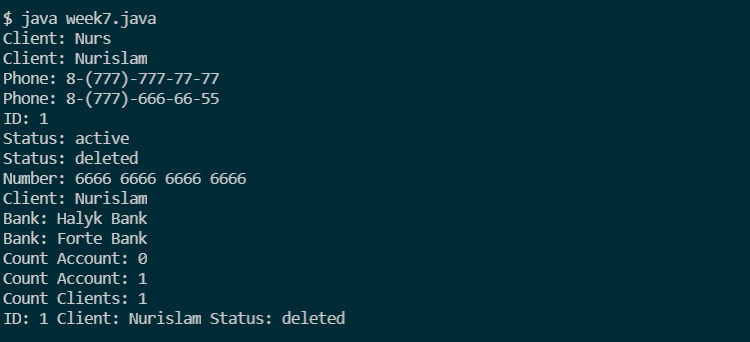
    }

**public** **String** countClients(){

        return "Count Clients: " + this.clientsCount;

    }

}



**HASHMAP**

import java**.**util**.\***;

import java**.**util**.**HashMap;

import java**.**util**.**Scanner;

**public** **class** a{

**public** **static** **void** main(**String** args[]){

**HashMap**<**Integer** , **String**> lab9 = new **HashMap**<>();

        lab9.put(1,"ggorgg");

        lab9.put(2,"shadowfiend");

        lab9.put(3,"pudge");

        lab9.put(4,"invoker");

**Scanner** n = new Scanner(System.in);

**int** m = n.nextInt();

        System.out.println("HashMap: " +  lab9);

        System.out.println("Is in HashMap: " + lab9.containsKey(m));

        System.out.println("Value: " + lab9.get(m));

    }

}

Изображение выглядит как текст

Автоматически созданное описание