

دستور کار کارگاه برنامه نویسی پیشرفته

به نام خدای رنگین کمان



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

دستور کار کارگاه کامپیوتر

آرین محسنی

فروردین 1402

دستور کار کارگاه برنامه نویسی پیشرفته

فهرست

۱ فهرست
۲ سیستم رأیگیری
۳ Person
۴ کلاس Vote
۴ کلاس vote
۵ کلاس Voting
۷ کلاس VotingSystem
۱۱ کلاس Main

دستور کار کارگاه برنامه نویسی پیشرفته

سیستم رأیگیری

در این تمرین با ساختن کلاس های مختلف مانند کلاس Voting و Person و VotingSystem و تعامل بین آن ها و ساختن اشیا از آن ها این کار را انجام می دهیم.

کلاس Person

این کلاس شامل ۲ فیلد و تعدادی متود است به شکل زیر:

```
class Person {  
    private final String firstname;  
    private final String lastname;  
  
    public Person(String firstname , String lastname){  
        this.firstname = firstname;  
        this.lastname = lastname;  
    }  
  
    @Override  
    public String toString() {  
        return "firstname :" + firstname +  
               " , lastname : " + lastname;  
    }  
}
```

دستور کار کارگاه برنامه نویسی پیشرفته

کلاس Vote

این کلاس شامل 2 فیلد به صورت final و چند متود (Structor) و getter و setter است

```
class Vote {  
    private final Person voter;  
    private final String date;  
  
    public Vote(Person voter, String date) {  
        this.voter = voter;  
        this.date = date;  
    }  
  
    public Person getVoter() {  
        return voter;  
    }  
}
```

کلاس vote

```
public class vote{  
  
    public static void main(String[] args) {  
        Scanner input = new Scanner(System.in);  
        VotingSystem votingSystem = new VotingSystem();  
  
        label:  
        while (true){  
            printMenu();  
            String temp = input.nextLine();  
            switch (temp) {  
                case "1":  
                    votingSystem.createVoting();  
                    break;  
                case "2":  
                    System.out.println("enter your name :");  
                    String name = input.nextLine();  
                    System.out.println("enter your last name :");  
                    String lastName = input.nextLine();  
            }  
        }  
    }  
}
```

دستور کار کارگاه برنامه نویسی پیشرفته

```
votingSystem.vote(name, lastName);
break;
case "3":
    System.out.println("enter your question number:");
    votingSystem.printResult();
    break;
case "4":
    break label;
default:
    System.out.println("invalid input!");
    break;
}
System.out.println("-----");
}
}

public static void printMenu(){
    System.out.println("please choose one of indexes : ");
    System.out.println("1. create a question");
    System.out.println("2. vote to an existing question");
    System.out.println("3. print results");
    System.out.println("4. exit");
}
}
```

Voting کلاس

در این کلاس از چند فیلد (hashmap و arraylist) استفاده میکنیم و سپس متود های getter و

setter

```
class Voting {
    private String question;
    private HashMap <String, HashSet<Vote>> choices;
    private boolean multiChoice;
    private boolean isAnonymous;
    private ArrayList <Person> voters = new ArrayList<>();

    public Voting(String question, boolean multiChoice, boolean isAnonymous) {
```

دستور کار کارگاه برنامه نویسی پیشرفته

```
        this.question = question;
        this.multiChoice = multiChoice;
        this.isAnonymous = isAnonymous;
    }

    public void createChoices(ArrayList<String> givenChoices){
        choices = new HashMap<>();
        for (String givenChoice : givenChoices) {
            choices.put(givenChoice, new HashSet<>());
        }
    }

    public String getQuestion() {
        return question;
    }

    public ArrayList<String> getChoices(){
        ArrayList<String> choices = new ArrayList<>(this.choices.keySet());
        return choices;
    }

    public boolean isMultiChoice() {
        return multiChoice;
    }

    public void vote(String firstName , String lastName , ArrayList<String>
choices){
        Person person = new Person(firstName , lastName);
        voters.add(person);
        for (String choice : choices) {
            HashSet<Vote> votes = this.choices.get(choice);
            Date date = new Date();
            votes.add(new Vote(person, date.toString()));
            this.choices.put(choice, votes);
        }
    }

    public boolean isAnonymous() {
        return isAnonymous;
    }
```

دستور کار کارگاه برنامه نویسی پیشرفته

```
public void printAnonymous(){
    System.out.println(question);
    for (String str:choices.keySet()) {
        System.out.println("\t" + str + " : " + choices.get(str).size());
    }
}

public void printNonAnonymous(){
    System.out.println(question);
    for (String str: choices.keySet()) {
        System.out.println(str + " : {");
        for (Vote vote : choices.get(str)) {
            System.out.println(vote.getVoter());
        }
        if(choices.get(str).size() == 0){
            System.out.println("No voters");
        }
        System.out.println("}");
    }
}
```

کلاس VotingSystem

این کلاس متشکل از یک ArrayList برای لیست رای دهنده و متود های مربوط به ساختن رای و چاپ آن ها می باشد.

```
class VotingSystem {
    ArrayList <Voting> votingList = new ArrayList<>();
    Scanner input = new Scanner(System.in);

    public void createVoting(){
        System.out.println("please type your question :");
        String question = input.nextLine();
```



دستور کار کارگاه برنامه نویسی پیشرفته

```
boolean anonymous;
while (true){
    System.out.println("do you want the vote to be anonymous
:\n1.yes\t2.no");
    String temp = input.nextLine();
    if(temp.equals("yes") || temp.equals("1")){
        anonymous = true;
        break;
    }
    else if(temp.equals("no") || temp.equals("2")){
        anonymous = false;
        break;
    }
    else {
        System.out.println("invalid input!");
    }
}
boolean multiChoice;
while (true){
    System.out.println("do you want the vote to be multiChoice
:\n1.yes\t2.no");
    String temp = input.nextLine();
    if(temp.equals("yes") || temp.equals("1")){
        multiChoice = true;
        break;
    }
    else if(temp.equals("no") || temp.equals("2")){
        multiChoice = false;
        break;
    }
    else {
        System.out.println("invalid input!");
    }
}
ArrayList <String> choices = new ArrayList<>();
System.out.println("please type your choices and when you're done type
\"DONE!\" :");
while (true){
    String choice = input.nextLine();
    if(choice.equals("DONE!")){
        break;
    }
}
```



دستور کار کارگاه برنامه نویسی پیشرفته

```
        }
        choices.add(choice);
    }
    System.out.println("voting successfully created.");

    Voting voting = new Voting(question , multiChoice , anonymous);
    voting.createChoices(choices);
    votingList.add(voting);
}

public boolean showQuestions(){
    if(votingList.size() == 0){
        System.out.println("there is no voting!");
        return false;
    }
    else {
        for (int i = 0; i <votingList.size() ; i++) {
            System.out.println((i+1) + ". " + votingList.get(i).getQuestion());
        }
        return true;
    }
}

public void vote(String firstname , String lastname){
    if(showQuestions()){
        int questionNumber = Integer.parseInt(input.nextLine());
        questionNumber--;
        System.out.println("-----");
        Voting voting = votingList.get(questionNumber);
        ArrayList<String> questionChoices = voting.getChoices();
        System.out.println(voting.getQuestion());
        if(voting.isMultiChoice()){
            for (int i = 0; i < questionChoices.size(); i++) {
                System.out.println((i+1) + ". " + questionChoices.get(i));
            }
            System.out.println("this is a multi choice question enter your
choices and then type\"DONE!\"");
            String temp;
            ArrayList<String> givenChoices = new ArrayList<>();
            while (true){
```

دستور کار کارگاه برنامه نویسی پیشرفته

```
        temp = input.nextLine();
        if(temp.equals("DONE!")){
            break;
        }
        else{
            int index = Integer.parseInt(temp) -1 ;
            givenChoices.add(questionChoices.get(index));
        }
    }
    voting.vote(firstname , lastname , givenChoices);
}
else {
    for (int i = 0; i < questionChoices.size(); i++) {
        System.out.println((i+1) + ". " + questionChoices.get(i));
    }
    ArrayList<String> givenChoices = new ArrayList<>();
    int index = Integer.parseInt(input.nextLine()) -1;
    givenChoices.add(questionChoices.get(index));
    voting.vote(firstname , lastname , givenChoices);
}
}
}

public void printResult(){
if(showQuestions()){
    int questionNumber = Integer.parseInt(input.nextLine());
    questionNumber--;
    System.out.println("-----");
    Voting voting = votingList.get(questionNumber);
    if(voting.isAnonymous()){
        voting.printAnonymous();
    }
    else{
        voting.printNonAnonymous();
    }
}
}

}
```

کلاس Main

```
public class vote {  
    public static void main(String[] args) {  
  
        VotingSystem test = new VotingSystem(new ArrayList<>());  
        Vote vote = new Vote(new Person("ali","mohammadi"),"some date");  
        System.out.println(vote);  
    }  
}
```