

```
public interface Drawable {
    public String draw();
}
```

```
public class Comment implements Drawable {
    private User author_;
    private String content_;

    public Comment(User author, String content) {
        author_ = author;
        content_ = content;
    }

    public void print() {
        System.out.print(author_.getUserName());
        System.out.print(" ");
        System.out.println(content_);
    }

    public String draw() {
        String str = "<div>" + author_.getUserName() + " írta: " + content_ + "</div>";
        return str;
    }
}
```

```
import java.util.HashMap;
import java.util.Map;
```

```
public class PersonalInfo implements Drawable {
    private Map<String, String> hmap_;

    public PersonalInfo() {
        hmap_ = new HashMap<String, String>();
    }

    public PersonalInfo(HashMap<String, String> hmap) {
        hmap_ = hmap;
    }

    public void setTrait(String trait, String value) {
        hmap_.put(trait, value);
    }

    public String getTrait(String trait) {
        return hmap_.get(trait);
    }
}
```

```

}

public String draw() {
    StringBuilder sb = new StringBuilder();

    sb.append("<table style=\"border:1px solid black;\">");

    for (String key : hmap_.keySet()) {
        sb.append("<tr>");

        sb.append("<td>");
        sb.append(key);
        sb.append("</td>");

        sb.append("<td>");
        sb.append(hmap_.get(key));
        sb.append("</td>");

        sb.append("</tr>");
    }

    sb.append("</table>");

    return sb.toString();
}
}

```

```

import java.util.ArrayList;
import java.util.List;

```

```

public class Facebook implements Drawable {
    private List<User> userArray_;
    private List<Comment> commentArray_;
    private List<Drawable> drawableArray_;

    public Facebook() {
        userArray_ = new ArrayList<User>();
        commentArray_ = new ArrayList<Comment>();
        drawableArray_ = new ArrayList<Drawable>();
    }

    public boolean checkUserName(String userNameToCheck) {
        for (User user : userArray_) {
            if (userNameToCheck.equals(user.getUserName())) {
                return false;
            }
        }
    }
}

```

```

        }
    }
    return true;
}

public boolean addUser(User userToAdd) {
    if (checkUserName(userToAdd.getUserName())) {
        userArray_.add(userToAdd);
        drawableArray_.add(userToAdd.getPersonalInfo());
        return true;
    } else {
        return false;
    }
}

public boolean addComment(Comment commentToAdd) {
    commentArray_.add(commentToAdd);
    drawableArray_.add(commentToAdd);
    return true;
}

public void printComments() {
    for (Comment comment : commentArray_) {
        comment.print();
    }
}

public String draw() {
    StringBuilder sb = new StringBuilder();

    sb.append("<html><body>");

    for (Drawable d : drawableArray_) {
        sb.append(d.draw());
        sb.append('\n');
    }

    sb.append("</body></html>");

    return sb.toString();
}
}

```

Userbe:

```
public PersonallInfo getPersonallInfo() {  
    return plInfo_;  
}
```

```
public class Main {  
    public static void main(String[] args) {  
        Facebook fb = new Facebook();  
  
        PersonallInfo kutyInfo = new PersonallInfo();  
        PersonallInfo macsInfo = new PersonallInfo();  
  
        kutyInfo.setTrait("Kor", "5");  
        kutyInfo.setTrait("Faj", "tacsko");  
        kutyInfo.setTrait("Szin", "sotetbarna");  
  
        macsInfo.setTrait("Kor", "8");  
        macsInfo.setTrait("Faj", "leopard");  
  
        User kuty = new User(kutyInfo, "kutya", "007");  
        User macs = new User(macsInfo, "macska", "008");  
        fb.addUser(kuty);  
        fb.addUser(macs);  
  
        fb.addComment(new Comment(kuty, "vauvau"));  
        fb.addComment(new Comment(macs, "miaumiau"));  
  
        System.out.print(fb.draw());  
    }  
}
```