Test.java

public class test {

public static void main(String[] args)

{

Teacher basic = new Teacher("2344567", "Lo", "Le", "LoLe", "lolasdfalulelo", true, "99.99","UNO", "1e4");

basic.setid("2111111");

System.out.println(basic.getUsername());

basic.setPassword("123212342134");

System.out.println(basic.getUserType().returnType());

basic.setPermission(PermissionType.Edit);

System.out.println(basic.getPermission());

System.out.println(basic.getFullName());

System.out.println(basic.verifyUsernameAndPassword("LoLe", "lolasdfalulelo"));

System.out.println(basic.verifyUsername("LoLe"));

System.out.println(basic);

UserLoginAndRegistration login = new UserLoginAndRegistration();

System.out.println(login.addUser(basic));

System.out.println(login.verifyLogIn("LoLe", "sdfasdfasd"));

basic.setSchoolName("lol123");

}

}

User.java

import java.util.\*;

import java.lang.String.\*;

public abstract class User{

private String ID;

private String FirstName;

private String LastName;

private String Username;

private String Password;

private UserType UserType;

private PermissionType Permission;

private boolean status;

protected StringBuilder fullDetail = new StringBuilder();

User(String id, String First, String Last, String username, String Pass, UserType user, PermissionType permis, Boolean value)

{

setid(id);

this.FirstName= First;

this.LastName= Last;

this.Username= username;

setPassword(Pass);

setUserType(user);

this.Permission= permis;

this.status=value;

}

protected void setid(String input)

{

ID = input;

}

public String getUsername()

{

return Username;

}

public void setPassword(String inputpass)

{

try

{

if(inputpass.length() < 7)

{

throw new NullPointerException();

}

else{

this.Password = inputpass;

}

}

catch(NullPointerException e){

System.out.println("the password is too short");

}

}

public UserType getUserType()

{

return UserType;

}

public void setUserType(UserType input)

{

UserType = input;

}

public PermissionType getPermission()

{

return Permission;

}

public void setPermission(PermissionType input)

{

Permission = input;

}

public String getFullName()

{

return FirstName + " " + LastName;

}

public boolean verifyUsernameAndPassword(String username, String password)

{

if( Username.equals(username) == true && Password.equals(password)== true)

{

return true;

}

else

{

return false;

}

}

public boolean verifyUsername(String input)

{

if (Username.equals(input) == true)

{

return true;

}

else

{

return false;

}

}

public String toString()

{

fullDetail.append("Your id is " + ID).append(System.getProperty("line.separator"));

fullDetail.append("Your full name is " + FirstName + " " + LastName).append(System.getProperty("line.separator"));

fullDetail.append("Your UserType is " + UserType).append(System.getProperty("line.separator"));

fullDetail.append("Your Permission level is " + Permission).append(System.getProperty("line.separator"));

if(status == true)

{

fullDetail.append("Your account status is: active").append(System.getProperty("line.separator"));

}

else

{

fullDetail.append("Your account status is: deativated").append(System.getProperty("line.separator"));

}

return fullDetail.toString();

}

public static User findUserByUsername(ArrayList<User> inputdata, String inputuser)

{

for (User i : inputdata)

{

if(i.getUsername() == inputuser)

{

return i;

}

else

{

}

}

return (User) null;

}

public static boolean logInByUsernameAndPassword(ArrayList<User> user, String username, String password)

{

for (User i : user)

{

if(i.getUsername() == username && i.Password == password)

{

return true;

}

else

{

return false;

}

}

return false;

}

}

Teacher.java

import java.util.\*;

public class Teacher extends SchoolUser

{

public Teacher(String id, String First, String Last, String username, String Pass, Boolean value, String grade, String schoolname , String CLASS)

{

super(id,First, Last, username, Pass, UserType.Teacher, value, grade, schoolname, CLASS);

try{

if(id.charAt(0) == '2' && id.length() == 7)

{

}

else {

throw new NullPointerException();

}

}

catch (NullPointerException e)

{

System.out.println(" this is not a Teacher ID or maybe this is not a 7 digit ID");

}

}

@Override

public void setPermission(PermissionType input)

{

if (input == PermissionType.Edit || input == PermissionType.None) {

super.setPermission(input);

}

else {

System.out.println("you can't set this level of permission for a Teacher account");

}

}

@Override

protected void setid(String input)

{

if(input.charAt(0) == '2' && input.length() == 7)

{

super.setid(input);

}

else {

System.out.println("This id can't be set to a Teacher");

}

}

}

Position.java

public enum Position {

NormalTeacher, HeadTeacher, SubtitudeTeacher;

public String returnType()

{

return this.toString();

}

};

SchoolUser.java

import java.lang.String.\*;

public abstract class SchoolUser extends User

{

private String ClassID;

private String Grade;

private String SchoolName;

public SchoolUser(String id, String First, String Last, String username, String Pass, UserType user, Boolean value, String grade, String schoolname, String CLASS)

{

super(id,First,Last,username,Pass,user,PermissionType.None, true);

ClassID=CLASS;

Grade = grade;

SchoolName = schoolname;

}

public String getClassID()

{

return ClassID;

}

public void setClassID(String input)

{

ClassID = input;

}

public String getGrade()

{

return Grade;

}

public void setGrade(String input)

{

Grade = input;

}

public String getSchoolName()

{

return SchoolName;

}

public void setSchoolName(String input)

{

try{

if (input.matches("[0-9]"))

{

throw new NullPointerException();

}

else

{

SchoolName = input;

}

}

catch(NullPointerException e)

{

System.out.println("Can't contain any number");

}

}

@Override

public String toString()

{

super.toString();

fullDetail.append("Your ClassID is: " + ClassID).append(System.getProperty("line.separator"));

fullDetail.append("Your Grade is: " + Grade).append(System.getProperty("line.separator"));

fullDetail.append("Your degree name is: " + SchoolName).append(System.getProperty("line.separator"));

return fullDetail.toString();

}

}

Student.java

import java.util.\*;

public class Student extends SchoolUser

{

Student(String id, String First, String Last, String username, String Pass, UserType user, Boolean value, String grade, String schoolname , String CLASS)

{

super(id,First, Last, username, Pass, UserType.Student, value, grade, schoolname, CLASS);

try{

if(id.charAt(0) == '1' && id.length() == 7)

{

}

else {

throw new NullPointerException();

}

}

catch (NullPointerException e)

{

System.out.println(" this is not a Student ID or maybe this is not a 7 digit ID");

}

}

@Override

public void setPermission(PermissionType input)

{

if (input == PermissionType.Test || input == PermissionType.None) {

super.setPermission(input);

}

else {

System.out.println("you can't set this level of permission for a Student account");

}

}

@Override

protected void setid(String input)

{

if(input.charAt(0) == '1' && input.length() == 7)

{

super.setid(input);

}

else {

System.out.println("This id Can't be set to a Student");

}

}

}

UserLoginAndResgistration.java

import java.util.\*;

public class UserLoginAndRegistration

{

private ArrayList<User> user = new ArrayList<User>();

public UserLoginAndRegistration()

{

}

public boolean verifyLogIn(String username, String password)

{

for (User i : user) {

if(i.verifyUsernameAndPassword(username, password) == true)

{

return true;

}

else {

return false;

}

}

return false;

}

public boolean addUser(User inputUser)

{

if(user.add(inputUser)==true)

{

return true;

}

else

{

return false;

}

}

}

UserType.java

enum UserType {

Student, Teacher, Parent;

public String returnType()

{

return this.toString();

}

};

PermissionType.java

import java.lang.\*;

public enum PermissionType

{

Edit("Allow to view or edit", 1),

View("Only allow to view the result", 2),

Test("Only allow to do the test", 3),

None("Don't have any persmission", 5);

private String name;

private int id;

PermissionType(String descriptor, int input)

{

this.name = descriptor;

this.id = input;

}

public String getName()

{

return name;

}

public int getId()

{

return id;

}

public String toString()

{

return name + " " + id;

}

}

