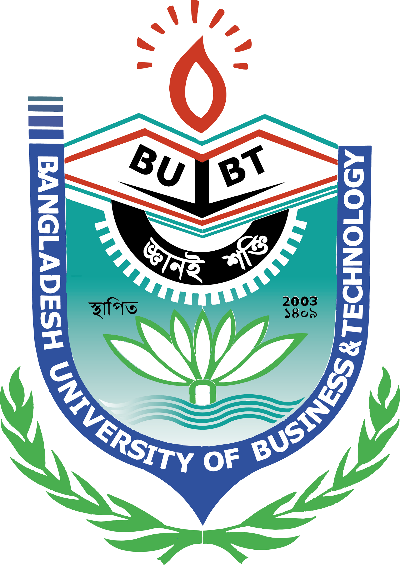
Bangladesh University of Business and Technology (BUBT)

LAB Test



Program: B.Sc. in CSE

Date : 14/02/2022

Semester : Fall 2021

Name of the Student : Md. Noman Faysal Ridoy

ID : 19201103126

Email : nfridoy@gmail.com

Intake : 43

Section : 3

Course Code : CSE 332

Course Title : Advanced Programing

Submitted To : Md. Anwar Hussen Wadud

(Dpt. Of CSE)

Lab Performance-1: (5 Marks)

Create a Car class to represent a fan.

• The car speed is represented by three constants labeled LOW,

NORMAL, and FAST, with values 1,2, and 3 respectively.

• The car speed is specified by an int data field called speed

(default LOW).

• A boolean data field called car on that indicates whether or

not the car is turned on (default false).

• The length and width of the car is specified by a double data

field called car len (default 4) and car width (default 3).

• A color data field that describes the color of the

car (default white).

• A constructor that builds a default car with no arguments.

• The car objects are initialized to provided values using a

parameterized constructor.

• The car description will be shown using the display()

function. When the car is turned on, the display() function

shows the speed, color, and area of the car. If the car is

turned off, the method returns the color and area of the car,

as well as the message "car engine is switched off."

Create two Car objects with a test program. One with default values and the other with

normal speed, length 6, width 4, brown color, and status true turned on. Show the

descriptions for two Car objects that you've built.

Lab Performance-1:

Code:

//Car Class

package LabPerformance1;

public class Car {

private int speed;

private boolean engineOn;

private double length;

private double width;

private String color;

public Car() {

speed = 1;

engineOn = false;

length = 4;

width = 3;

color = "White";

}

public Car(int speed, boolean engineOn, double length, double width, String color) {

this.speed = speed;

this.engineOn = engineOn;

this.length = length;

this.width = width;

this.color = color;

}

void display() {

if(engineOn==true) {

System.out.println("Speed = "+Speed());

System.out.println("Color = "+color);

System.out.println("Area = "+area());

}

else if(engineOn==false) {

System.out.println("Color = "+color);

System.out.println("Area = "+area());

System.out.println("Car engine is switched off.");

}

}

double area() {

return length\*width;

}

String Speed() {

if(speed==1) {

return "LOW";

}

else if(speed==2) {

return "NORMAL";

}

else if(speed==3) {

return "FAST";

}

else {

return "Invalid";

}

}

public int getSpeed() {

return speed;

}

public void setSpeed(int speed) {

this.speed = speed;

}

public boolean isEngineOn() {

return engineOn;

}

public void setEngineOn(boolean engineOn) {

this.engineOn = engineOn;

}

public double getLength() {

return length;

}

public void setLength(double length) {

this.length = length;

}

public double getWidth() {

return width;

}

public void setWidth(double width) {

this.width = width;

}

public String getColor() {

return color;

}

public void setColor(String color) {

this.color = color;

}

}

//Test Class or Main Class

package LabPerformance1;

public class Test {

public static void main(String[] args) {

System.out.println("Weclome");

Car car1 = new Car();

Car car2 = new Car(2,true,6,4,"Brown");

System.out.println("Car 1:::::");

car1.display();

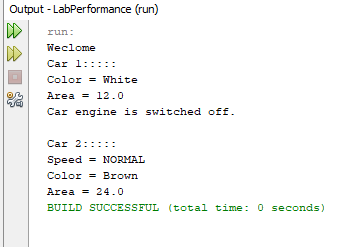
System.out.println("\nCar 2:::::");

car2.display();

}

}

Output:



Lab Performance-2: (5 Marks)

Create a Rectangle class that extends Geometric class. The class has three double data  
fields named length, width, and depth that have default values of 1.0.  
• A default rectangle is created using a no-arg constructor.  
• A rectangle with the given length, width, and depth is created  
with a parameterized constructor.  
• Getter and Setter methods for all data fields.  
• getArea() is a method that returns the area of this rectangle.  
• The perimeter of this rectangle is returned by the  
getPerimeter() function.  
• toString() is a method that returns a string description of  
the rectangle.  
Create a Rectangle object with sides of 1, 1.5, 1, color yellow, and filled true, and display  
the area, perimeter, color, and whether it is filled or not.

Lab Performance-2:

Code:

// Geometric class

package LabPerformance2;

public class Geometric {

protected String color;

protected boolean filled;

public Geometric() {

}

public Geometric(String color, boolean filled) {

this.color = color;

this.filled = filled;

}

}

//Rectangle Class

package LabPerformance2;

public class Rectangle extends Geometric {

private double length;

private double width;

private double depth;

public Rectangle() {

double length;

double width;

double depth;

}

public Rectangle(double length, double width, double depth, String color, boolean filled) {

super(color, filled);

this.length = length;

this.width = width;

this.depth = depth;

}

double getArea() {

return length\*width;

}

double getPerimeter() {

return 2\*length\*width;

}

@Override

public String toString() {

return "Rectangle{" + "length=" + length + ", width=" + width + ", depth=" + depth + ", Color=" + super.color + ", Filled="+ super.filled +"}";

}

public double getLength() {

return length;

}

public void setLength(double length) {

this.length = length;

}

public double getWidth() {

return width;

}

public void setWidth(double width) {

this.width = width;

}

public double getDepth() {

return depth;

}

public void setDepth(double depth) {

this.depth = depth;

}

public String getColor() {

return color;

}

public void setColor(String color) {

this.color = color;

}

public boolean isFilled() {

return filled;

}

public void setFilled(boolean filled) {

this.filled = filled;

}

}

// Test class

package LabPerformance2;

public class Test {

public static void main(String[] args) {

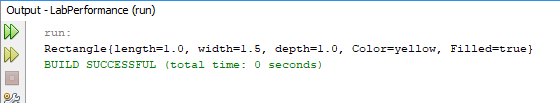
Rectangle r1 = new Rectangle(1,1.5,1,"yellow",true);

System.out.println(r1.toString());

}

}

Output:



Lab Performance-3: (5 Marks)  
Create the User class and its two subclasses, Pupil and Job. Make Teacher and  
Administrative class for Job subclasses.  
A user's information includes his or her name, address, phone number, and email address.  
A pupil's position is determined by his or her class (freshman, sophomore, junior, or  
senior). Assign a constant to the status. Each job has a title, a pay, and a start date. A  
professor has office hours and is assigned a rank. A title is given to a member of the  
administrative job. To show the class name and the user's name, override the toString  
function in each class.  
Create a test program that calls the toString() methods from the objects of all classes.

Lab Performance-3:

Code:

// User Class

package LabPerformance3;

public class User {

protected String name;

protected String address;

protected String phoneNumber;

protected String emailAddress;

public User() {

}

public User(String name, String address, String phoneNumber, String emailAddress) {

this.name = name;

this.address = address;

this.phoneNumber = phoneNumber;

this.emailAddress = emailAddress;

}

@Override

public String toString() {

return "User{" + "name = " + name + ", address = " + address + ", phoneNumber = " + phoneNumber + ", emailAddress = " + emailAddress + '}';

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getAddress() {

return address;

}

public void setAddress(String address) {

this.address = address;

}

public String getPhoneNumber() {

return phoneNumber;

}

public void setPhoneNumber(String phoneNumber) {

this.phoneNumber = phoneNumber;

}

public String getEmailAddress() {

return emailAddress;

}

public void setEmailAddress(String emailAddress) {

this.emailAddress = emailAddress;

}

}

//Pupil class

package LabPerformance3;

public class Pupil extends User {

private String position;

public Pupil() {

}

public Pupil(String position) {

this.position = position;

}

public Pupil(String position, String name, String address, String phoneNumber, String emailAddress) {

super(name, address, phoneNumber, emailAddress);

this.position = position;

}

@Override

public String toString() {

return "Pupil{" + "name = " + getName() + ", address = " + getAddress() + ", phoneNumber = " + getPhoneNumber() + ", emailAddress = " + getEmailAddress() + "position = " + position + '}';

}

public String getPosition() {

return position;

}

public void setPosition(String position) {

this.position = position;

}

}

//Job class

package LabPerformance3;

public class Job extends User {

private String title;

private String pay;

private String startDate;

public Job() {

}

public Job(String title, String pay, String startDate) {

this.title = title;

this.pay = pay;

this.startDate = startDate;

}

public Job(String title, String pay, String startDate, String name, String address, String phoneNumber, String emailAddress) {

super(name, address, phoneNumber, emailAddress);

this.title = title;

this.pay = pay;

this.startDate = startDate;

}

@Override

public String toString() {

return "Job{" + "name = " + getName() + ", address = " + getAddress() + ", phoneNumber = " + getPhoneNumber() + ", emailAddress = " + getEmailAddress() + "title = " + title + ", pay = " + pay + ", startDate = " + startDate + '}';

}

public String getTitle() {

return title;

}

public void setTitle(String title) {

this.title = title;

}

public String getPay() {

return pay;

}

public void setPay(String pay) {

this.pay = pay;

}

public String getStartDate() {

return startDate;

}

public void setStartDate(String startDate) {

this.startDate = startDate;

}

}

//Teacher class

package LabPerformance3;

public class Teacher extends Job {

private String officeHours;

private String rank;

public Teacher() {

}

public Teacher(String officeHours, String rank) {

this.officeHours = officeHours;

this.rank = rank;

}

public Teacher(String officeHours, String rank, String title, String pay, String startDate) {

super(title, pay, startDate);

this.officeHours = officeHours;

this.rank = rank;

}

public Teacher(String officeHours, String rank, String title, String pay, String startDate, String name, String address, String phoneNumber, String emailAddress) {

super(title, pay, startDate, name, address, phoneNumber, emailAddress);

this.officeHours = officeHours;

this.rank = rank;

}

@Override

public String toString() {

return "Teacher{" + "name = " + getName() + ", address = " + getAddress() + ", phoneNumber = " + getPhoneNumber() + ", emailAddress = " + getEmailAddress() + "title = " + getTitle() + ", pay = " + getPay() + ", startDate = " + getStartDate() + ", officeHours = " + officeHours + ", rank = " + rank + '}';

}

public String getOfficeHours() {

return officeHours;

}

public void setOfficeHours(String officeHours) {

this.officeHours = officeHours;

}

public String getRank() {

return rank;

}

public void setRank(String rank) {

this.rank = rank;

}

}

//Administrative class

package LabPerformance3;

public class Administrative extends Job {

public Administrative() {

}

public Administrative(String title, String pay, String startDate) {

super(title, pay, startDate);

}

public Administrative(String title, String pay, String startDate, String name, String address, String phoneNumber, String emailAddress) {

super(title, pay, startDate, name, address, phoneNumber, emailAddress);

}

@Override

public String toString() {

return "Administrative{" + "name=" + getName() + ", address=" + getAddress() + ", phoneNumber=" + getPhoneNumber() + ", emailAddress=" + getEmailAddress() + "title=" + getTitle() + ", pay=" + getPay() + ", startDate=" + getStartDate() + '}';

}

}

//Test class

package LabPerformance3;

public class TestProgram {

public static void main(String[] args) {

User u1 = new User("Md Noman Faysal Ridoy","Nilphamai","019201103126","nfridoy@gmal.com");

Pupil p1 = new Pupil("senior","Noman","Nilphamai","019201103126","nfridoy@gmal.com");

Pupil p2 = new Pupil("junior","NF","Nilphamai","019201103126","nfridoy@gmal.com");

Job j1 = new Job("Software Engineer","100000","23 Feb, 2024","Rahim","Nilphamai","03126","nfridoy@gmal.com");

Teacher t1 = new Teacher("5hrs","Associate Professor","Software Engineer","100000","23 Feb, 2024","Ridoy","Nilphamai","019201103126","nfridoy@gmal.com");

Administrative a1 = new Administrative("Software Engineer","50000","15 Feb, 2024","Faysal","Nilphamai","019201103126","nfridoy@gmal.com");

System.out.println(u1.toString());

System.out.println(p1.toString());

System.out.println(p2.toString());

System.out.println(j1.toString());

System.out.println(t1.toString());

System.out.println(a1.toString());

}

}

OUTPUT:

