

Course: Object Oriented Programming
Lab 01
Primitive Data Types, Conditions, Loops

Task 1: Write a java code, which prints student's information i.e. Roll number, marks in assignment1, GPA, CGPA, section (A or B), gender (print True for male and False for female) or not. For each information, you have to use appropriate data type.

Solution Code:

```
import java.util.Scanner;

public class Lab01 {

    public static void main(String[] args) {
        //Task1(Hard Coded)
        String RollNum = "FA20-BCE-021";
        String name = "Rana Fahad Aman\n";
        int Assignment1 = 25;
        double GPA = 3.14;
        double CGPA = 3.185;
        //true for male, false for female
        boolean gender = true;
        System.out.printf("Name: %s", name);
        System.out.printf("Roll No: %s\n", RollNum);
        System.out.printf("Assignment1 Marks: %d\n", Assignment1);
        System.out.printf("GPA: %.2f\n", GPA);
        System.out.printf("CGPA: %.3f", CGPA);
        if (gender)
            System.out.println("\nGender: Male");
        else {
            System.out.println("\nGender: Female");
        }

        //Task1(User-Input)
        Scanner input = new Scanner(System.in);
        System.out.print("Enter your name: ");
        String name = input.nextLine();
        System.out.print("Enter your Roll No: ");
        String rollnumber = input.nextLine();

        System.out.print("Assignment1 Marks: ");
        int Assign1= input.nextInt();
        System.out.print("GPA: ");
        double GPA = input.nextDouble();
        System.out.print("CGPA: ");
        double CGPA = input.nextDouble();
        System.out.print("Enter Gender (True for Male, False for Female): ");
        boolean gender = input.nextBoolean();

        System.out.println("\n|DISPLAYING DATA|\n");
        System.out.printf("Name: %s\n", name);
        System.out.printf("Roll No: %s\n", rollnumber);
        System.out.printf("Assignment1: %d\n", Assign1);
        System.out.printf("GPA: %.2f\n", GPA);
        System.out.printf("CGPA: %.2f\n", CGPA);
        if (gender)
            System.out.println("Gender: Male\n");
        else
            System.out.println("Gender: Female");
    }
}
```

Task 2: Write java code that prints table of 5.

Solution Code:

```
import java.util.Scanner;

public class Lab01 {

    public static void main(String[] args) {

        int a=5;
        int i;
        System.out.println("Printing the table of 5:"
        );
        for(i=0;i<=10;i++)
        {
            int multiple = a*i;
            System.out.println(a+" * "+i+" = "+multiple);
        }
    }
}
```

Task 3: Write java code that prints even and odd numbers from 1 to 10.

Solution Code:

```
import java.util.Scanner;

public class Lab01 {

    public static void main(String[] args) {
        //Task3

        System.out.print("Even Numbers: ");
        for (int i = 0; i <= 10; i += 2) {
            System.out.print(i + " ");
        }
        System.out.print("\n\nOdd Numbers: ");
        for (int i = 1; i <= 10; i += 2) {
            System.out.print(i + " ");
        }
    }
}
```

Task 4: Write a code, to check whether a given number is positive number or negative number.

Solution Code:

```
import java.util.Scanner;

public class Lab01 {

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("Enter a Number:");
        int num = input.nextInt();
    }
}
```

```
if (num<0)
{
    System.out.println("The entered number is Negative.");
}
else
{
    System.out.println("The entered number is Positive.");
}
}}
```

Task 5: Write java code, which prints grades of students according to marks obtained.

If marks are greater than or equal to 90 print A grade

If marks are greater than or equal to 80 and less than 90 print B grade

If marks are greater than or equal to 70 and less than 80 print C grade

If marks are greater than or equal to 60 and less than 70 print D grade

If marks are greater than or equal to 50 and less than 60 print E grade

If marks are less than 50 print F grade.

Solution Code:

```
import java.util.Scanner;

public class Lab01 {

    public static void main(String[] args) {
        //Task5

        Scanner input= new Scanner(System.in);
        System.out.println("Enter your Marks: ");
        int marks = input.nextInt();
        if (marks>=90){
            System.out.println("Grade : A");
        }
        else if(marks>=80&&marks<=90){
            System.out.println("Grade : B");
        }
        else if ( marks>=70 && marks<=80){
            System.out.println("Grade : C");
        }
        else if ( marks>=60 && marks<=70){
            System.out.println("Grade : D");
        }
        else if ( marks>=50 && marks<=60){
            System.out.println("Grade : E");
        }
        else if ( marks<=50){
            System.out.println("Grade : F" );
        }
    }
}
```

Task 6: Write java code, which prints sum of first ten natural numbers.

Solution Code:

```
import java.util.Scanner;

public class Lab01 {

    public static void main(String[] args) {
        int i,a;
        int sum = 0;
        Scanner input = new Scanner(System.in);
        System.out.print("How many natural numbers do you want to sum together?: ");
        int num = input.nextInt();
        for (i = 1; i <= num; i++) {
            sum = sum+i;
        }
        System.out.println("The sum of the first "+num+" natural numbers is: "+sum);}}}
```

Task 7: Write java code, which prints factorial of given number.

Solution Code:

```
import java.util.Scanner;

public class Lab01 {

    public static void main(String[] args) {
        System.out.print("Enter a number whose factorial you want to calculate: ");
        Scanner input = new Scanner(System.in);
        int num = input.nextInt();
        int i, factorial = 1;
        for (i = 1; i <= num; i++) {
            factorial = factorial * i;
        }
        System.out.printf("The factorial of the entered number is: %d", factorial);}}
```

Task 8: Write java code, which prints the number in reverse order.

Expected Output: Number 1234

Number in reverse order: 4321

Solution Code:

```
import java.util.Scanner;

public class Lab01 {

    public static void main(String[] args) {
        Scanner number = new Scanner(System.in);
        System.out.printf("Enter a number to reverse: ");
        int num = number.nextInt();
        int reverse = 0, remainder;
        while (num != 0) {
            remainder = num % 10;
            reverse = reverse * 10 + remainder;
            num = num / 10;
        }
        System.out.println("The reverse of the number is: " + reverse);
    }
}
```

Course: Object Oriented Programming

Lab 02

Classes, Objects, Methods, User Input

Task 1: Write java code to create the *GradeBook* class that contains a *displayMessage* method to displays a message on the screen. You will need to make an object of this class and call its method to execute display the message.

Now declare a separate class that contains a main method. The *GradeBookTest* class declaration will contain the main method that will control your application's execution.

Expected Output:

```
E:\>java GradeBookTest
Welcome to GradeBook
```

Solution Code:

```
public class gradeBookTest {
    public static void main(String[] args) {
        gradeBook gb1 = new gradeBook();
        gb1.displayMessage();
    }
}
```

```
public class gradeBook{
    public void displayMessage() {
        System.out.println("Welcome to GradeBook.");
    }
}
```

Task 2: Write a class Circle, which will model the functionality of a Circle.

1. Attributes

- ▶ radius

2. Methods

- ▶ *calculateArea()*: To compute area
- ▶ *calculatePerimeter()*: To compute perimeter

Note: For value of pi, use Math.PI

Expected Output:

```
Radius is 3.50
Area is: 38.47
Perimeter is: 21.98
```

Solution Code:

```
import java.lang.Math;

public class Circle {
    double radius;
    double area;
    double perimeter;
    double pi = 3.142;

    public void calculateArea() {
        area = pi * (radius * radius);
    }

    public void calculatePerimeter() {
        perimeter = 2 * pi * radius;
    }
}
```

```
import java.util.Scanner;

public class circleTest {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        Circle c1 = new Circle();
        System.out.println("Enter a value of radius: ");
        c1.radius = input.nextDouble();

        c1.calculateArea();
        c1.calculatePerimeter();

        System.out.println("The area of the circle is: " + c1.area);
        System.out.println("The perimeter of the circle is: " +
c1.perimeter);
    }
}
```

Task 3: Write a class Rectangle, which will model the functionality of a Rectangle.

1. Attributes

- ▶ Length
- ▶ Width

2. Methods

- ▶ *calculateArea()*: To compute area
- ▶ *calculatePerimeter()*: To compute perimeter

Expected Output:

```
Length is 4
Width is 2
Area is: 8.00
Perimeter is: 12.00
```

Solution Code:

```
public class Rectangle {
    double length;
    double width;

    double calculateArea() {
        return width * length;
    }

    double calculatePerimeter() {
        return (2 * width) + (2 * length);
    }
}
```

```
import java.util.Scanner;

public class RectangleTest {
    public static void main(String[] args) {
        Rectangle rect1 = new Rectangle();
        Scanner input = new Scanner(System.in);
        System.out.println("Enter the Length: ");
        rect1.length = input.nextDouble();
        System.out.println("Enter the Width: ");
        rect1.width = input.nextDouble();
        System.out.println("Length is " + rect1.length);
        System.out.println("Width is " + rect1.width);
        System.out.println("Area is: " + rect1.calculateArea());
        System.out.println("Perimeter is: " + rect1.calculatePerimeter());
    }
}
```

Task 4: Make BankAccount class with balance and name attributes of type double and String. Define public void deposit(double amount) and public void withdraw(double amount) methods. Deposit should increase the balance by passed value and withdraw should decrease the balance with passed amount.

Make BankAccountTest class. In main method, create a new object of BankAccount class. Get balance and name of account holder from user input and initialize both object attributes. Then show this menu:

Press 1: To Deposit an amount

Press 2: To Withdraw an amount

Press 3: To View the current balance

If user press 1: show following:

Enter the amount you want to desposit in your account >

For example. If user enters 500, call the desposit method of BankAccount object and pass 500 to it. Do it for option 2 but call the withdraw method. If user choose 3 from menu, print the current balance.

Solution Code:

```
import java.util.Scanner;

public class BankAccount {
    String name;
    double balance;
    double deposit;
    double withdraw;
    Scanner input = new Scanner(System.in);

    public void withdrawAmount() {
        System.out.println("Enter an amount to withdraw from account> ");
        withdraw = input.nextDouble();
        if (balance >= withdraw) {
            balance = balance - withdraw;
            System.out.println("Successfully withdrew " + withdraw + " from account");
            System.out.println("New Balance: " + balance);
        } else {
            System.out.println("\nInsufficient Balance");
        }
    }
}
```



```

        public void depositAmount() {
            System.out.println("Enter an amount to deposit into account> ");
            deposit = input.nextDouble();
            balance = balance + deposit;
            System.out.println("Amount successfully deposited.\nNew balance: " +
balance);
        }
    }
}

```

```

import java.util.Scanner;

import static java.lang.System.exit;

public class BankAccountTest {
    public static void main(String[] arg) {
        int choice;
        Scanner input = new Scanner(System.in);
        BankAccount B1 = new BankAccount();
        System.out.println("Enter name of account holder: ");
        B1.name = input.nextLine();
        System.out.println("Enter balance of account: ");
        B1.balance = input.nextDouble();
        System.out.println("Name of account holder: " + B1.name);
        System.out.println("Balance of account: " + B1.balance);

        do {
            System.out.println("\nMENU:");
            System.out.println("Press 1: To Deposit an amount");
            System.out.println("Press 2: To Withdraw an amount");
            System.out.println("Press 3: View the current balance");
            System.out.println("Press 4: To exit program\n");
            System.out.println("Enter your choice: ");
            choice = input.nextInt();

            switch (choice) {
                case 1 -> B1.depositAmount();
                case 2 -> B1.withdrawAmount();
                case 3 -> System.out.println("Current Balance: " +
B1.balance);
                case 4 -> exit(0);
                default -> System.out.println("Invalid Entry.\nEnter an
option between 1-4");
            }
        } while (true);
    }
}

```

Course: Object Oriented Programming
Lab 03
Encapsulation, Constructors

Task 1: Make a Rectangle class that has color, width and height attribute. Color is of String type, while other two are int type attribute. All the attributes should be private and exposed via setter/getter methods. Value of width and height should be greater than zero.

Define a method inside Rectangle class: *int calculateArea()* that returns area of Rectangle. Define another method in Rectangle class: *int calculatePerimeter()* that returns perimeter of rectangle. Make a RectangleTest class, in main method, instantiate a rectangle object. Ask user to input value of color, width and height, and prints its area and perimeter.

Solution Code:

```
public class Rectangle {
    private String color;
    private int width, height;

    public int calculateArea() {
        int area = width * height;
        return area;
    }

    public int calculatePerimeter() {
        int perimeter = (2 * width) + (2 * height);
        return perimeter;
    }

    public int getWidth() {
        return width;
    }

    public void setWidth(int W) //Width
    {
        if (W < 0)
            System.out.println("Value of Width can not be negative!");
        else
            width = W;
    }

    public int getHeight() {
        return height;
    }

    public void setHeight(int H) //Height
    {
        if (H < 0)
            System.out.println("Value of Height can not be negative!");
        else
            height = H;
    }

    public String getColor() {
        return color;
    }
}
```

```

        public void setColor(String C)    // Color
        {
            color = C;
        }
    }

import java.util.Scanner;

public class testRectangle {
    public static void main(String[] args) {
        Scanner S1 = new Scanner(System.in);
        Rectangle R1 = new Rectangle();
        System.out.println("Enter Color of Rectangle> ");
        R1.setColor(S1.nextLine());
        System.out.println("Enter Height of Rectangle> ");
        R1.setHeight(S1.nextInt());
        System.out.println("Enter Width of Rectangle> ");
        R1.setWidth(S1.nextInt());

        System.out.println("\nColor of Rectangle is: " + R1.getColor());
        System.out.println("Perimeter of Rectangle is: " +
R1.calculatePerimeter());
        System.out.println("Area of Rectangle is: " + R1.calculateArea());

    }
}

```

Task 2: Write a class BankAccount, which have following attributes:

1. Attributes

- ▶ accountNumber
- ▶ accountTitle
- ▶ balance (should be greater than zero)

All attributes are private.

Create a test class named BankAccountTest. In main method, instantiate BankAccount object and display information of account.

Note: Ask user to input value of accountNumber, accountTitle and balance.

Solution Code:

```
public class BankAccount {
    private String accountNumber;
    private String accountTitle;
    private int balance;

    public void setBalance(int B) {
        if (B < 0)
            System.out.println("Balance can not be Negative!");
        else
            balance = B;
    }

    public String getAccountNumber() {
        return accountNumber;
    }

    public void setAccountNumber(String ACN) {
        accountNumber = ACN;
    }

    public String getAccountTitle() {
        return accountTitle;
    }

    public void setAccountTitle(String ACT) {
        accountTitle = ACT;
    }

    public int getAccountBalance() {
        return balance;
    }
}
```

```
import java.util.Scanner;

public class BankAccountTest {
    public static void main(String[] args) {
        Scanner S1 = new Scanner(System.in);
        BankAccount BA1 = new BankAccount();
        System.out.println("Enter Bank Account Number> ");
        BA1.setAccountNumber(S1.nextLine());
        System.out.println("Enter Bank Account Title> ");
        BA1.setAccountTitle(S1.nextLine());
        System.out.println("Enter Bank Account Balance> ");
        BA1.setBalance(S1.nextInt());
        System.out.println("Bank Account Number> " +
BA1.getAccountNumber());
        System.out.println("Bank Account Title> " + BA1.getAccountTitle());
        System.out.println("Bank Account Balance> " +
BA1.getAccountBalance());
    }
}
```

Task 3: Design a class Cylinder, which has following members:

- ▶ Data:
 - ▶ radius
 - ▶ height
- ▶ Constructor
 - ▶ A no-argument constructor. This constructor print this statement “A no argument constructor”.
 - ▶ A one-argument constructor receiving height and setting radius to default value 1
 - ▶ A two-argument constructor receiving height and radius (demonstrate the use of this reference by using same names for parameters and instance variables)
- ▶ Methods:
 - ▶ Setter functions (total 2)
 - ▶ Getter functions (total 2)
 - ▶ computeArea
 - ▶ computeVolume
 - ▶ look for the relevant formulae over internet
 - ▶ displayInfo
 - ▶ Showing cylinder height, radius, area and volume
- ▶ Test class to test Cylinder
 - ▶ Demonstrate all the above constructors/methods by creating different objects and calling each constructor/method at least once

Solution Code:

```
public class Cylinder {
    double radius;
    double height;

    public Cylinder() {
        System.out.println("A no argument constructor.");
    }

    public Cylinder(double height) {
        this.height = height;
        radius= 1;
    }
}
```

```

    }

    public Cylinder(double height, double radius) {
        this.height = height;
        this.radius = radius;
    }

    public double getHeight() {
        return height;
    }

    public void setHeight(double height) {
        this.height = height;
    }

    public double getRadius() {
        return radius;
    }

    public void setRadius(double radius) {
        this.radius = radius;
    }

    public double computeArea() {
        return ((2 * Math.PI * radius* height) + (2 * Math.PI * radius * radius));
    }

    public double computeVolume() {
        return (Math.PI*radius*radius*height);
    }

    public void displayInfo() {
        System.out.println("Height of Cylinder> " + getHeight());
        System.out.println("Radius of Cylinder> " + getRadius());
        System.out.println("Area of Cylinder> " + computeArea());
        System.out.println("Volume of Cylinder> " + computeVolume());
    }
}

```

```

public class CylinderTest {
    public static void main(String[] args) {
        System.out.println("|No Argument Constructor|");
        Cylinder C1 = new Cylinder(); //No Argument Constructor
        Cylinder C2 = new Cylinder(10); //One-Argument Constructor
        Cylinder C3 = new Cylinder(25,10); //Two-Argument Constructor

        /*
        System.out.println("|Data of C1|");
        System.out.println("Radius: " + C1.radius + "\n" + "Height: " +
C1.height + "\n" + "Area: " + C1.computeArea() + "\n" + "Volume:" +
C1.computeVolume());
        System.out.println("\n|Data of C2|");
        System.out.println("Radius: " + C2.radius + "\n" + "Height: " +
C2.height + "\n" + "Area: " + C2.computeArea() + "\n" + "Volume:" +
C2.computeVolume());
        System.out.println("\n|Data of C3|");
        System.out.println("Radius: " + C3.radius + "\n" + "Height: " +
C3.height + "\n" + "Area: " + C3.computeArea() + "\n" + "Volume:" +
C3.computeVolume());
        */
    }
}

```

```
System.out.println("\n|Data of Cylinder 1|");
C1.displayInfo();
System.out.println("\n|Data of Cylinder 2|");
C2.displayInfo();
System.out.println("\n|Data of Cylinder 3|");
C3.displayInfo();
}
}
```

Task 4: Write a date class to model a date object.

► Date

- year
- month
- day

Value of day should not be greater than 30, if day value is greater than 30 reset it one.

Value of month should not be greater than 12, if month value is greater than 12 then reset it to one.

► Constructors

- No argument constructor. In this constructor set Day value to 7, month value to 10 and year to 2020.
- One parameterized constructor for setting all three fields

► Methods

- A method to print date in “dd/mm/yy” format
- Implement the setter and getter methods for day and month.

► Write DateTest class for the Date class

- Create two date objects **d1** and **d2** by calling constructors with appropriate values
- Input date, month and year from user using Scanner object
- Create two reference variables (**d3** and **d4**) of type date.
- Print date for **d1**, **d2** by calling print method of date.

- ▶ Assign **d2** to a new object **d3** and **d3** to **d4**
- ▶ Change **d3** month through the setter methods
- ▶ Print **d1, d2, d3 and d4** by calling print method of date and observe the effect of reference types

Solution Code:

```
public class Date {
    private int year, month, day;

    public Date() {
        day = 7;
        month = 10;
        year = 2020;
    }

    public Date(int day, int month, int year) {
        setDay(day);
        setMonth(month);
        this.year = year;
    }

    public int getMonth() {
        return month;
    }

    public void setMonth(int month) {
        if (month > 12)
            setMonth(1);
        else
            this.month = month;
    }

    public int getDay() {
        return day;
    }

    public void setDay(int day) {
        if (day > 30)
            setDay(1);
        else
            this.day = day;
    }

    public void printDate() {
        System.out.printf("%d/%d/%d", day, month, year);
    }
}
```



```
import java.util.Scanner;

public class DateTest {
    public static void main(String[] args) {
        Scanner S = new Scanner(System.in);
        Date d1 = new Date();
        System.out.println("Enter day: ");
        int a = S.nextInt();
        System.out.println("Enter month: ");
        int b = S.nextInt();
        System.out.println("Enter year: ");
        int c = S.nextInt();
        Date d2 = new Date(a,b,c);
        System.out.println("\n|Before using reference variables|");
        System.out.println("|Date 1|");
        d1.printDate();
        System.out.println("\n|Date 2|");
        d2.printDate();

        Date d3 = d2;
        Date d4 = d3;

        d3.setMonth(6);

        System.out.println("\n|After using reference variables:");
        System.out.println("|Date 1|");
        d1.printDate();
        System.out.println("\n\n|Date 2|");
        d2.printDate();
        System.out.println("\n\n|Date 3|");
        d3.printDate();
        System.out.println("\n\n|Date 4|");
        d4.printDate();
    }
}
```