

## MTK467 Nesneye Yönelik Programlama Alıştırma Çözümleri

Aşağıdaki programların(1-2) çıktılarını yazınız.

1)

```
Metodu çağırmadan önce: times = 3
n = 3
Mesajınız var!
n = 2
Mesajınız var!
n = 1
Mesajınız var!
Metodu çağırdıktan sonra: times = 3
```

2)

```
2 6 10 14
2 6 10 14
```

3)

```
public class PrintStars {
    public static void main(String[] args){
        Scanner input = new Scanner(System.in);

        int numOfRows = input.next(); //int tipli kullanıcı girdisi next() metoduyla alınmaz
        int numOfColumns = input.next(); //int tipli kullanıcı girdisi next() metoduyla alınmaz

        for(int i=0; i<numOfRows; i++){
            for(int j=0; j<numOfColumns; j++){
                System.out.print("*");
                System.out.println(); //içteki for döngüsünün dışında olmalı, yoksa yıldızları alt alta yazdırır
            }
        }
    }
}
```

Doğrusu:

```
public class PrintStars {
    public static void main(String[] args){
        Scanner input = new Scanner(System.in);

        int numOfRows = input.nextInt(); //int tipli kullanıcı girdisi next() metoduyla alınmaz
        int numOfColumns = input.nextInt(); //int tipli kullanıcı girdisi next() metoduyla alınmaz

        for(int i=0; i<numOfRows; i++){
            for(int j=0; j<numOfColumns; j++){
                System.out.print("*");
            }
            System.out.println();
        }
    }
}
```

4) Aşağıdaki programdaki hata veya hataları bulunuz ve nedenini açıklayınız.

```

public class TestMethodOverload {

    public static void displaySum(int x, int y){ //Birebir aynı parametre listesiyle method overloading yapılamaz.
        System.out.printf("Sum of %d and %d is %d ", x, y, x+y);
    }

    public static int displaySum(int x, int y){

        System.out.printf("First number is %.2f \n", x); //x, y ve x+y int tipli olacağından %d olmalılar
        System.out.printf("Second number is %.2f \n", y);
        System.out.printf("Sum is %.2f \n", x+y);
        return x + y;
    }

    public static void main(String[] args){

        int num1 = 3;
        int num2 = 2;

        displaySum(num1, num2);
    }
}

```

Çözüm olarak overloaded displaySum metotlarından biri silinebilir veya ikinci metot aşağıdaki gibi tanımlanabilir

```

public static float displaySum(float x, float y){

    System.out.printf("First number is %.2f \n", x);
    System.out.printf("Second number is %.2f \n", y);
    System.out.printf("Sum is %.2f \n", x+y);
    return x + y;
}

```

5) Aşağıdaki programda bir dizi not değerleri 10 arttırılıp yeni bir dizi oluşturmak isteniyor. Buna göre programdaki hata veya hataları bulunuz ve düzeltiniz.

```

public static void main(String[] args){

    double[] originalGrades = {35.5, 67.8, 82.3, 90.2};

    final int GRADE_COUNT = originalGrades.length;

    double[] modifiedGrades = new double[GRADE_COUNT];

    System.arraycopy(originalGrades, 0, modifiedGrades, 0, GRADE_NUMBER);

    for(double value:modifiedGrades){ //özel for döngüsüyle yalnızca dizinin elemanlarına
        value += 10; //ulaşabiliriz, onları değiştiremeyiz.
    }
}

```

Doğrusu:

```

public static void main(String[] args){

    double[] originalGrades = {35.5, 67.8, 82.3, 90.2};

    final int GRADE_COUNT = originalGrades.length;

    double[] modifiedGrades = new double[GRADE_COUNT];

    System.arraycopy(originalGrades, 0, modifiedGrades, 0, GRADE_NUMBER);

    for(int i=0; i<GRADE_NUMBER; i++){
        modifiedGrades[i] += 10;
    }
}

```