Guessing game (10 points)

Write a guessing game program that asks the user to guess your secret number. If the guess is too high, write a message saying so and ask them to guess again. Similarly, if the guess is too low, write a message saying so and ask them to guess again. If they guess correctly, congratulate them and end the program. You can assume the user will only enter integers.

Program

GuessingGame.java

```java
import java.util.Scanner;
public class GuessingGame{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        System.out.println("-----------------------------------------------------");

        int MyNumber,Guess;
        MyNumber=78;

        System.out.print("Guess my secret number: ");
        Guess=sc.nextInt();

        while(Guess != MyNumber){
            if(Guess > MyNumber){
                System.out.print("Too high,Try again: ");
                Guess=sc.nextInt();
            }else{
                System.out.print("Too low!,try again: ");
                Guess=sc.nextInt();
            }
        }
        System.out.println("\nCongratulation!,You guesed the number ("+MyNumber+").");
        System.out.println("-----------------------------------------------------");
    }
}

```

Output

```
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1> javac GuessingGame.java
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1> java GuessingGame
-----------------------------------------------------
Guess my secret number: 34
Too low!,try again: 100
Too high,Try again: 78

Congratulation!,You guesed the number (78).
-----------------------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1>
```

Expanding word (5 points)

Write a program that asks the user for a word and prints back out a string in which the first letter of the input word appears once, followed the second letter appearing twice, followed by the third letter appearing three times, etc.

For example, the word cat would be output as `caattt`, the word zebra would be output as `zeebbbrrrraaaaa`, and the word a would be output as a.

Program

ExpandingWord.java

```java
import java.util.Scanner;
public class ExpandingWord{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);

        System.out.println("----------------------------------------");
        System.out.print("Enter any word: ");
        String word=sc.nextLine();

        String ExpandedWord="";

        for(int i = 0; i < word.length(); i++){
            for(int j = 0; j <= i; j++){
                ExpandedWord += word.charAt(i);
            }
        }
        System.out.println("\nYour word is: "+word+"\nExpanded format: "+ExpandedWord);
        System.out.print("----------------------------------------");
    }
}

```

Output

```
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1> javac ExpandingWord.java
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1> java ExpandingWord
----------------------------------------
Enter any word: zebra

Your word is: zebra
Expanded format: zeebbbrrrraaaaa
----------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1>
```

Pig Latin (10 points)

Write a program that asks the user for a sentence and prints the translation of the sentence in the "language" Pig Latin. Each word in the sentence should be translated into Pig Latin in the following way:

1) If the English word starts with one or more consonants, all consonants before the first vowel are removed from the front of the English word, and added to the end of it, keeping the order the same. The string "ay" is then added to the end of this string.
2) If the English word starts with a vowel, append the string "hay" to the end of that word.

Treat 'y' as a consonant if it is at the beginning of a word, and otherwise treat it as a vowel. You may assume that the only punctuation used in the sentence are periods and commas. The displayed sentence should have the same punctuation as the original sentence, but should be all lower-case.

For example, the input `hello` should be displayed at `ellohay`, and the sentence `The elephant ate a mango`. Should be displayed at `ethay elephanthay atehay ahay angomay`.

Program

PigLatin.java

```java
import java.util.Scanner;
public class PigLatin{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);

        System.out.println("-----------------------------------------------------------------");

        System.out.print("Enter an English sentence: ");
        String sentence = sc.nextLine().toLowerCase();

        String[] vowels = {"a", "e", "i", "o", "u"};
        String latin = "";
        String[] words = sentence.split(" ");

        for(String word : words){
            boolean startsWithVowel = false;
            for(String vowel : vowels){
                if(String.valueOf(word.charAt(0)).equals(vowel)){
                    startsWithVowel = true;
                    break;
                }
            }

            if(startsWithVowel){
                latin += word + "hay ";
            }else{
                int index = -1;
                for(int i = 0; i < word.length(); i++){
                    for(String vowel : vowels){
                        if(String.valueOf(word.charAt(i)).equals(vowel)){
                            index = i;
                            break;
                        }
                    }
                    if(index != -1){
                        break;
                    }
                }
                if(index != -1){
                    latin += word.substring(index) + word.substring(0, index) + "ay ";
                }else{
                    latin += word + "ay ";
                }
            }
        }
        System.out.println("\nPIG LATIN: " + latin.trim());
        System.out.println("-----------------------------------------------------------------");
    }
}

```

Output

```
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1> javac PigLatin.java
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1> java PigLatin
-----------------------------------------------------------------
Enter an English sentence: The elephant ate a mango

PIG LATIN: ethay elephanthay atehay ahay angomay
-----------------------------------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1>
```

Grade Calculator (10 points)

Write a program that asks the user for a list of grades, and returns the average of the grades. You can chose the format in which the grades are entered, but you should communicate this clearly to the user in the program.

For example, if the user enters the grades 89 81 94, the program returns the average 88.

Program

GradeCalculator.java

```java
import java.util.Scanner;
public class GradeCalculator{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);

        System.out.println("----------------------------------------");

        int average;
        int sum = 0;
        System.out.print("Enter list of grades(separate by space): ");
        String put=sc.nextLine();

        String[] grades = put.split(" ");

        for(String grade : grades){
            int g = Integer.valueOf(grade);
            sum+=g;
        }
        average = sum/grades.length;
        System.out.println("\nThe average of the grades: "+average);

        System.out.println("----------------------------------------");
    }
}

```

Output

```
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1> javac GradeCalculator.java
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1> java GradeCalculator
----------------------------------------
Enter list of grades(separate by space): 81 89 94

The average of the grades: 88
----------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1>
```

Extra credit (5 points):

Modify your program so that the user can enter a weight for each grade. The program then returns the weighted average of the grades.

For example, if the user enters the grades 89 81 94 with the weights 0.25 0.25 0.5, the program returns the weighted average 89.5.

Program

ExtraCredit.java

```java
import java.util.Scanner;
public class ExtraCredit{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);

        System.out.println("-----------------------------------------------------");

        double WeightedAverage = 0;

        System.out.print("Enter list of grades(separate by space): ");
        String putGr=sc.nextLine();
        String[] grades = putGr.split(" ");

        System.out.print("Now Enter list of grades weight(separate by space): ");
        String putWt=sc.nextLine();
        String[] weights = putWt.split(" ");

        while(grades.length != weights.length){
            System.out.println("invalid length!,You enter many or few weights according to grades");
            System.out.print("Enter grade weight again(separate by space): ");
            putWt=sc.nextLine();
            weights = putWt.split(" ");
        }

        for(int i = 0; i < grades.length; i++){
            int grade = Integer.valueOf(grades[i]);
            double weight = Double.valueOf(weights[i]);
            WeightedAverage += (grade*weight);
        }
        System.out.println("\nThe average of the grades: "+WeightedAverage);

        System.out.println("-----------------------------------------------------");
    }
}

```

Output

```
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1> javac ExtraCredit.java
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1> java ExtraCredit
-----------------------------------------------------
Enter list of grades(separate by space): 81 89 94
Now Enter list of grades weight(separate by space): 0.25 0.25 0.25 0.5
invalid length!,You enter many or few weights according to grades
Enter grade weight again(separate by space): 0.25 0.25 0.5

The average of the grades: 89.5
-----------------------------------------------------
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1>
```

MotorBoat Class (25 points)

Implement a class MotorBoat that represents motorboats. A motorboat has attributes for:

- The capacity of the fuel tank
- The amount of fuel in the tank
- The maximum speed of the boat
- The current speed of the boat
- The efficiency of the boat's motor
- The distance traveled

The class has methods to

- constructor
- Return the distance traveled so far

If the boat has efficiency *e*, the amount of fuel used when traveling at a speed *s* for time *t* is *e x s^2 x t*. The distance traveled in that time is *s x t*.

Include a driver program that instantiates several objects of type MotorBoat and demonstrates the class.

Program

MotorBoat.java

```java
public class MotorBoat{
    private double capacity;
    private double amount;
    private double MaxSpeed;
    private double CurrentSpeed;
    private double efficiency;
    private double distance;

    public MotorBoat(double c,double a,double ms,double cs,double e){
        capacity =c;
        amount = a;
        MaxSpeed = ms;
        CurrentSpeed = cs;
        efficiency = e;
        distance = 0;
    }
    public double getMaxDistance(int time){
        return MaxSpeed * time;
    }
    public double getMinDistance(int time){
        return CurrentSpeed * time;
    }
    public static void main(String[] args){
        MotorBoat M1 = new MotorBoat(12.0,1000.0,180,60,15.0);
        System.out.println("\ndistance travell with curent speed: "+M1.getMinDistance(12));
        System.out.println("distance travell with Maximum speed: "+M1.getMaxDistance(12));
    }
}

```

Output

```
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1> javac MotorBoat.java
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1> java MotorBoat

distance travell with curent speed: 720.0
distance travell with Maximum speed: 2160.0
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1>
```

PersonAddress Class (20 points)

Implement a class PersonAddress that represents an entry in an address book. Its attributes are:

- The first name of the person
- The last name of the person

- The email address of the person
- The telephone number of the person

It will have methods to

- constructor
- Test whether two instances are equal based solely on name

Include a driver program that instantiates several objects of type PersonAddress and demonstrates the class.

Program

PersonAddress.java

```java
public class PersonAddress{
    private String FirstName;
    private String LastName;
    private String Email;
    private String TelephoneNo;

    public PersonAddress(String fn,String ln,String em,String tel){
        FirstName =fn;
        LastName =ln;
        Email = em;
        TelephoneNo = tel;
    }

    public boolean isEqual(PersonAddress person){
        return FirstName.equals(person.FirstName) && LastName.equals(person.LastName)
        && Email.equals(person.Email) && TelephoneNo.equals(person.TelephoneNo);
    }

    public static void main(String[] args){
        PersonAddress person1 = new PersonAddress("abdillah","rashid","abdi@gmail.com","0773279297");
        PersonAddress person2 = new PersonAddress("abdillah","rashid","abdi@gmail.com","0773279297");

        if(person1.isEqual(person2)){
            System.out.println("name one and two are equal");
        }else{
            System.out.println("name of first person is not equal to that of second person");
        }
    }
}

```

Output

```
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1> javac PersonAddress.java
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1> java PersonAddress
name one and two are equal
PS C:\Users\Abdillah's PC\Desktop\Java\OOP_Problem_set_1>
```