```
 /* HW3
The purpose of this homework is to make you understand: classes and objects,
    methods , if statements and Loops.

The idea of this HW is as followes :
A password must meet special requirements, for instance , it has to be of
specific length, contains at least 2 capital letters , 2 lowercase letters,
2 symbols and 2 digits. A customer is hiring you to create a class that
can be utilized for several clients of the customer.

The first object created will be for the first level clients which requires
a moderate level password which contains ( at least 1 capital letters ,
1 lowercase letters,1 symbol and 1 digit).
The first level should be created using a default constructor.
The length of any password should be at least 8 characters for all levels.
Hint, should be static.

The second object created will be for the second level clients which requires
a harder level password which contains ( at least 2 capital letters ,
2 lowercase letters,2 symbol and 2 digits).
The second level should be created using an overloaded constructor.
the length of the second level password should be at least 8 as it
required for any levels.


*/


import java.util.Scanner;

public class Password{

/*(1 point| COMPLETE) Create a private static default password variable and set
    it to Def@u1tPa$$w0rd which meets the standards of accepted password*/

//(1 point| COMPLETE) Create a private static int length of the password and set to
8.

// Private settings that should be met for each password instance/objects
//(1 point) Create a private variable to store the number of symbols

//(1 point) Create a private variable to store the number of capital letters

//(1 point) Create a private variable to store the number of lower case letters

//(1 point) Create a private variable to store the number of digits

// (1 point)Create a private variable to store the password
```

```
/** (6 points) Create the default constructor, set the default required number
 * for capital and lowercase letters, symbols, digits to 1
 * set the password variable to the default password
*/
    Password()
    {   // default values


    }

    /**( 6 points) Create an overloaded constructor ,that takes
     * number of ( symbols , capital letters, small letters, digits)
     * these will be considered settings for initializing an instance of
     * the password set the global private variables (also known as data fields)
     *  to the passed in arguments
     */
    Password (  int numSymbols, int numCap, int numSmall, int numDigits )
    {

    }



    /**(3 points) Create a method that takes a string password and check if
     * it is equal to the length specified then return true, false if not */
    public boolean validLength(String pass)
    {


    }

     /**(10 points) Create a method that takes a string password ,
     * the method checks if the password has at least the
     * required number of symbols and return true, false if not


            // Declare a counter variable

            // Declare a boolean to hold the answer

           // Loop through the length of the password

         // Once you counted the required number, set answer to true and break


         // Using the Ascii table, check each index if in the range [32 -47] or
[58-64]
```

```
                    // increment the count


            // return the answer

        */


    /**Create a method that takes a string password ,
    * the method checks if the password has at least the
    * required number of digits and return true, false if not
    * the style of this method will be similar to the previous method
    * use the range in the Ascii table [48 -57] for digits
    */




    /**Create a method that takes a string password ,
    * the method checks if the password has at least the
    * required number of capital letters and return true, false if not
    * the style of this method will be similar to the previous method
    * use the range in the Ascii table [65 -90] for capital letters
    */


    /**Create a method that takes a string password ,
    * the method checks if the password has at least the
    * required number of lowercase letters and return true, false if not
    * the style of this method will be similar to the previous method
    * use the range in the Ascii table [97 -122] for lowercase letters
    */


    /**(2 points) create a getter method to return the password*/


    /** Create a setter method to set the password*/
    public void setPassword(  )
    {
        // Declare a String to hold a password

        // Declare a scanner object to receive a password from the keyboard

        // Declare a boolean variable to be set whenever a password meets the
settings.

        // Loop until the user provides a correct password

                // prompt the user to enter the password , specify the requirements
```

```
        // scan the next line and store in the String holding the password


        /* If the password provided is not equal to the length required,
           then print out an error message*/


        /* Else if the password doesn't have the required number of capital letters
        then print out a message*/


        /* Else if the password doesn't have the required number of lowercase
letters
        then print out a message*/


        /* Else if the password doesn't have the required number of symbols
        then print out a message*/


        /* If the password doesn't have the required number of digits
        then print out a message*/

        /* Else , password provided is correct
            set the global variable password to the new qualified password
            set the flag to false, to stop iterations
        */
        }



    /**
     * create the main method
     */
    public static void main(String[] args) {

    //Declare an instance of Password using the default constructor

    // print out the password, using the getter method

    // What is the  default password ?.................

    // Use the setter method to set a password

    // Print out the password using the getter method


    /*Declare an instance of the password using the overloaded constructor,
```

```
     the settings for new password object are:
    (2 Captial letters,2 Lowercase letters, 2 Symbols , 2 Numbers)*/

     // Set the password using the setter method.

     // Print out the password using the getter method




     }

}
```