

Secret Agent

Make an Agent class.

You decide which information is needed, but id, real name and code name must be present.

The Agent must also have a list of 2 spoken languages,

The real name, code name, first language and second language is created by the value the user inserts in the webform. The id is created automatically, the counter variables always grow, so the ids of the agents is never the same.

I have made the two languages as two string variables, but I could have made an arrayList with a max of 2 indexes.

```
//Instance variables
private int id;
public string realName;
private string codeName;
private string firstLanguage;
private string secondLanguage;

//Constructor
public Agent(string realName, string codeName, string first, string second)
{
    this.realName = realName;
    this.codeName = codeName;
    this.firstLanguage = first;
    this.secondLanguage = second;
    counter++;
    this.id = counter;
}
```

Secret Agent

A method which can set or update the 2 languages.

The Agent class must have a ToString method.

I use the property method to change the two languages.

And a simple ToString method with all the values in each agent objects.

```
//Property for id
public int Id { get { return id; } }

//Property for real name and code name
public string RealName { get { return realName; } }
public string CodeName { get { return codeName; } }

//Property for first and second language
public string FirstLanguage
{
    get { return firstLanguage; }
    set { firstLanguage = value; }
}

public string SecondLanguage
{
    get { return secondLanguage; }
    set { secondLanguage = value; }
}

//ToString
public override string ToString()
{
    return realName + "'s codename is " + codeName + ", and has the id: " + id + ". "
    + codeName + " can speak " + firstLanguage + " and " + secondLanguage;
}
```

```

public class Encryption
{
    //Class variable
    private static string[] alphabet = { "a", "b", "c", "d", "e", "f", "g", "h",
    "i", "j", "k", "l", "m", "n", "o", "p", "q", "r", "s", "t", "u", "v", "w",
    "x", "y", "z" };

    //Encryption method
    public static string Encrypt(string msg)
    {
        string write = "";

        for (int i = 0; i < msg.Length; i++)
        {
            string letter = msg.Substring(i, 1);

            if (alphabet.Contains(letter))
            {
                string newLetter = "";
                int indexOfLetter = Array.FindIndex(alphabet, row =>
                row.Contains(letter));
                if (indexOfLetter == 0)
                {
                    newLetter = "w";
                    write += newLetter;
                }
                else if (indexOfLetter == 1)
                {
                    newLetter = "x";
                    write += newLetter;
                }
                else if (indexOfLetter == 2)
                {
                    newLetter = "y";
                    write += newLetter;
                }
                else if (indexOfLetter == 3)
                {
                    newLetter = "z";
                    write += newLetter;
                }
                else
                {
                    newLetter = letter;
                    write += newLetter;
                }
            }
            else
            {
                write += letter;
            }
        }

        return write;
    }
}

```

Secret Agent

Make an Encryption class.

The class should have a method that receive a string and a key for a Caesar encryption.

The method should return the encrypted string. (See en.wikipedia.org "Caesar Cipher")

I chose to use the alphabet in an array, where it search for the single letters found by the Substring method in a loop. After that, it picks the letter in the index 4 numbers smaller.

The new letter is then added to the write variable. At last, it returns the new message (string write).

Secret Agent

Make a single webpage with two tasks :
Creation of agents, and handling agent data and encryption.

This is how the webpage looks.

The image shows a wireframe for a webpage titled "Secret Agent". The page is divided into four distinct sections, each with a different background color and specific form elements:

- Create a Agent (Light Green):** Contains three input fields labeled "Real name", "Code name", and "Languages", followed by a "Create" button.
- Make changes to Agent (Light Orange):** Contains two input fields labeled "Real name" and "The new Languages", followed by an "Update" button.
- Agent list (Light Blue):** Contains a "Show All Agents" button and a large, empty white rectangular area with a vertical scrollbar on the right side, intended for displaying a list of agents.
- Encrypt a message (Dark Red):** Contains one input field labeled "Your Message" and a "Button" below it.

Secret Agent

The creation part should place the agents an ArrayList.
(Note : In this hand in the ArrayList must be static (belong to a class), else it is lost after aPostBack).

The array list is first declared, so I can use it later.

When the user press create this function will be executed. It checks if the Text boxes is filled, and if they all are it creates an agent object and adds it to the agentArrayList.

```
// Creates an array for the agent objects
static ArrayList agentArrayList;
protected void Page_Load(object sender, EventArgs e)
{
    if (!Page.IsPostBack)
    {
        agentArrayList = new ArrayList();
    }
}
```

```
protected void ButtonCreateAgent_Click(object sender, EventArgs e)
{
    if(TextBoxRealName.Text != "")
    {
        if(TextBoxCodeName.Text != "")
        {
            if(TextBoxCreateFirstLang.Text != "" ||
                TextBoxCreateSecondLang.Text != "")
            {
                Agent a = new Agent(TextBoxRealName.Text,
                    TextBoxCodeName.Text, TextBoxCreateFirstLang.Text,
                    TextBoxCreateSecondLang.Text);
                agentArrayList.Add(a);
                TextBoxRealName.Text = "";
                TextBoxCodeName.Text = "";
                TextBoxCreateFirstLang.Text = "";
                TextBoxCreateSecondLang.Text = "";
            }
            else
            {
                CreateAgentInfo.Text = "Every agent needs to know two
                    languages";
            }
        }
        else
        {
            CreateAgentInfo.Text = "You need to fill in the code name";
        }
    }
    else
    {
        CreateAgentInfo.Text = "You need to fill in the real name";
    }
}
```

Secret Agent

The handling part should have TextBox's for a real name, two languages, a text and a key.

If the real name does not exist the page should do nothing.

Else the page should change the agents languages, display all data for the agent and display the coded message.

I chose to separate the two options, just for the user experience.

Again the function checks if all the input fields is filled, and if they are it loop throw the agentArrayLists objects.

If the text in the TextBoxCheckRealName is found in the agent objects then the two languages is going to be updated with the two new from the input fields.

```
protected void ButtonUpdateAgent_Click(object sender, EventArgs e)
{
    ListBoxAgentInfo.Items.Clear();
    if (TextBoxCheckRealName.Text != "")
    {
        if(TextBoxNewFirstLang.Text != "" && TextBoxNewSecondLang.Text != "")
        {
            for (int n = 0; n < agentArrayList.Count; n++)
            {
                if (((Agent)agentArrayList[n]).RealName ==
                    TextBoxCheckRealName.Text)
                {
                    LabelUpdateAgentInfo.Text = "";
                    ((Agent)agentArrayList[n]).FirstLanguage =
                        TextBoxNewFirstLang.Text;
                    ((Agent)agentArrayList[n]).SecondLanguage =
                        TextBoxNewSecondLang.Text;
                    LabelUpdateAgentInfo.Text = "Agent " +
                        ((Agent)agentArrayList[n]).CodeName + " profile is
                        updated";
                }
                else
                {
                    LabelUpdateAgentInfo.Text = "Could not find an agent with
                        that name";
                }
            }
        }
        else
        {
            LabelUpdateAgentInfo.Text = "You have to write the two new
                languages";
        }
    }
    else
    {
        LabelUpdateAgentInfo.Text = "You have to write a name";
    }
}
```

Make changes to Agent

Real name
Henrik

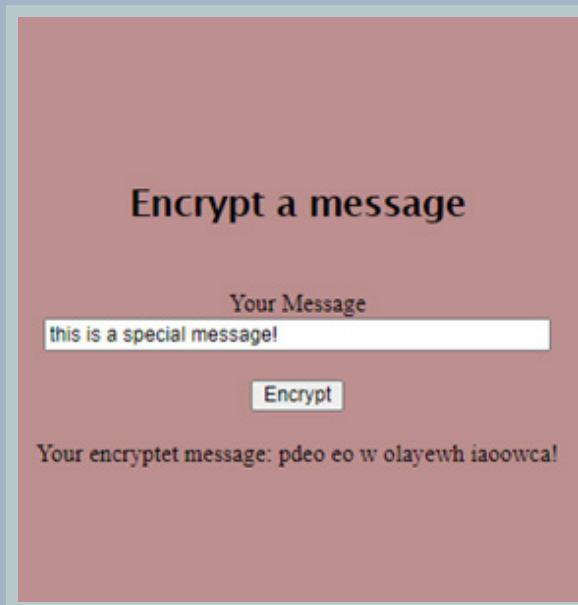
The new Languages
Japanese

The new Languages
English

Update

Agent Lynet profile is updated

Secret Agent



```
//Encrypt a message
protected void ButtonEncryptMsg_Click(object sender, EventArgs e)
{
    LabelEncryptMsg.Text = "Your encryptet message: " +
        Encryption.Encrypt(TextBoxMessage.Text);
}
```

When the encrypt button is pressed a function sends the text to the encrypt class where go through the encrypt method