Girls’ Programming Network

2017

Sassy Security Chatbots!

This project was created by GPN Australia for GPN sites all around Australia!

This workbook and related materials were created by tutors at:
GPN Sydney and GPN Canberra

Girls' Programming Network - Sydney

Girls' Programming Network - Canberra

If you see any of the following tutors don't forget to thank them!!

Writers

Renee Noble
Courtney Ross
Jasmine Wong
Christie Boyd
Kelly Stewart
Kassandra Di Bona

Testers

Wendy Dourado
Sarah Tabassum
Natalie Bartolo
Sondarya Krishnan
Krithi Suresh
Xian Lim
Jennifer Ho

Sassy Security Chatbot

Keeping your secrets safe is just plain hard! Especially if you have nosey people in your family who won't stop snooping around!

Let's make a security chatbot that will keep our secrets safe and trick any snoops lurking around! Only people with the real password will be able to access the secrets!

Part 0: Getting set up!

Goal: Let's get our file and our ideas set up before we start!

Task 0.1: Set Up the File

Create a file where we are going to write the code for our game.

1) In your Python 3 IDLE click ***File*** and create a ***New File***.
2) Start by saving your file and calling it ***chatbot.py***

Now when you want to run your code, just click ***Run*** (or press ***F5*** on the keyboard)! Great! Now we're ready to code!

Task 0.2: Fun Themes

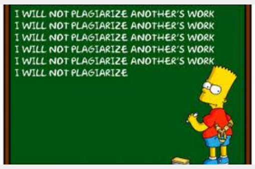
In the instructions, we're going to base it on keeping a secret diary safe from our little siblings! **But you can add your own theme!!**

Think about a theme you want to add to your chatbot! Some ideas are below!!

A spy theme!!

The chatbot protects the secret identities and speaks like a secret agent.

A secret club for a school project!!


I WILL NOT PLAGIARIZE ANOTHER'S WORK
I WILL NOT PLAGIARIZE ANOTHER'S WORK
I WILL NOT PLAGIARIZE ANOTHER'S WORK
I WILL NOT PLAGIARIZE ANOTHER'S WORK
I WILL NOT PLAGIARIZE ANOTHER'S WORK
I WILL NOT PLAGIARIZE

Keep copycats at bay and teach them a lesson for trying to steal your ideas! Teach cheaters a lesson!

Pokemon Go!!

Use your chatbot to store the secret locations of rare pokemon!

A fight between Good and Evil!

Every team has to store their secret info somewhere whether they are in Dumbledore's Army, The Rebel Alliance or the Powerpuff Girls!

Part 1: Welcome to the secret club

To start our chatbot, let’s welcome the user to the secret club.

Goal: Print a welcome message.

Task 1.1: Welcome

In your file ***chatbot.py*** write some code that will `print` out an introduction message. You can make it say whatever you would like!

For example, make it say:

```
Welcome to the secret club!
Let's find out if you're cool enough to be here…
```

Hint: *To do this, we'll need to use `print`.*

★Bonus 1.2: Wait a second!★

Let's make our chatbot a bit more human. We can make it wait a bit between responses, so it looks like it's thinking.

Step 1) At the top of your file write `import time`

This will let us use what we need to use to make our program sleep for a few seconds

Step 2) Before any `print`, add `time.sleep(3)`

This will make our program 'sleep' for 3 seconds! You can adjust it to any time you want. Try adding different delays between different responses.

Part 2: Who’s there?

Our chatbot now needs to know who it’s talking to. Addressing the user by name will make the bot seem more human!

Goal: Ask the user for their name and greet them, with sass!

Task 2.1: Ask the user for their name and store it in a variable!

It should look something like:

```
    Please enter your username:
```

Hint: *Use `input` to ask for their name.*

Task 2.2: Print a personalised welcome message

It should look something like:

```
    Please enter your username: Jasmine
    Well, well, well, it’s Jasmine
```

Hint: *Remember how we added strings together? Use it to add on the user’s name*

★BONUS 2.3: A Sassier Welcome★

Add some extra sass to your welcome to the user. It should look something like:

```
    Please enter your username: Jasmine
    Well, well, well, it’s Jasmine. We’ll see if you are worthy...
```

Hint: *Add some more strings together*

Part 3: Password Please!

To keep our server secure we need a **secret password** to give to authorised members.
To trick snoops, we'll also have a **decoy password** to give them trick information.

Goal: Check if the user has the real password, the decoy password or a password that is just wrong. Give them the information they are allowed to access.

Task 3.1: Make up some passwords!

Step 1) Create a variable called `real_password` and store the real password in there
Step 2) Create another variable called `decoy_password` and store the decoy password

Hint: Some password ideas are…. `"opensesame"` or `"PlEaSeOpEn"`

Task 3.2: Password Please...

Step 1) Ask the user for their password, and store it in a variable called `user_password`

It should look something like:

```
What's the secret password: opensesame
```

Hint: *remember how we ask the user questions?*

Task 3.3: Is the Password Correct?

Check if they got the correct password!

Step 1) Add an *`if`* statement that checks If the password matches the real password,
Step 2) If it does match, let's show them some real secret information inside the if statement

It should look something like if the real password is `"opensesame"`:

```
What's the secret password: opensesame
Looks like you pass, welcome to the secret club!
Secret club documents in here!
```

Task 3.4: Imposter Alert

If the password is anything other than the real password, tell them they're an imposter!
Hint: *Add an* `else` *statement to your* `if` *statement*

It should look something like:

```
    What's the secret password: password1234
    Well, Well, Well, I see we have an imposter here!
```

Task 3.3: Decoy Password Trick!

If the password matches the <u>decoy password</u>, let's show them some <u>trick information</u>!

It should look something like if the decoy password is `"mybirthday"`

```
    What's the secret password: mybirthday
    Well here's the link to the secret club information:
    https://sites.google.com/site/girlsprogrammingnetwork/
    It would have just been easier to google it :P
```

Hint: *Add an* `elif` *statement to your* `if` *statement , before your* `else` *statement*

★Bonus 3.4: Add a Personal Touch★

Make the text you print out more personal! Mention their name, maybe give them a cool nickname based on the first letter of their name!

It could look something like

```
    Please enter your username: Jasmine
    Well well well, it's Jasmine
    What's the secret password: mybirthday
    Well J-Swizzle here's the link to the secret club information:
    https://sites.google.com/site/girlsprogrammingnetwork/
    It would have just been easier to google it :P
```

Or even

```
    Please enter your username: Maddie
    Well well well, it's Maddie
    What's the secret password: opensesame
    Welcome Agent M, glad you have joined the fight against evil!
```

Part 4: Wrong Password! Try again!

So the users put in a password! But it’s wrong! Let’s give the user another chance!

Goal: Allow the user to try and enter the password again if they get it wrong!

Task 4.1: Try again!

Step 1) Find the spot in your code that is after you ask the user for their password, but before you check if the password is correct

Step 2) Add a while loop that keeps running while the user’s password doesn’t equal the real secret password

Step 3) Inside the while loop print a message to the user telling them they got the wrong password

Step 4) Next Inside the while loop ask the user to `input` the secret password again

This would look something like this, if the secret password was “`opensesame`”

```
What's the secret password: open
Well it looks like someone's forgetful :P
Try again, what's the secret password: pLea5e0P3N
Well it looks like someone's forgetful :P
Try again, what's the secret password: opensesame
Looks like you pass, welcome to the secret club!
Secret club documents in here!
```

Task 4.2: Don’t forget the decoy!

But we want the loop to stop if we put in the **decoy password** too, not only the secret password!

Step 1) Add to the condition of you while loop, keep going while….

The user password doesn’t equal the secret password
and
The user password doesn’t equal the decoy password

Hint: *use and to join the two conditions together*

★BONUS 4.3: Too many tries!★

We can’t let the user try and guess the password forever, give them maximum 3 tries!

Step 1) Before the loop, create a variable that counts how many guesses the user has had. Set it to 1.

Step 2) In the loop, update the counter after the user has another guess.

Step 3) Update the while loop condition *to check:*
the Secret password *and* *the decoy password* *and* *that the* *count is less than 3*

Bonus) Tell the user how many tried they have left each time you ask for the password