

Yodabot in Python – Day 1



What do you need to get started?

1. Python (3.3.+): <https://www.python.org/downloads/>
2. PyCharm: <http://www.jetbrains.com/pycharm/download/>
3. Install pip,virtual env
`curl -O http://python-distribute.org/distribute_setup.py
sudo python distribute_setup.py
sudo easy_install pip
sudo pip install virtualenv
sudo pip install virtualenvwrapper`

(Make sure you download this before the class and you have right version for your platform, Windows or Mac)

Yodabot in Python



1. Code Download:

<https://github.com/zacharythomasabraham/workshops-yodabot>

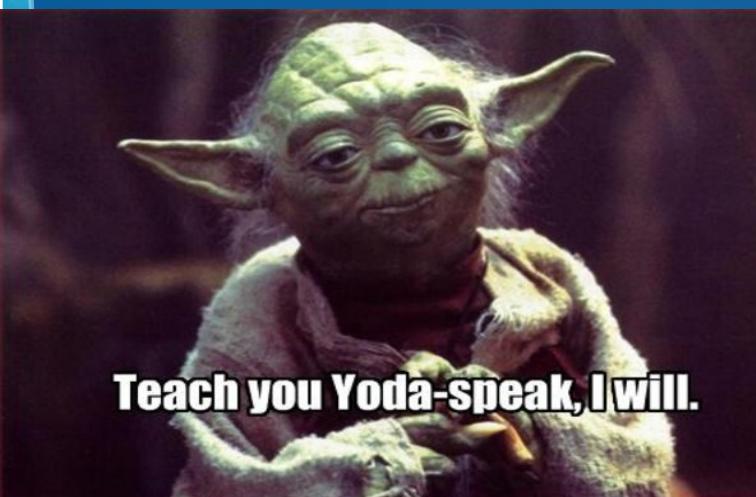
(Make sure you download this before the class and you have right version for your platform, Windows or Mac)

Zachary Abraham

- 10th grade Aragon High School ,San Mateo
- Passionate about Computer Science and Robotics

What are we going to do today ?

1. *Create a slack account on www.slack.com*
2. *Create a yodabot*
3. *Create a python server*
4. *YodaSpeak !!!*



Teach you Yoda-speak, I will.

Generate the slack api token

- First you need to get the slack api token for your bot. You have two options:
- If you use a [bot user integration of slack, you can get the api token on the integration page.](#)
- If you use a real slack user, you can generate an api token on [slack web api page.](#)

Generate the slack api token

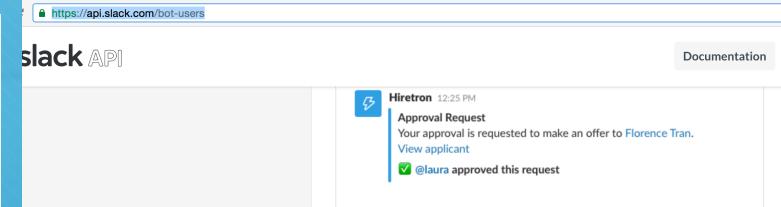
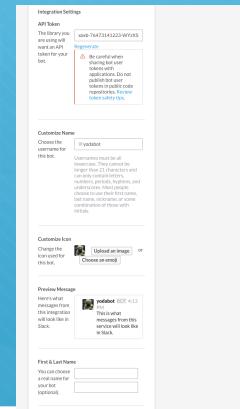
- First you need to get the slack api token for your bot. You have two options:
- If you use a [bot user integration of slack, you can get the api token on the integration page.](#)
- If you use a real slack user, you can generate an api token on [slack web api page.](#)

Generate the slack api token

- First you need to get the slack api token for your bot. You have two options:
- If you use a [bot user integration of slack, you can get the api token on the integration page.](#)
- If you use a real slack user, you can generate an api token on [slack web api page.](#)

Create a Yodabot

- On <https://api.slack.com/bot-users> select new bot integration link to use slack RTM api
- Give your bot a name (Yodabot1 or Yodabot[number] and picture from internet
- Copy api token
- SAVE



Setting up your bot user

After you've figured out what you want your bot user to do and have an idea of how you'll go about implementing it, you'll want to prepare Slack for the arrival of your bot user.

How do I create custom bot users for my team?

Start by creating a [new bot user integration](#). You'll need to pick a username for your new bot. Bot usernames can be up to 21 characters long and sorry, you can't name your bot [Slackbot](#). Additionally, your bot username cannot be the same name as one of yo

Integration Settings

API Token

The library you are using will want an API token for your bot.

xoxb-76473141223-WYzXS:

[Regenerate](#)



Be careful when sharing bot user tokens with applications. Do not publish bot user tokens in public code repositories. [Review token safety tips.](#)

Customize Name

Choose the username for this bot.

@yodabot

Usernames must be all lowercase. They cannot be longer than 21 characters and can only contain letters, numbers, periods, hyphens, and underscores. Most people choose to use their first name, last name, nickname, or some combination of those with initials.

Customize Icon

Change the icon used for this bot.



[Upload an image](#)

or

[Choose an emoji](#)

Preview Message

Here's what messages from this integration will look like in Slack.



yodabot BOT 4:13

PM

This is what messages from this service will look like in Slack.

First & Last Name

You can choose a real name for your bot (optional).

Invite your yodabot to channel yodaspeak

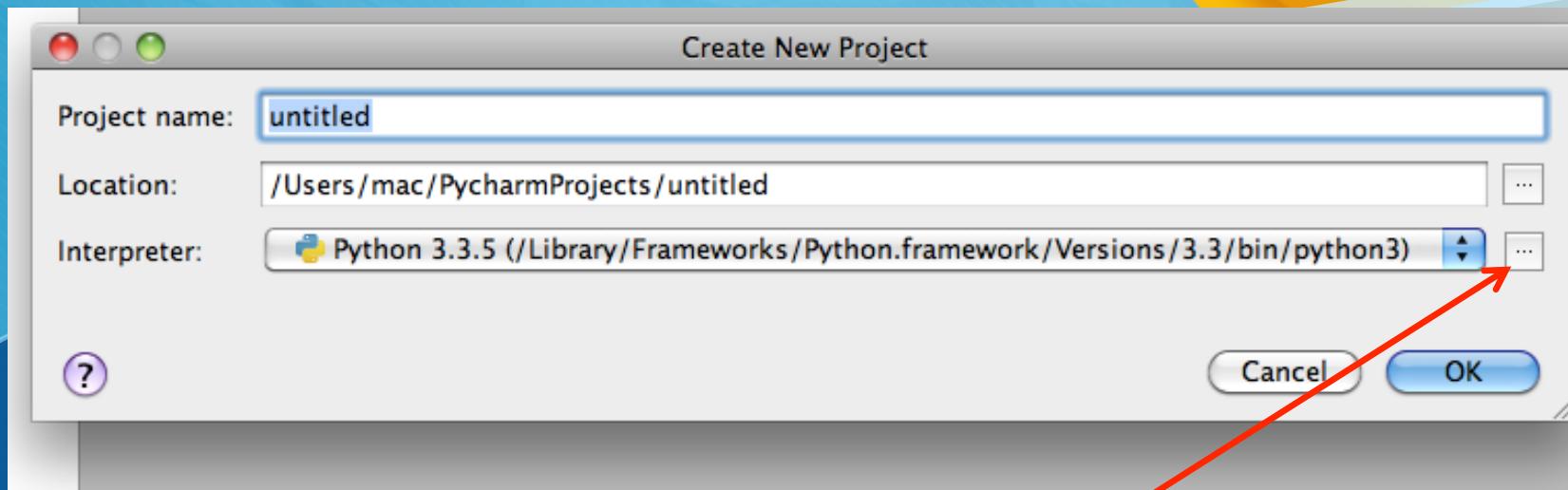
A screenshot of a Slack interface. At the top, a yellow banner displays the message "Slack needs your permission to enable desktop notifications." Below the banner, the channel header "#yodaspeak" is shown, indicating it has 2 members and an option to "Add a topic". The left sidebar shows a list of channels: general, jokes, random, and #yodaspeak, which is highlighted with a green bar. A list of recent messages shows messages from slackbot, sachary (you), hatterbot, yodabot, and invite people. The main channel view for "#yodaspeak" shows the channel name at the top, followed by the message "You created this channel today. This is the very beginning of the #yodaspeak channel. Purpose: speak yoda (edit)". Below this, there are links to "+ Add an app or custom integration" and "& Invite others to this channel". On the right side, a context menu is open over the channel name, listing options: "Jump to date ...", "Invite team members to join ...", "View channel details", "Additional options ...", "Notification preferences ...", "Mute #yodaspeak", "Add an app or integration", and "Leave #yodaspeak". The "Invite team members to join ..." option is highlighted with a blue box.

Create Project Yodabot(Server)

- Virtualenv yodabot
 - Creates a folder yodabot
- source yodabot/bin/activate
- Pip install slackclient
- Open project Yodabot on Pycharm

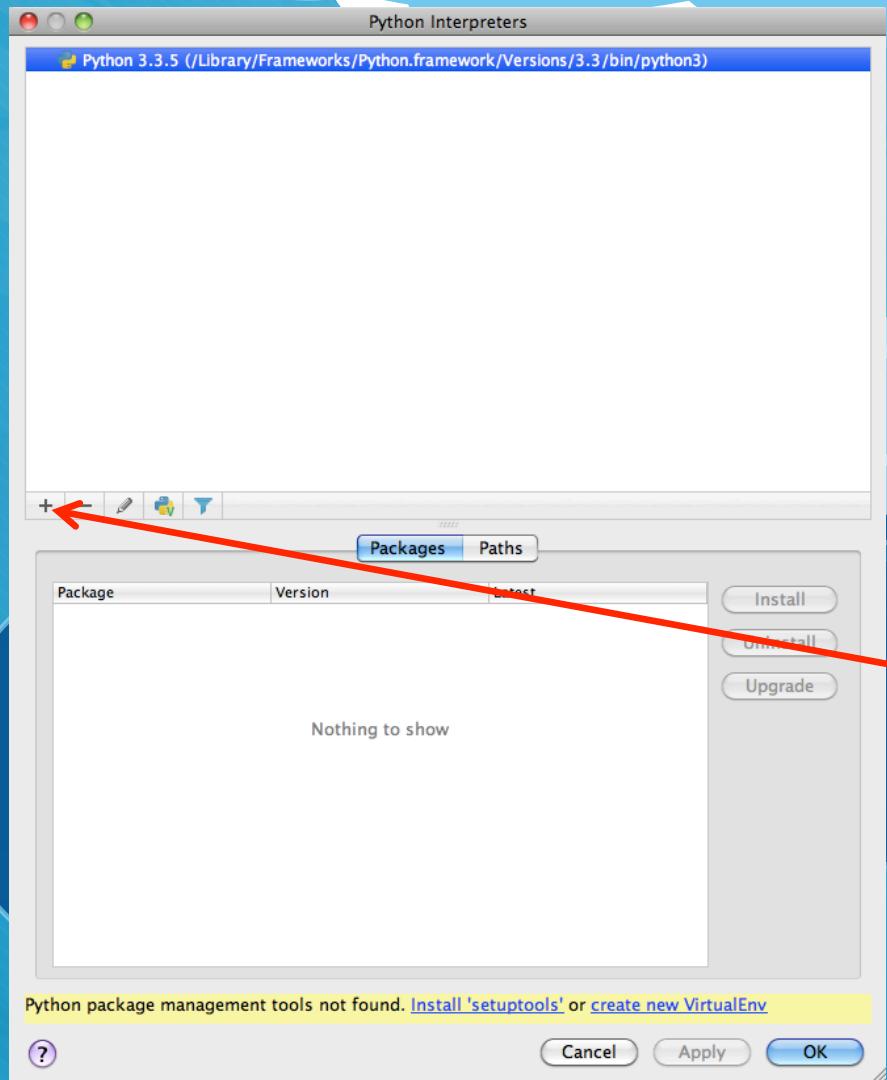
Python Setup:

Create New Project in PyCharm called yodabot



Click here to choose your interpreter

Python Setup:



Click on + sign to choose interpreter

The screenshot shows the PyCharm IDE interface. On the left, the project navigation bar displays a project named "yodabot". A context menu is open over the project name, listing options like "New Project...", "Save As...", "Open Recent", and "Close Project". Below these, under the "File" section, are "Default Settings...", "Import Settings...", "Export Settings...", and "Settings Repository...". Further down are "Save All", "Synchronize", and "Invalidate Caches / Restart...". The "File" menu also includes "Export to HTML...", "Print...", "Add to Favorites", "File Encoding", "Line Separators", "Make File Read-only", and "Power Save Mode".

The main workspace shows a file named "display_bot_id.py" with the following content:

```
import os
from slackclient import SlackClient

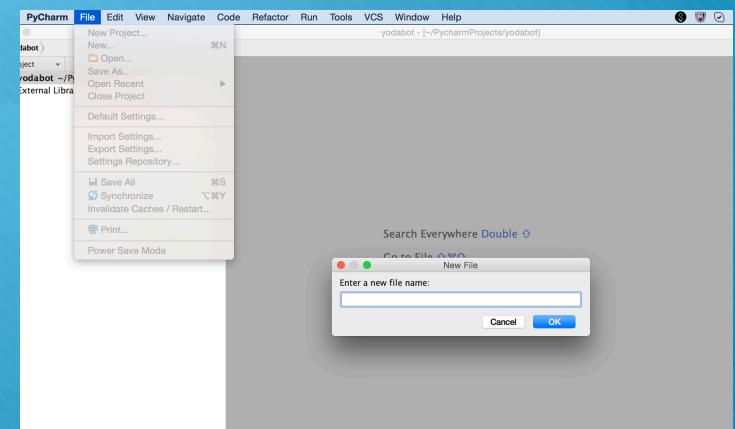
BOT_NAME = 'yodabot'

slack_client = SlackClient(os.environ.get('SLACK_BOT_TOKEN'))

if __name__ == "__main__":
    api_call = slack_client.api_call("users.list")
    if api_call.get('ok'):
        # retrieve all users so we can find our bot
        users = api_call.get('members')
        for user in users:
            if 'name' in user and user.get('name') == BOT_NAME:
                print("Bot ID for " + user['name'] + " is " + user.get('id'))
    else:
        print("could not find bot user with the name " + BOT_NAME)
```

Bot Id

- We have api token from slack bot settings page
- We need Yodabot ID
- Create New Python File – display_bot_id



Edit Configuration

- Run ->Edit Configuration →Environment Variables
- Click + sign and enter key “SLACK_BOT_TOKEN”
- Give value “your bot api token id” from slack settings

Integration Settings

API Token

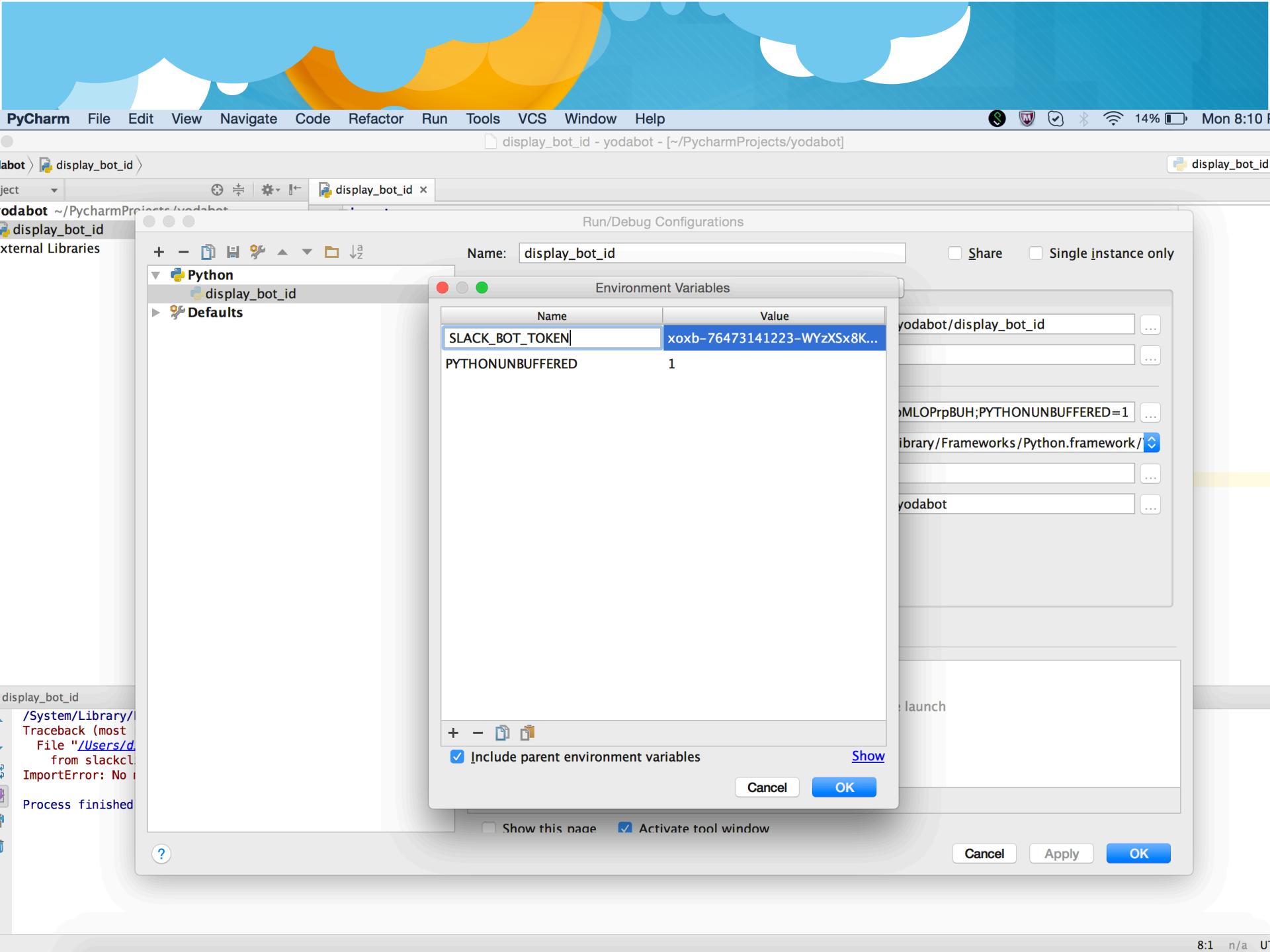
The library you are using will want an API token for your bot.

xoxb-76473141223-WYzXSx8KntXVZrpMLOPrpBUH

Regenerate

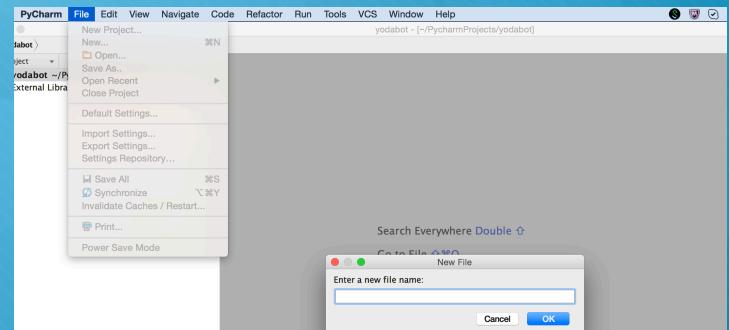


Be careful when sharing bot user tokens with applications. Do not publish bot user tokens in public code repositories. [Review token safety tips](#).



To connect to yodabot we need

- o Api token – settings page
- o Yoda bot id ?



Bot Id

The image shows a screenshot of the PyCharm IDE running on a Mac OS X desktop. The title bar indicates the project is named 'yodabot' and the file being edited is 'display_bot_id'. The code editor displays a Python script that uses the SlackClient library to find a bot's ID. The script imports os and SlackClient, defines a BOT_NAME constant, and retrieves all users via the Slack API to find the bot's ID. The output of the run command in the bottom terminal shows that the bot ID for 'yodabot' is U28DX456K.

```
PyCharm File Edit View Navigate Code Refactor Run Tools VCS Window Help
display_bot_id - yodabot - [~/PycharmProjects/yodabot]
yodabot > display_bot_id
Project ~/PycharmProjects/yodabot
yodabot
display_bot_id
External Libraries
display_bot_id
import os
from slackclient import SlackClient

BOT_NAME = 'yodabot'

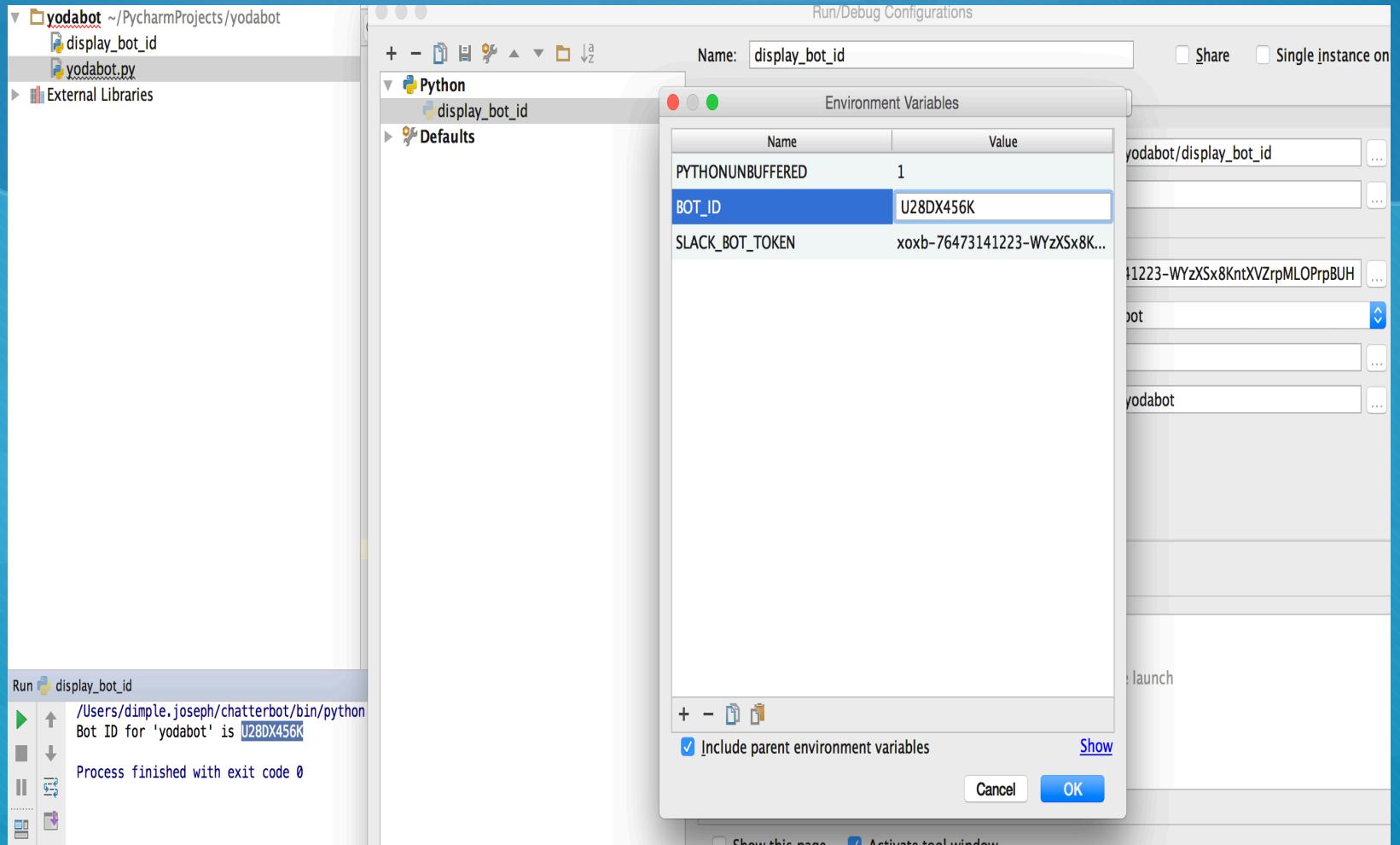
slack_client = SlackClient(os.environ.get('SLACK_BOT_TOKEN'))

if __name__ == "__main__":
    api_call = slack_client.api_call("users.list")
    if api_call.get('ok'):
        # retrieve all users so we can find our bot
        users = api_call.get('members')
        for user in users:
            if 'name' in user and user.get('name') == BOT_NAME:
                print("Bot ID for '" + user['name'] + "' is " + user.get('id'))
    else:
        print("could not find bot user with the name " + BOT_NAME)

Run display_bot_id
/Users/dimple.joseph/chatterbot/bin/python /Users/dimple.joseph/PycharmProjects/yodabot/display_bot_id
Bot ID for 'yodabot' is U28DX456K
Process finished with exit code 0
```

Create New Python File

- Yodabot
- Add environment variables BOT_ID and SLACK_API_TOKEN
- Run → Edit Configuration → Environment Variable -> +
- Run → Edit Configuration → Environment Variable -> +
- Add BOT_ID and SLACK_API_TOKEN



Lets Test

- Go to channel YodaSpeak -
<https://chatterbotgroup.slack.com/messages/yodaspeak/>
- @yodabot:How are you?

CHANNELS (4)

general

jokes

random

yodaspeak

DIRECT MESSAGES (4)

slackbot

zachary (you)

chatterbot

yodabot

+ Invite people

#yodaspeak

You created this channel today. This is the very beginning of the [#yodaspeak](#) channel. Purpose:
[speak yoda \(edit\)](#)

+ Add an app or custom integration Invite others to this channel

Today**zachary** 7:39 PMjoined #yodaspeak, and invited [@yodabot](#)**zachary** 7:39 PM

set the channel purpose: speak yoda

**zachary** 8:42 PM

YodaSpeak!!

