

# Python

## INTERMEDIATE

Rapid prototyping using Python libraries and integration with local and remote services



POLITECNICO  
DI TORINO



# Text To Speech

- Write a program to make the computer «speak» text strings given by users on the console
- The program must ask for new words until the word «exit» is entered, in such a case it should greet the user by saying «Goodbye»

# Problems

- Converting a text into an audio file
- Playing an audio file

# Problems

- Converting a text into an audio file
- Playing an audio file



...others?



...others?

# Google TTS

- Unofficial API
  - Warning: subject to changes, and may block you if you abuse it
- [http://translate.google.com/translate\\_tts?t1=en&q=hello&client=tw-ob](http://translate.google.com/translate_tts?t1=en&q=hello&client=tw-ob)
- Language: t1=en
- Test to convert: q=hello
- Trick to avoid captcha (maybe): client=tw-ob

# Voice RSS

- <http://www.voicerss.org/>
- Free up to 350 daily requests
- Needs an API key
- `http://api.voicerss.org/?key=abc012345&src=hello&hl=en-us`
- Language: `hl=en-us`
- Test to convert: `src=hello`
- Application key (free): `key=abc012345`

# How to call an external URL?

- Combination of tools
  - wget unix command line tool
  - Python os package to access shell commands
    - `os.system()`
- Dedicated python packages
  - urllib package
  - wget package

# Mplayer

```
fcorno@ami-guest: ~/PycharmProjects/python-intermediate
MPlayer(1)                               The Movie Player                               MPlayer(1)

NAME
  mplayer - movie player
  mencoder - movie encoder

SYNOPSIS
  mplayer [options] [file|URL|playlist|-]
  mplayer [options] file1 [specific options] [file2] [specific options]
  mplayer [options] {group of files and options} [group-specific options]
  mplayer [br]://[title][/]device [options]
  mplayer [dvd|dvdnav]://[title|[start_title]-end_title][/]device
  [options]
  mplayer vcd://track[/device]
  mplayer tv://[channel][/]input_id [options]
  mplayer radio://[channel|frequency][/]capture [options]
  mplayer pvr:// [options]
  mplayer dvb://[card_number@]channel [options]
  mplayer mf://[filemask|@listfile] [-mf options] [options]
  mplayer [cdda|cddb]://track[-endtrack][:speed][/]device [options]
  mplayer cue://file[:track] [options]
  mplayer
  [file|mms[t]|http|http_proxy|rt[s]p|ftp|udp|unsv|icyx|noicyx|smb]://

Manual page mplayer(1) line 1 (press h for help or q to quit)
```



# Voice mailbox

- Write a program that
  - Monitors your inbox folder for unseen messages
  - Provides a vocal summary of newly received messages including the message count, and for each message the sender(s) and the title

# Problems

- Reading e-mail protocols: POP, IMAP, custom APIs
  - Preferred: IMAP
  - Python package: imaplib
- Need to parse e-mail structure
  - Python package: email
  - Understands RFC 2822
- Combined with the previous TTS exercise

# Twitter speak

- Write a program that
  - Asks for a twitter hashtag
  - Searches twitter for new messages with that tag
  - Prints and read aloud the message content
- Question: Python twitter libraries?

# Twitter?

<http://mike.verdone.ca/twitter/>

**ptt** about why? install screenshots downloads development

python twitter tools

CHECK THIS OUT! LOL LUNCH? WE'RE MEETING AT 8

lean. sweet. tweet.

Download

### What is it?

Python Twitter Tools (PTT) includes a **Twitter API**, **command-line tool**, and **IRC bot**. It is developed by Mike Verdone and the Python Twitter Tools developer team.

The Minimalist Twitter API for Python is a Python API for **Twitter**, everyone's favorite Web 2.0 Facebook-style status updater for people on the go. Also included is a **twitter command-line tool** for getting your friends' tweets and setting your own tweet from the safety and security of your favorite shell. Also included is an IRC bot that can announce Twitter updates to an IRC channel.

### Why PTT is better than X?

The other two Python Twitter APIs in PyPi pale in comparison to PTT because...

- It's minimalist. Small. Simple. Don't you like simple?
- Its methods are directly bound to the Twitter API's URLs. For instance `twitter.statuses.friends.timeline()` will always hit the URL `http://twitter.com/statuses/friends.timeline.json`. Always. When Twitter decides to change them you won't

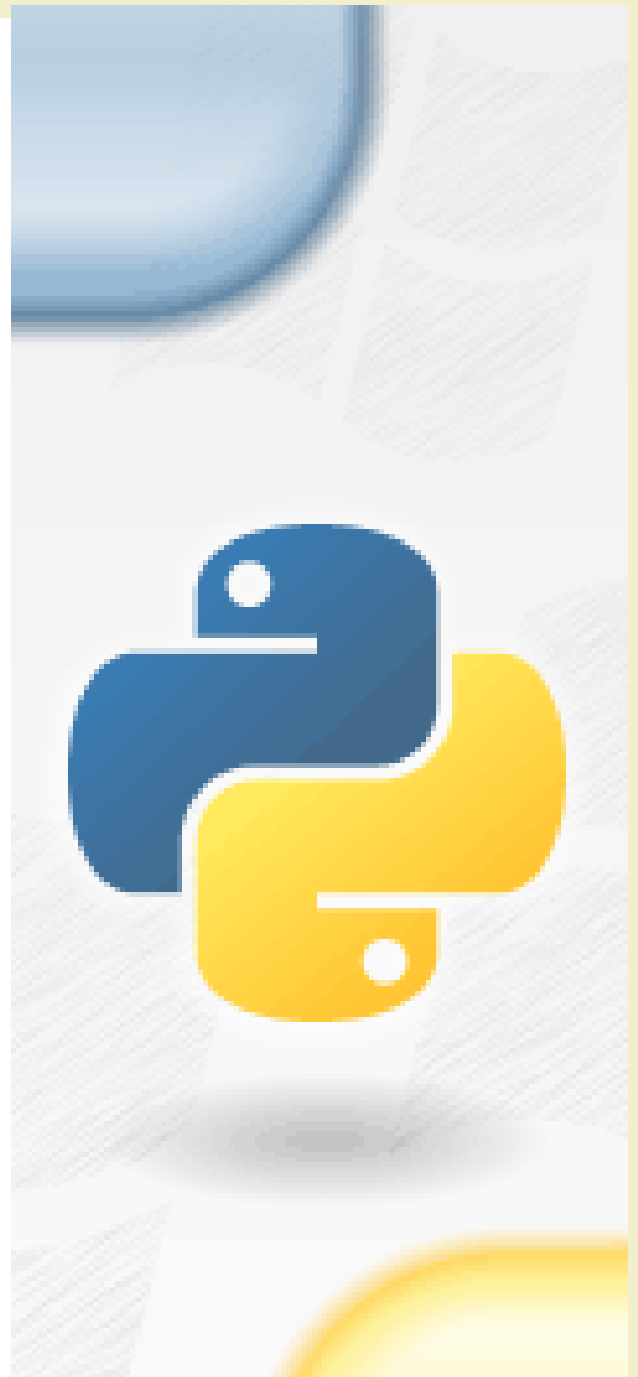
<https://github.com/sixohsix/twitter>

# Questions?

01PRD AMBIENT INTELLIGENCE: TECHNOLOGY AND DESIGN

Fulvio Corno

[fulvio.corno@polito.it](mailto:fulvio.corno@polito.it)



# License

- This work is licensed under the Creative Commons “Attribution-NonCommercial-ShareAlike Unported (CC BY-NC-SA 3,0)” License.
- You are free:
  - to **Share** - to copy, distribute and transmit the work
  - to **Remix** - to adapt the work
- Under the following conditions:
  - **Attribution** - You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).
  - **Noncommercial** - You may not use this work for commercial purposes.
  - **Share Alike** - If you alter, transform, or build upon this work, you may distribute the resulting work only under the same or similar license to this one.
- To view a copy of this license, visit <http://creativecommons.org/license/by-nc-sa/3.0/>