

컴퓨터네트워크

- 10. Multimedia-

제 출 일	2018.11.24
학 과	컴퓨터공학과
학 번	201402391
이 름	이 병 만

※ 과제 목표 및 해결 방법

· 과제 목표

- EBS, KBS 라디오 자동 녹음 서비스 구현
- 웹 서버에 mp3 다운로드 서비스 구현

· 해결 방법

```
import subprocess as sub
import os
import mutagen
import mutagen.mp3
from mutagen.easyid3 import EasyID3
import datetime
import time
import signal
import glob
```

[그림1]

➔ 구현에 필요한 모듈을 import한다.

```
# Edit mp3 tag
def edit_mp3_tag(transfer_file):
    try:
        meta = EasyID3(transfer_file)
    except mutagen.id3.ID3NoHeaderError:
        meta = mutagen.File(transfer_file, easy=True)
        meta.add_tags()

    date = datetime.date.today()
    meta['title'] = str(date) + "_leebyeongman"
    meta['artist'] = "201402391"
    meta.save()
```

[그림2]

➔ mp3 태그를 수정하는 함수이다. 인자로 수정 할 파일의 경로와 이름을 받는다. title은 현재 날짜와 이름으로 작성하였고, artist는 학번으로 작성하도록 하였다.

```

def kbs_download():
    now = datetime.datetime.now()
    date = str(now.year)+str(now.month)+str(now.day)+str(now.hour)+str(now.minute)+str(now.second)

    # variables
    original_file = './original_path/' + date + '_KBS' + '.flv'
    transfer_file = './transfer_path/' + date + '_KBS' + '.mp3'
    cmd = 'mplayer -ao pcm:file=' + original_file + ' $(curl -s "http://onair.kbs.co.kr/index.html?pname=onair" | grep service_url | tail -1 | cut -d\\\" -f16 | cut -d\\\" -f1) -vc dummy -vo null'

    # KBS radio download
    proc = sub.Popen(cmd, shell=True, preexec_fn=os.setsid)
    time.sleep(30)
    os.killpg(os.getpgid(proc.pid), signal.SIGTERM)

    # transfer flv to mp3
    proc = sub.Popen(["ffmpeg", "-i", original_file, "-acodec", "mp3", transfer_file], stdout = sub.PIPE)
    proc.communicate()

    # Edit_exeute
    edit_mp3_tag(transfer_file)

```

[그림3]

➔ kbs의 라디오를 다운로드하는 함수이다. 먼저 현재 시간을 저장해서 date 변수에 문자열로 저장한다. 이것은 파일명에 사용하기 위함이다. 변수로 flv 파일이 저장되는 original_file, mp3가 저장되는 transfer_file로 경로와 함께 저장해준다. 라디오를 실시간 녹음을 위한 명령어를 설정해주고 subprocess로 실행한다. 이 때 녹음은 30초 동안 진행하였다. 30초 후 프로세스를 죽이고 flv 파일을 mp3로 변환해주었다. 그 다음 태그를 수정하였다.

```

def ebs_download():
    now = datetime.datetime.now()
    date = str(now.year)+str(now.month)+str(now.day)+str(now.hour)+str(now.minute)+str(now.second)
    ip = "rtmp://58.229.187.11/iradio/iradiolive_m4a"
    original_file = './original_path/' + date + '_EBS' + '.flv'
    transfer_file = './transfer_path/' + date + '_EBS' + '.mp3'

    # EBS radio download
    proc = sub.Popen(["rtmpdump", "-r", ip, "-B", "10", "-o", original_file], stdout=sub.PIPE)
    proc.communicate()

    # Transfer flv to mp3
    proc = sub.Popen(["ffmpeg", "-i", original_file, "-acodec", "mp3", transfer_file], stdout=sub.PIPE)
    proc.communicate()

    # Edit_exeute
    edit_mp3_tag(transfer_file)

```

[그림4]

➔ ebs의 라디오를 다운로드하는 함수이다. 먼저 현재 시간을 저장해서 date 변수에 문자열로 저장한다. 이것은 파일명에 사용하기 위함이다. ebs라디오가 진행중인 ip를 저장하였고 변수로 flv 파일이 저장되는 original_file, mp3가 저장되는 transfer_file로 경로와 함께 저장해준다. 라디오를 실시간 녹음을 위한 명령어를 설정해주고 subprocess로 실행한다. 이 때 녹음은 -B를 사용해서 10초 동안 진행하였다. 녹음이 끝나고 flv 파일을 mp3로 변환해주었다. 그 다음 태그를 수정하였다.

```
def create_html():
    f = open('download.html', 'w')

    f.write('<html>\n')
    f.write('    <head>\n')
    f.write('        <meta charset="utf-8">\n')
    f.write('        <title>Download</title>\n')
    f.write('    </head>\n')

    f.write('    <body>\n')
    kbs_file = glob.glob('/home/leebyeongman/CN_201402391/10/transfer_path/*_KBS.mp3')
    ebs_file = glob.glob('/home/leebyeongman/CN_201402391/10/transfer_path/*_EBS.mp3')
    print(kbs_file)
    print(ebs_file)
    f.write('    <h2>KBS_RADIO_List</h2>\n')
    j = 1
    for i in kbs_file:
        f.write('        <a href="'+i.split('/')[5]+'/'+i.split('/')[6]+' download">'+str(j)+'</a>\n')
        j += 1

    f.write('    <h2>EBS_RADIO_List</h2>\n')
    j = 1
    for i in ebs_file:
        f.write('        <a href="'+i.split('/')[5]+'/'+i.split('/')[6]+' download">'+str(j)+'</a>\n')
        j += 1

    f.write('    </body>\n')
    f.write('</html>\n')
```

[그림5]

➔ html을 생성하는 함수이다. 기본적인 html형식에 맞게 작성하였고 glob를 사용해서 해당 경로에 있는 파일들을 가져왔다. 그래서 kbs, ebs파일을 나누어서 리스트에 저장하였고 각 분류해서 a태그를 사용해서 화면에 뿌려주었다.

※ 실행 결과

