U5NMSURAQ : `s///g` or `s|||g` or any symbol instead of `|` is how a basic replacing expression is constructed.The first field is what to match, the second is what to replace it with.

U5NMSURAQ : `(.+)/[^/]+/([^/]+)/.+``(.+)/` matches anything from the start until the first `/` and puts found characters in the first group (`\1`)

U5NMSURAQ : `[^/]+/` matches anything that is not a slash, and then a slash (`vag/` or `home/`)
U5NMSURAQ : `([^/]+)/` matches the same thing, but puts the stuff found in-between slashes in the second group `\2`
U5NMSURAQ : and then `.+` matches whatever comes next to the end of line
U5NMSURAQ : and the second field tells sed to replace the line with `\1\2`, so our saved groups side-by-side: the first group was everything before the first slash, and the second group was the stuff between 2nd and 3rd slashes
U6BM4GY0G : Ah ok, thanks a lot for the explanation!
U4533716J : I just whizzed through some boto3 S3 scripts. Now I'm working on a boto3 route53 script, and I can't for the life of me figure out what's going wrong. On top of that I'm not finding much good documentation. Can someone point me to some good example scripts or documentation? I just want to start by figuring out how to get it to list my Zones, and I think from there it will click.
U13L8J76J : <@U4533716J> s3 works globally. It doesn't even have the concept of regions, not mentioning availability zones.
U13L8J76J : What exactly you're trying to do and what exactly fails?
U5ZPMJA06 : <@U3UR8LD18> This little program does that:
```
inputdata = """
&gt; 1234 alphabet /vag/one/arun
&gt; 1454 bigdata /home/two/ogra
&gt; 5684 apple /vinay/three/dire
"""

for line in inputdata.splitlines():
    parts = line.strip().rsplit(" ", 1)
    if len(parts) &lt; 2: continue
    subparts = parts[1].split("/")
    print parts[0] + " " + subparts[2]
# Output is:
&gt; 1234 alphabet one
&gt; 1454 bigdata two
&gt; 5684 apple three
```

U3UR8LD18 : <@U5NMSURAQ> <@U5ZPMJA06> thanks guys
U5ZPMJA06 : But wait! I can squeeze in a regular expression: :stuck_out_tongue_winking_eye:
```
import re

inputdata = """
&gt; 1234 alphabet /vag/one/arun
&gt; 1454 bigdata /home/two/ogra
&gt; 5684 apple /vinay/three/dire
"""

transform = re.compile("(.*)/.*/(.*)/.*")

for line in inputdata.splitlines():
    m = transform.match(line)
    if not m: continue
    print "".join(m.groups())
```

U13L8J76J :
```
echo '&gt; 1234 alphabet /vag/one/arun
&gt; 1454 bigdata /home/two/ogra
&gt; 5684 apple /vinay/three/dire' | awk '{ split($4, var, /\//); $4=""; print $0 var[3];}'
&gt; 1234 alphabet one
&gt; 1454 bigdata two
&gt; 5684 apple three
```

U13L8J76J : Do not underestimate `awk`!
U5NMSURAQ : awk is cool
U13L8J76J : And its much more readable than `sed` version :wink:
U5ZPMJA06 : Yeah, but if you want to build this into an already existing Python program, a few lines of code will do.
U5ZPMJA06 : Oh wait OP mentioned two text files on his file system... :stuck_out_tongue_winking_eye:
U4533716J : <@U13L8J76J> I figured it out ... I was doing everything right, but there was a missing _ in main ... can't believe I missed that and can't believe it didn't give me an error on that
U13L8J76J : "Missing _"?
U4533716J : I'm just building up a set up scripts to do everything in AWS as practice, as well as to help create a more automated envirnment for my partner (they do dev, I do ops)