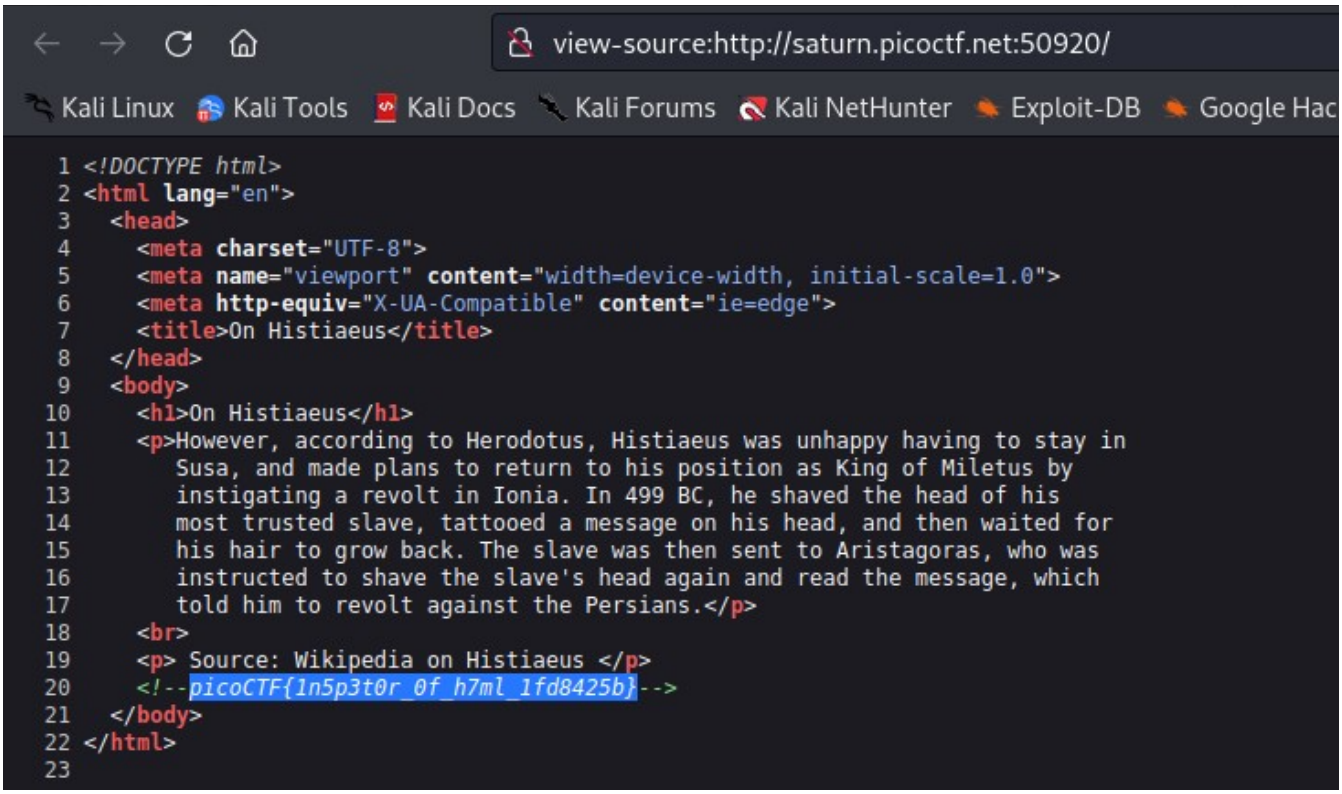
“Inspect HTML”

- open web page then show the inspect the flag appeared as comment in HTML code:



“Files types”

- Download flag file from the website picoctf.
- Try to open file but the file didn’t open.
- know the type of file by order “file” in kali.
- know that it contains bash, archive and text.

```
┌──(vampire㉿kali)-[~/Downloads]
└─$ file Flag.pdf
Flag.pdf: shell archive text
```

- Use “sh flag.pdf” in kali.

```
┌──(vampire㉿kali)-[~/Downloads]
└─$ sh Flag.pdf
x - created lock directory _sh00046.
x - extracting flag (text)
x - removed lock directory _sh00046.
```

- Then know the bash file after open it write with tool and installed this tool.
- Use order “file flag” again to check the file type the file become archive and text.

```
┌──(vampire㉿kali)-[~/Downloads]
└─$ file flag
flag: current ar archive
```

- Use many tools like cpio, rar, lzip, lz4, lzma and so on to decompress and extract file.

```
┌──(vampire㉿kali)-[~/Downloads]
└─$ ar xv flag
x - flag
```

```
┌──(vampire㉿kali)-[~/Downloads]
└─$ lzip -d flag.out
lzip: flag.out: Bad magic number (file not in lzip format).
lzip: Deleting output file 'flag.out.out', if it exists.
```

```
┌──(vampire㉿kali)-[~/Downloads]
└─$ file flag.out
flag.out: LZ4 compressed data (v1.4+)
```

- Finally file flag become text it contains a hex code.

```
└─$ file flag
flag: ASCII text
┌──(vampire㉿kali)-[~/Downloads]
└─$ cat flag
7069636f4354467b66316c656e406d335f6d406e3170756c407431306e5f
6630725f3062326375723137795f33633739633562617d0a
```

- Decrypt it by: xxd tool then get the flag.

```
File  Actions  Edit  View  Help
┌──(vampire㉿kali)-[~/Downloads]
└─$ cat flag | xxd -r -ps
picoCTF{f1len@m3_m@n1pul@t10n_f0r_0b2cur17y_3c79c5ba}
```

“enhance”

1- Download file from picoctf.
2- use gedit tool in kali linux to catch flag

```
—$ gedit drawing.flag.svg
```

3-the flag: picoCTF{3nh4nc3d_aab729dd}

```
114         style="font-size:0.00352781px;line-height:1.25;fill:#ffffff;stroke-width:0.26458332;"
115         id="tspan3764">F { 3 n h 4 n </tspan><tspan
116         sodipodi:role="line"
117         x="107.43014"
118         y="132.11588"
119         style="font-size:0.00352781px;line-height:1.25;fill:#ffffff;stroke-width:0.26458332;"
120         id="tspan3752">c 3 d _ a a b 7 2 9 d d }</tspan></text>
121   </g>
122 </svg>
```

XML ▾ Tab Width: 8 ▾ Ln 115, Col 35

“File-run1”

- Download from terminal in kali Linux: “wget https://artifacts.picoctf.net/c/308/run”
- Change the permission of file to make it executable by “chmod +x run” ⟶ (+x: make all executable file)
- Can’t open the file because it is executable file.

```
└─$ file run
run: ELF 64-bit LSB pie executable, x86-64, version 1 (SYSV), dynamically linked,
interpreter /lib64/ld-linux-x86-64.so.2, BuildID[sha1]=d21045f436f558afb1bd44f95c6
63de57f1c8926, for GNU/Linux 3.2.0, not stripped
```

- Open the file with “./run”

```
┌──(vampire㉿kali)-[~/Downloads]
└─$ ./run
The flag is: picoCTF{U51N6_Y0Ur_F1r57_F113_9bc52b6b}
```

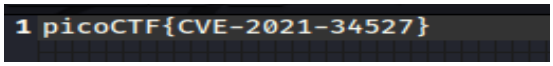
“File-run2”

- Download from terminal in kali Linux: “wget https://artifacts.picoctf.net/c/351/run”
- Change the permission of file to make it executable by “chmod +x run” ⟶ (+x: make all executable file).

- Can’t open the file because it is executable file.
- Open the file with “./run Hello!” and get the flag.

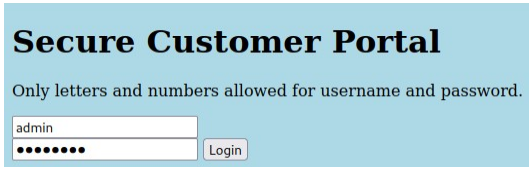
“CVE-XXXX-XXXX”

- Search on CVE in google remote code execution (RCE) vulnerability in 2021 in the windows print spooler service, find the vulnerability.
- Write the vulnerability with the same form of flag.



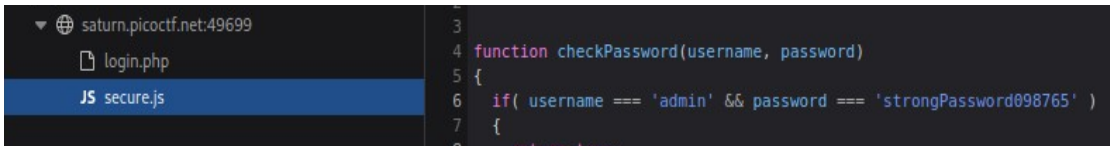
“Local Authority”

- Open website from link “http://saturn.picoctf.net:49699/” .
- Try to input any username and password like as:
  - username: admin password: pass

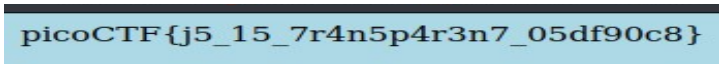


- Press on inspect (Q), then enter to debugger window show 2 file:
  - login.php
  - secure.js

- Enter to secure.js, I will be found function for check password contain to real username and password.



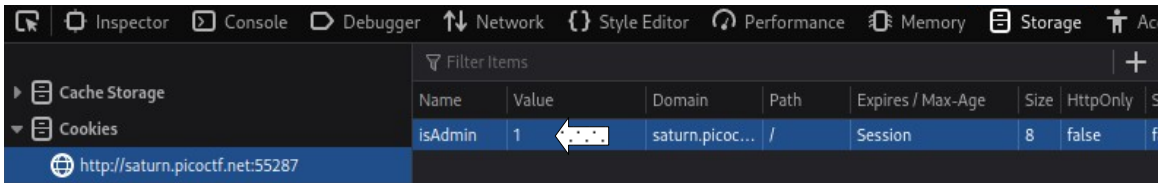
- Write username and password in login page then I will get the flag after login successful messag:
  - picoCTF{j5_15_7r4n5p4r3n7_05df90c8}



“Power Cookie”

- Open the link “http://saturn.picoctf.net:55287/” to enter website
- Click continue as guest.

- Press on inspect (Q), enter to storage window and click on cookies. Change the value of isAdmin from 0 ⟶1 as figure:



- Reload website a gain and I will get the flag:

picoCTF{gr4d3_A_c00k13_5d2505be}

picoCTF{gr4d3_A_c00k13_5d2505be}

“Forbidden Paths”

- Enter to the website through the link “http://saturn.picoctf.net:49700/”
- Enter “../../../../../flag.txt” in search bar then press to read.

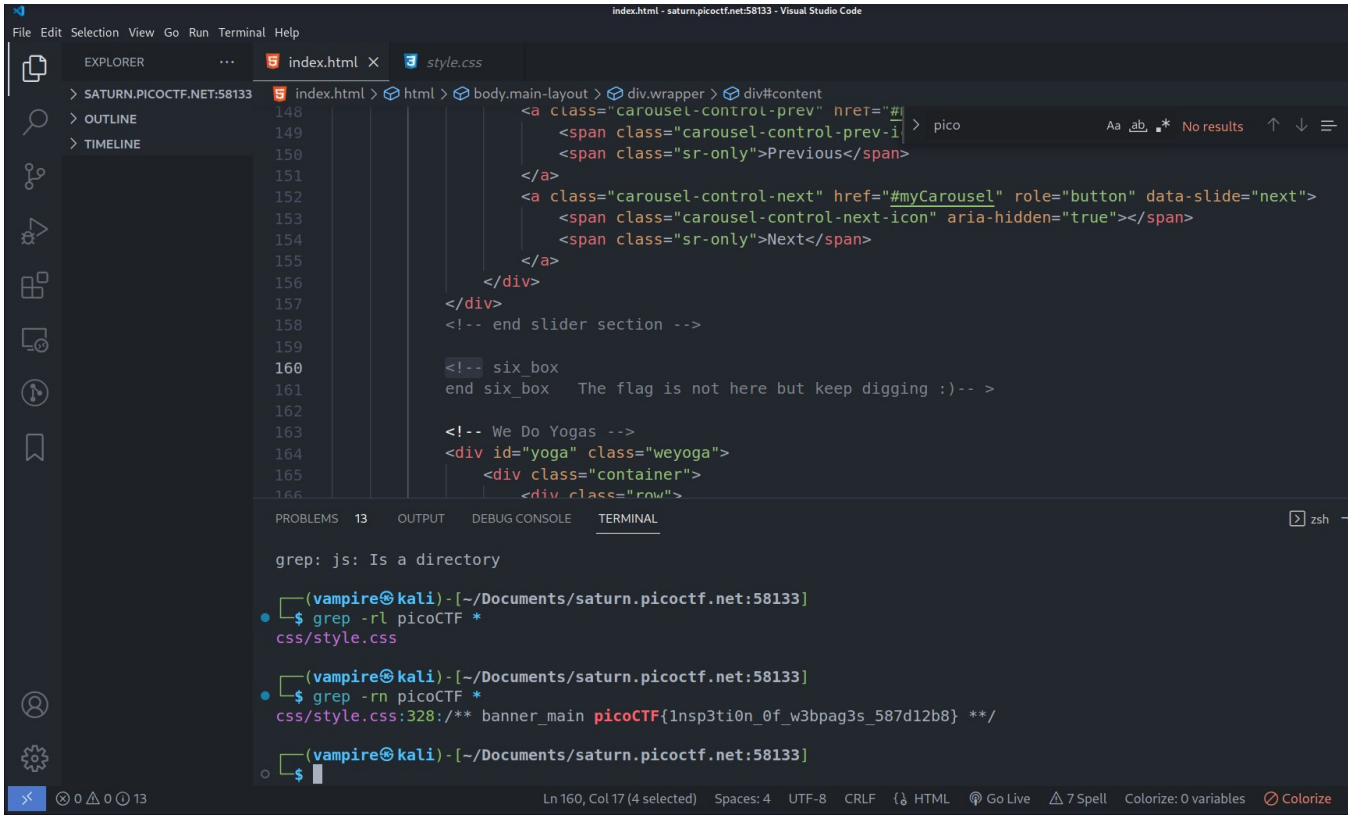
- Found the flag:
  - picoCTF{7h3_p47h_70_5ucc355_6db46514}

---

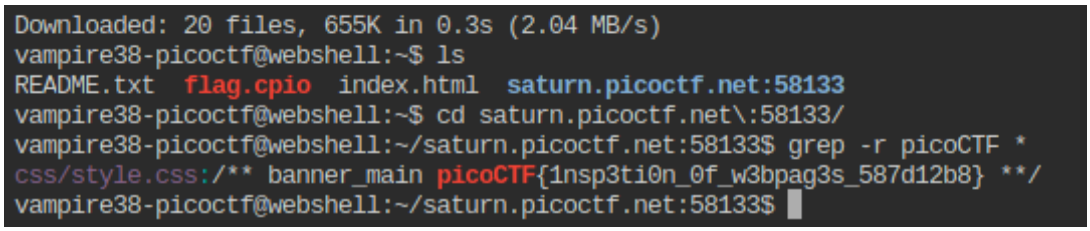
picoCTF{7h3_p47h_70_5ucc355_6db46514}

“Search source”

1- First way:
- open terminal in kali Linux and write “wget -m http://saturn.picoctf.net:58133/”
- open folder saturn.picoctf.net in vscode.
- use ‘grep’ tool in terminal of vscode to get the picoctf flag

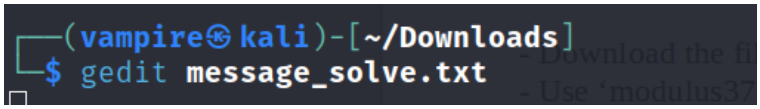


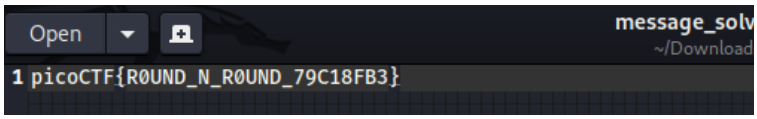
2- Second way:
- open webshell of picoctf and download the folder of flag with “ wget http://saturn.picoctf.net:58133/ ”.
- use ‘grep’ tool to see the picoctf flag with “grep -r picoCTF * ”.



“Basic-mod1”

- Download the file open it and found it contains a strings number.
- Use ‘modulus37’ to get the write number then decode this number as say: 0-25 (upper alphabet), 26-35 (decimal digits) and 36 (_).
- Get the picoCTF flag: picoCTF{R0UND_N_R0UND_79C18FB3}





“Credstuff”

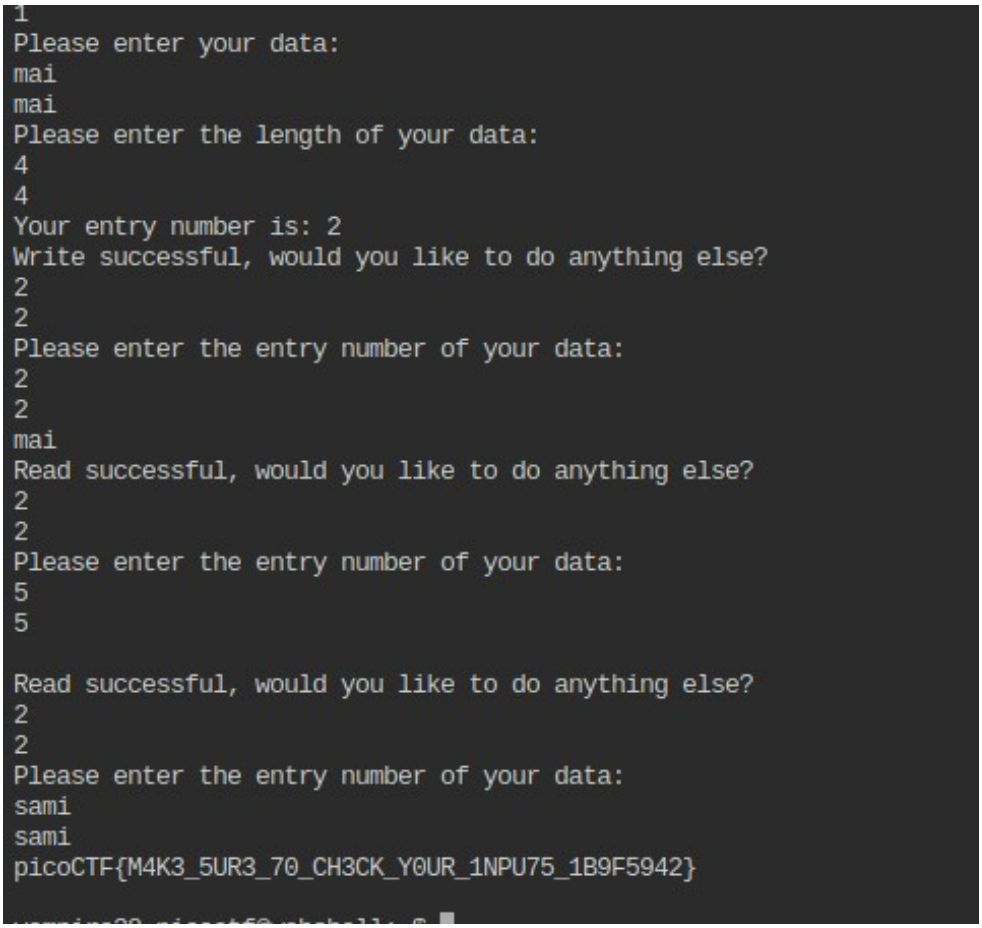
- Download the file then compressed file leak.tar with “tar -xf leak.tar”.
- Get 2 files from leak : 1- username.txt 2- password.txt

- Open username.txt and search for a name ‘cultiris’ I found it in raw -378-
  then open password.txt search on raw -378- I found value had made encoded with rot13. I decoded this value manually with rot13.

“Basic-file-exploit”

- Open web shell for picoctf enter net-cat website:
- Try to see the flag
- Enter “1” for data entry then wrote the data length then choosed “2” for echo if I wrote the data entry with different data I would see the flag:

picoCTF{M4K3_5UR3_70_CH3CK_YOUR_1NPU75_1B9F5942}

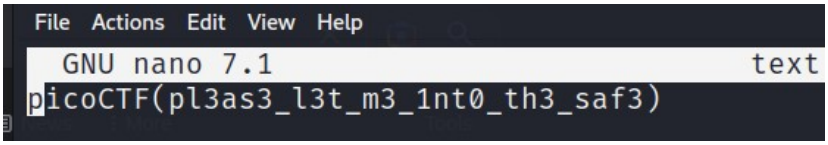


“Safe Opener”

- Use wget “https://artifacts.picoctf.net/c/463/SafeOpener.java” to download file.
- Find the flag decoded with base64 because it contain upper and lower alphabet also numbers.

- Open terminal in kali linux to decode base64 and save the flag in text file with the same of flag format.

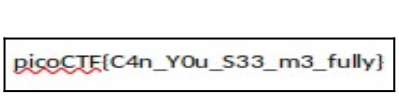
- Open the file then you see the flag.



“Redaction gone wrong”

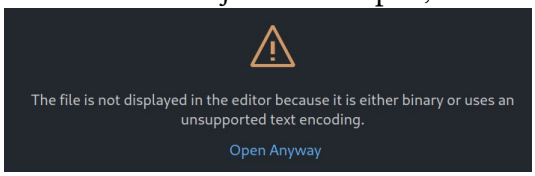
- Download PDF file in kali terminal with “wget -q https://artifacts.picoctf.net/c/264/Financial_Report_for_ABC_Labs.pdf”
- Open the PDF file, you will see redaction text if you press with mouse you will see the mouse cursor change and become texting only in 2 lines:
  - First redacted line contain: Breakdown.

  - Second redacted line contain: The flag.

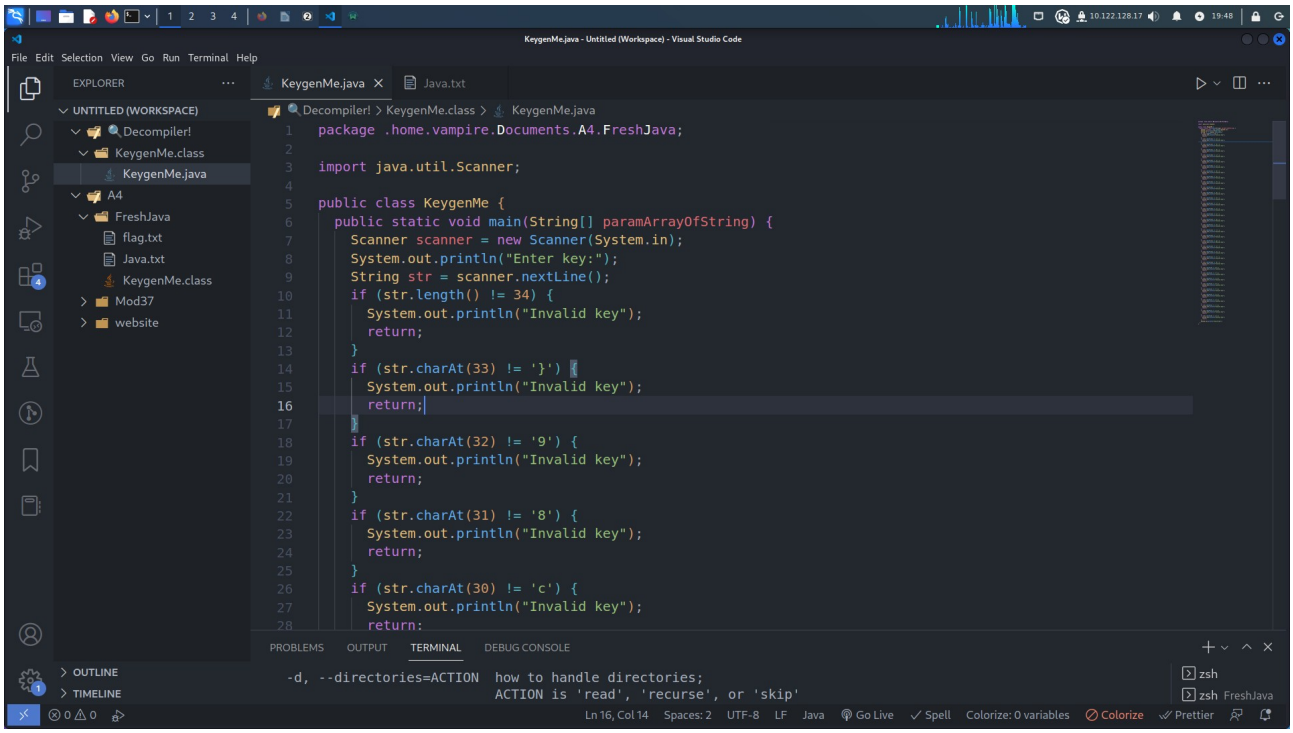


“Fresh Java”

- Open vscode and use terminal with “wget -q https://artifacts.picoctf.net/c/207/KeygenMe.class” to download file.
- Find the file in java don’t open, download extension de-compiler in vscode.



- Copy the code from de-compiler into Java.txt.



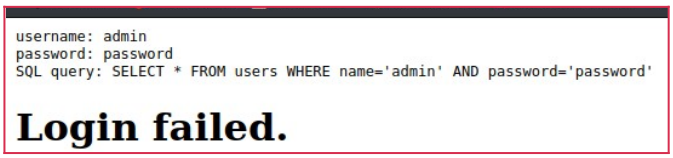
- Open terminal in vscode use grep tool to find the flag with removing all unwanted strings then print the flag in flag.txt.

```
┌──(vampire㉿kali)-[~/Documents/A4/FreshJava]
└─$ cat Java.txt| grep -oE "'.'" | tac | tr -d "'\n" > flag.txt
```

```
┌──(vampire㉿kali)-[~/Documents/A4/FreshJava]
└─$ cat flag.txt
picoCTF{700l1ng_r3qu1r3d_738cac89}
```

“SQLilite”

- Press on launch instance then login.
- Try to write any password with user name admin as example:
  Username: admin Password: password
- Open burb suite or press login in your browser then send the request of website into repeater to show what happened in the response. You will show Login failed.



- Take the request again to intruder to make SQL injection attack with loading the library of payloads.
- Show the response, you will found response with three values:
  500 response: only see the user name and password that you entered.
  200 response: show 2 output
    First: Login failed

    Second: Login success but the flag in plain text if you search carefully, you will find the flag in response at message replied.