

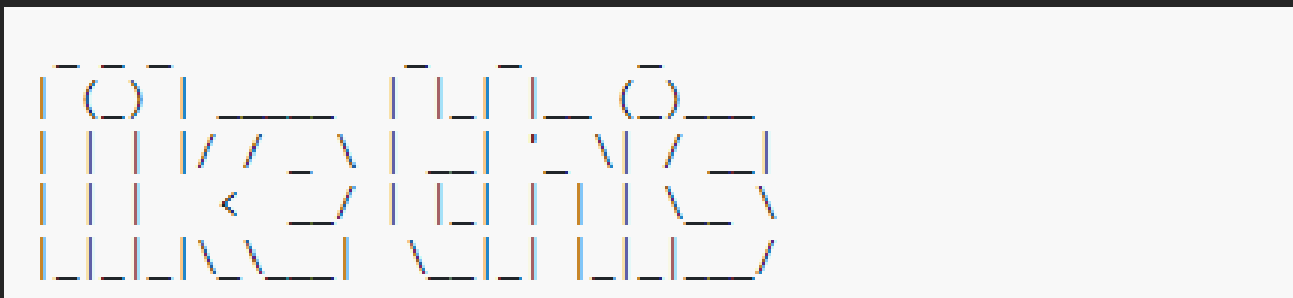


# *Frank, Ian & Glen's Letters*

*Problem set*



[FIGlet](#) برنامه ایست برای تبدیل متن های معمولی به تکست هایی با استایل و فونت خاص که در اوایل دهه 90 توسعه پیدا کرده است . این برنامه که اسم آن مخفف عبارت "[Frank, Ian, and Glen's letters](#)" است ، از [ASCII art](#) برای انجام این کار استفاده میکند.



از طریق [این لینک](#) میتوانید فونت هایی که FIGlet پشتیبانی می کند را مشاهده کنید.  
این ابزار در پایتون از طریق کتابخانه ای به نام [pyfiglet](#) قابل دسترسی و اجراست.

در فایلی به نام `figlet.py` برنامه ای بنویسید که:

- در دو حالت ، بدون پارامتر ورودی `command-line` و یا دقیقا با 2 پارامتر ورودی `command-line` اجرا شود.
  - اگر بدون پارامتر ورودی اجرا شد، به این معنی است که کاربر تمایل دارد فونت خروجی فایل بصورت تصادفی انتخاب شود.
  - اگر برنامه با 2 پارامتر ورودی اجرا شد، به این معنی است که کاربر قصد تعیین نوع فونت خروجی برنامه را دارد . در این حالت ، پارامتر اول بین `f-` و یا `font--` ، پارامتر دوم اسم استاندارد یک فونت خواهد بود.
  - سپس از کاربر یک رشته کارکتر را دریافت کنید
  - متن تبدیل شده را با فونت مشخص شده ارائه دهید.
- در صورتی که پارمتر اول از بین `f-` و یا `font--` نبود و یا پارامتر دوم اسم استاندارد یک فونت نبود، برنامه شما باید با `sys.exit` همراه با یک متن ارور از برنامه خارج شود.

▼ نکته

- برای نصب `pyfiglet` دستور زیر را در ترمینال خود وارد کنید.

```
pip install pyfiglet
```

- چون داکيومنت خود `pyfiglet` به اندازه کافی واضح و روشن نیست، شما میتوانید از این مازول به شکل زیر در برنامه خود استفاده کنید:

```
from pyfiglet import Figlet  
  
figlet = Figlet()
```

با این دستور میتوانید لیستی از فونت های قابل پشتیبانی توسط `pyfiglet` را دریافت کنی

```
figlet.getFonts()
```

برای تعیین کردن فونت میتوانید از تابع زیر استفاده کنید که در آن `f` اسم آن فونت می باشد.

```
figlet.setFont(font=f)
```

و در نهایت برای تبدیل متن میتوانید از تابع زیر استفاده کنید که در آن `s` متن وارد شده توسط کاربر میباشد.

```
print(figlet.renderText(s))
```

- برای مطالعه بیشتر درباره متدهای مختلف مازول `random` میتوانید از [این لینک](#) استفاده کنید.

**توجه :** خروجی این دمو از یک فونت تصادفی استفاده می‌کند.  
خروجی شما ممکن است متفاوت باشد.

$\overline{//} \overline{/_} \overline{\_} \overline{///} \overline{\_}$        $\overline{\_} \overline{\_} \overline{\_} \overline{//} \overline{/_} \overline{///}$   
 $// \overline{\_} \vee \overline{\_} \vee /// \backslash$        $| | // // \overline{\_} \vee \overline{\_} // \overline{\_} //$   
 $/// \overline{\_} // // //$        $| | // // // // // //$   
 $// // \backslash \overline{\_} // \backslash \overline{\_} ( )$        $| \overline{\_} | \overline{\_} \backslash \overline{\_} // // \backslash \overline{\_} , //$   
 $| /$

وارد [code.cs50.io](https://code.cs50.io) شوید، سپس بر روی پنجره‌ی Terminal کلیک کنید. توجه داشته باشید که دستور پنجره‌ی Terminal شما باید به صورت زیر باشد:

```
$
```

سپس کد زیر را اجرا کنید تا یک پوشه به نام `figlet` در `codespace` ایجاد شود:

```
$ mkdir figlet
```

سپس `cd figlet` را اجرا کنید تا به پوشه‌ی `figlet` منتقل شوید. اکنون شما باید آدرس `$ figlet/` را در Terminal مشاهده کنید:

```
$ cd figlet  
figlet/ $
```

حالا می‌توانید دستور `code figlet.py` را اجرا کنید تا فایل با نام `figlet.py` ایجاد شود:

```
figlet/ $ code figlet.py
```

اکنون می‌توانید در این فایل برنامه‌ی خود را بنویسید.

در این قسمت توضیح داده شده که چطور خودتان به عنوان یک کاربر برنامه نوشته شده را تست کنید

- برنامه‌ی خود را با دستور `python figlet.py test` اجرا کنید، اگر برنامه‌ی شما درست کار می‌کند با `sys.exit` خارج می‌شود و ارور زیر را نمایش می‌دهد:

```
Invalid usage
```

- برنامه‌ی خود را با کد `python figlet.py -a slant` اجرا کنید، اگر برنامه‌ی شما درست کار می‌کند با `sys.exit` خارج می‌شود و ارور زیر را نمایش می‌دهد:

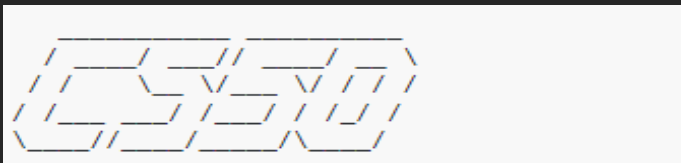
```
Invalid usage
```

- برنامه‌ی خود را با کد `python figlet.py -f invalid_font` اجرا کنید، اگر برنامه‌ی شما درست کار می‌کند با `sys.exit` خارج می‌شود و ارور زیر را نمایش می‌دهد:

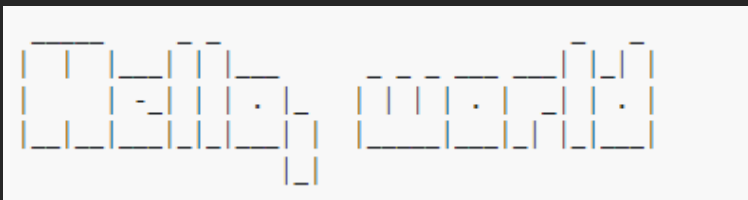
```
Invalid usage
```



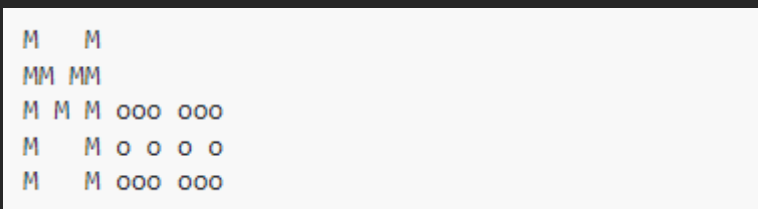
- برنامه‌ی خود را با کد `python figlet.py -f slant` اجرا کنید و CS50 را به عنوان ورودی بدهید. در صورتی که کد شما درست باشد خروجی زیر را مشاهده می‌کنید:



- برنامه‌ی خود را با کد `python figlet.py -f rectangles` اجرا کنید و Hello, world را به عنوان ورودی بدهید. در صورتی که کد شما درست باشد خروجی زیر را مشاهده می‌کنید:



- برنامه‌ی خود را با کد `python figlet.py -f alphabet` اجرا کنید و Moo را به عنوان ورودی بدهید. در صورتی که کد شما درست باشد خروجی زیر را مشاهده می‌کنید:



❖ شما می‌توانید از آدرس زیر برای بررسی کردن کد خود استفاده کنید. CS50 از این برنامه برای آزمایش کد شما استفاده می‌کند. از این دستور استفاده کنید تا کدهایتان را تست کنید.

```
$ check50 cs50/problems/2022/python/figlet
```

لبخند های سبز :) به این معنی هستند که برنامه‌ی شما در تست **قبول شده** و اخم های قرمز :) یعنی برنامه‌ی شما **دارای ایراد** هست، همچنین پیام های زرد ! بدین معنی است که برای اینکه check50 بتواند این بخش برنامه شما را بررسی کند باید ابتدا ارورهایی که با اخم های قرمز :) جلوتر آمده اند را برطرف کنید.

. برای دیدن جزییات بیشتر درباره این تست که توسط check50 صورت گرفت، وارد لینکی که به شما خروجی داده است شوید.

با اجرا کردن صورت زیر در Terminal پاسخ خود را ارسال کنید.

```
$ submit50 cs50/problems/2022/python/figlet
```

توجه کنید هرچند بار که لازم باشد میتوانید کد خود را اصلاح و ارسال کنید ، نمره آخرین پاسخ شما در نظر گرفته میشود .

# CS50x Iran

Harvard's Computer Science 50x Iran

