



A CLI*
DEDICATED
OPEN
PLANNER
CALENDAR

*COMMAND LINE INTERFACE

JANUARY

MONDAY

TUESDAY

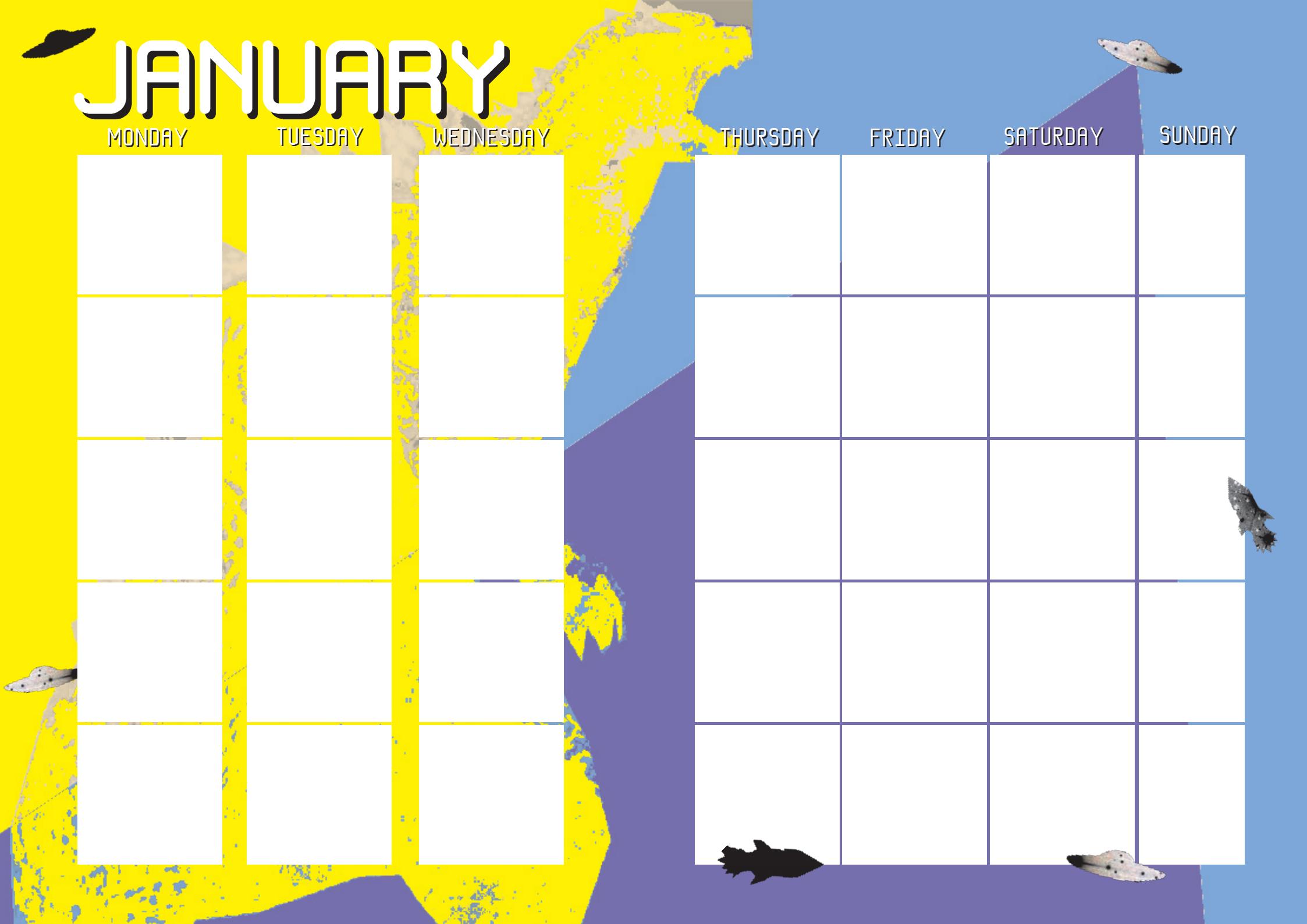
WEDNESDAY

THURSDAY

FRIDAY

SATURDAY

SUNDAY



YOUR GEEKY
TECHIE OPEN CALENDAR
MADE WITH CARE USING
OPEN SOURCE SOFTWARE
INKSCAPE SCRIBUS
AND GIMP
FONT IS THE FANTASTIC
KP PROGRAMMER NBP

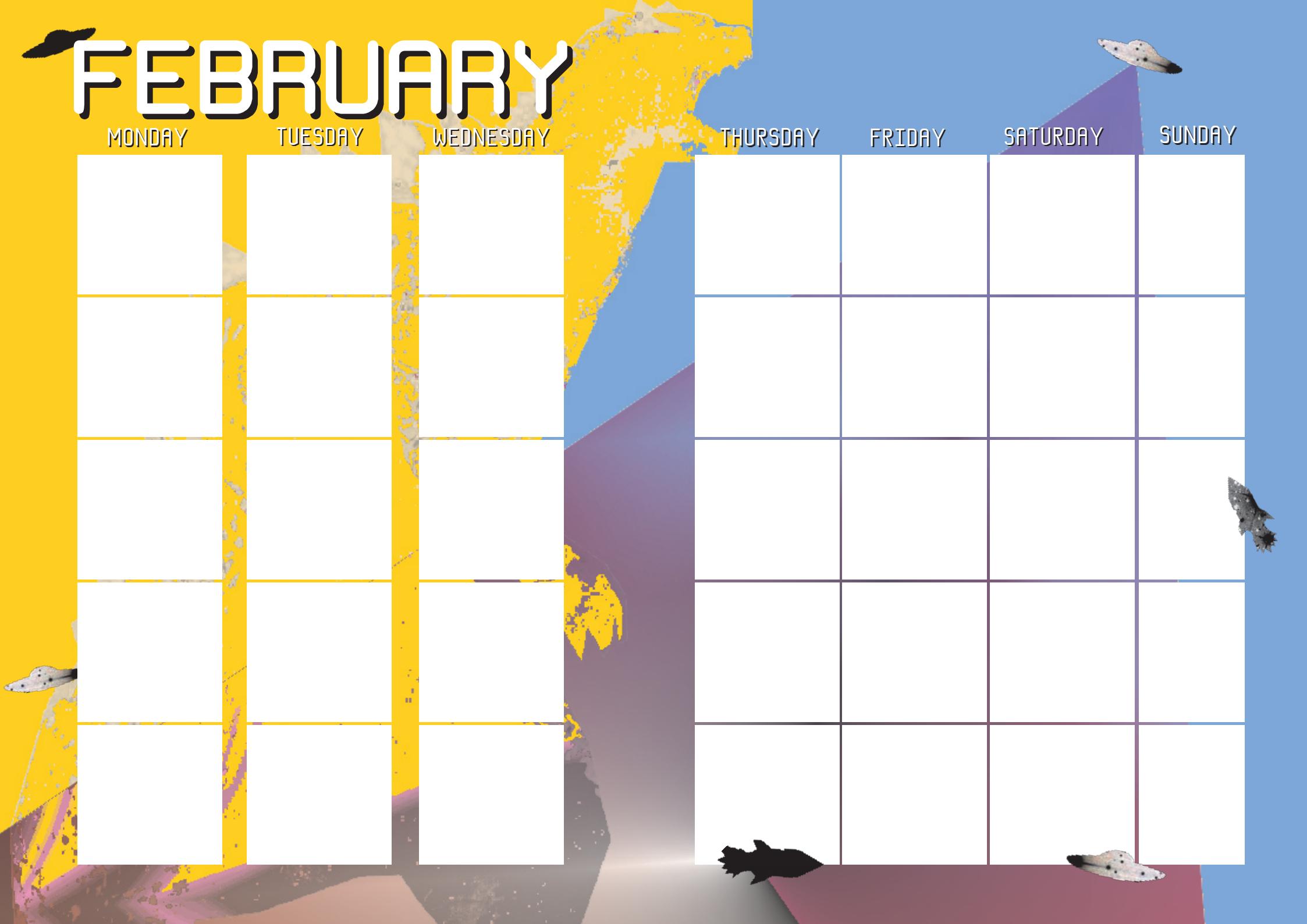
BY PSAROSKALAZINES.GR

```
$ grep -r "word" *
# searches all files starting from the current dir for the word
```

```
$ var="foo"; echo "${var}suffix" # double quotes for var  
expansion and square brackets when a suffix is added
```

```
$ my_list=(foo bar dope poop)
# makes a list of the items within the brackets (...)
```

`$ '@'` makes an array of all script's arguments
e.g. `my_script.sh arg1 arg2`



FEBRUARY

MONDAY

TUESDAY

WEDNESDAY

THURSDAY

FRIDAY

SATURDAY

SUNDAY

\$ which <command> finds which binary is used e.x which python

```
$ alias ltr="ls -ltr" makes a shortcut for listing  
files by timestamp from earliest to last
```

```
$ echo "hello" " world" to add strings instead of +
```

```
$ cmd 2> /var/log/error_log redirects error to log file
```

```
$ cmd > /var/log/access.log 2>&1 # redirects output of cmd  
and errors to access.log
```

MARCH

MONDAY

TUESDAY

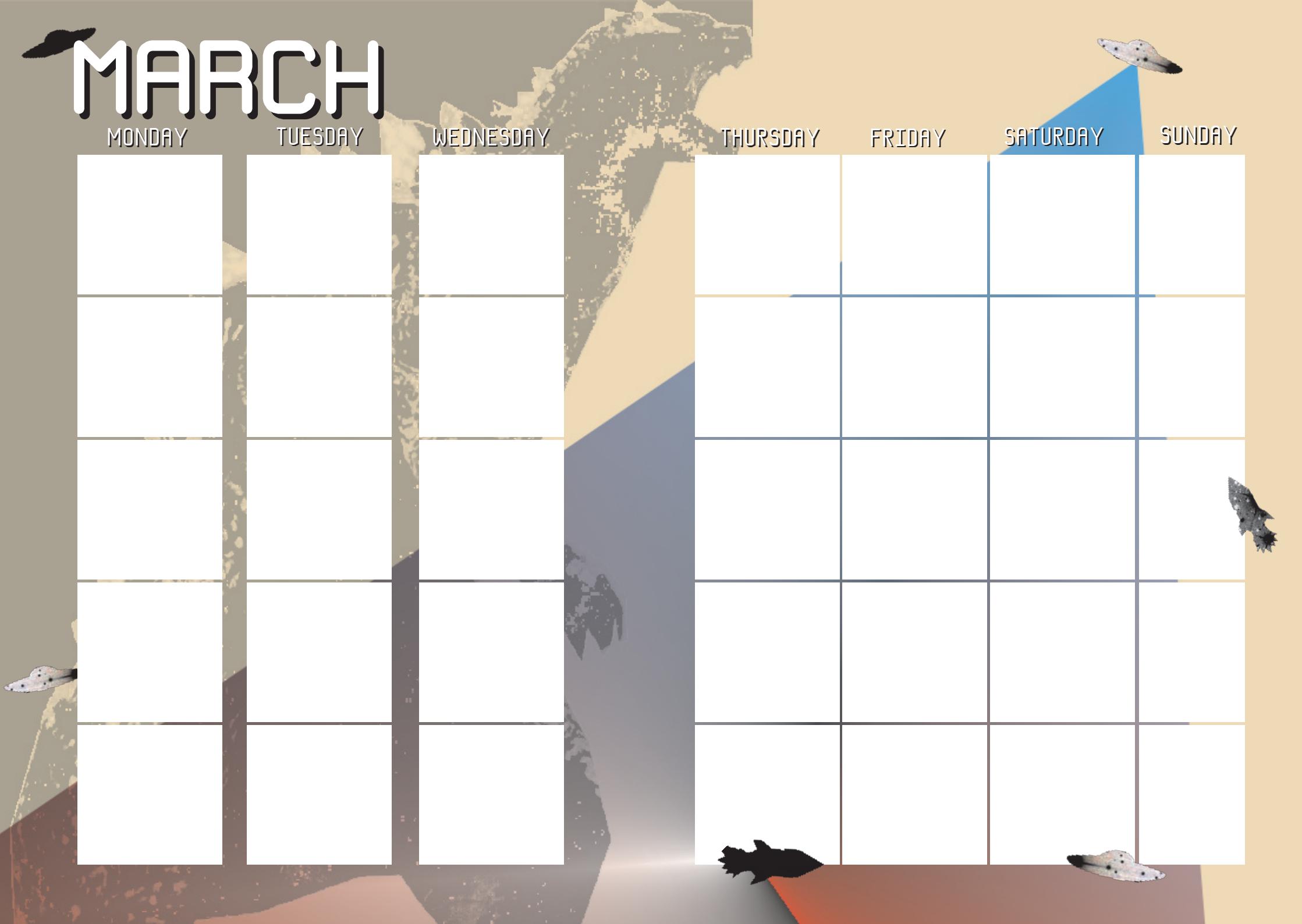
WEDNESDAY

THURSDAY

FRIDAY

SATURDAY

SUNDAY



```
$ cat file.txt | sudo tee /etc/openvpn/clients/user.conf  
# adds content of file.txt in a root owned file
```

```
$ var=$(ls *.png)  
# captures stdout of command and assigns it to var
```

```
$ for f in $(ls images/*.png);do convert $f -resize 800x $f;done  
# resizes images in a directory, requires imagemagick
```

```
$ var=$((x+y)) # double brackets for calculations
```

```
$ if [ x = y ];then echo true;fi  
# evaluates true or false, works for POSIX shell
```

APRIL

MONDAY

TUESDAY

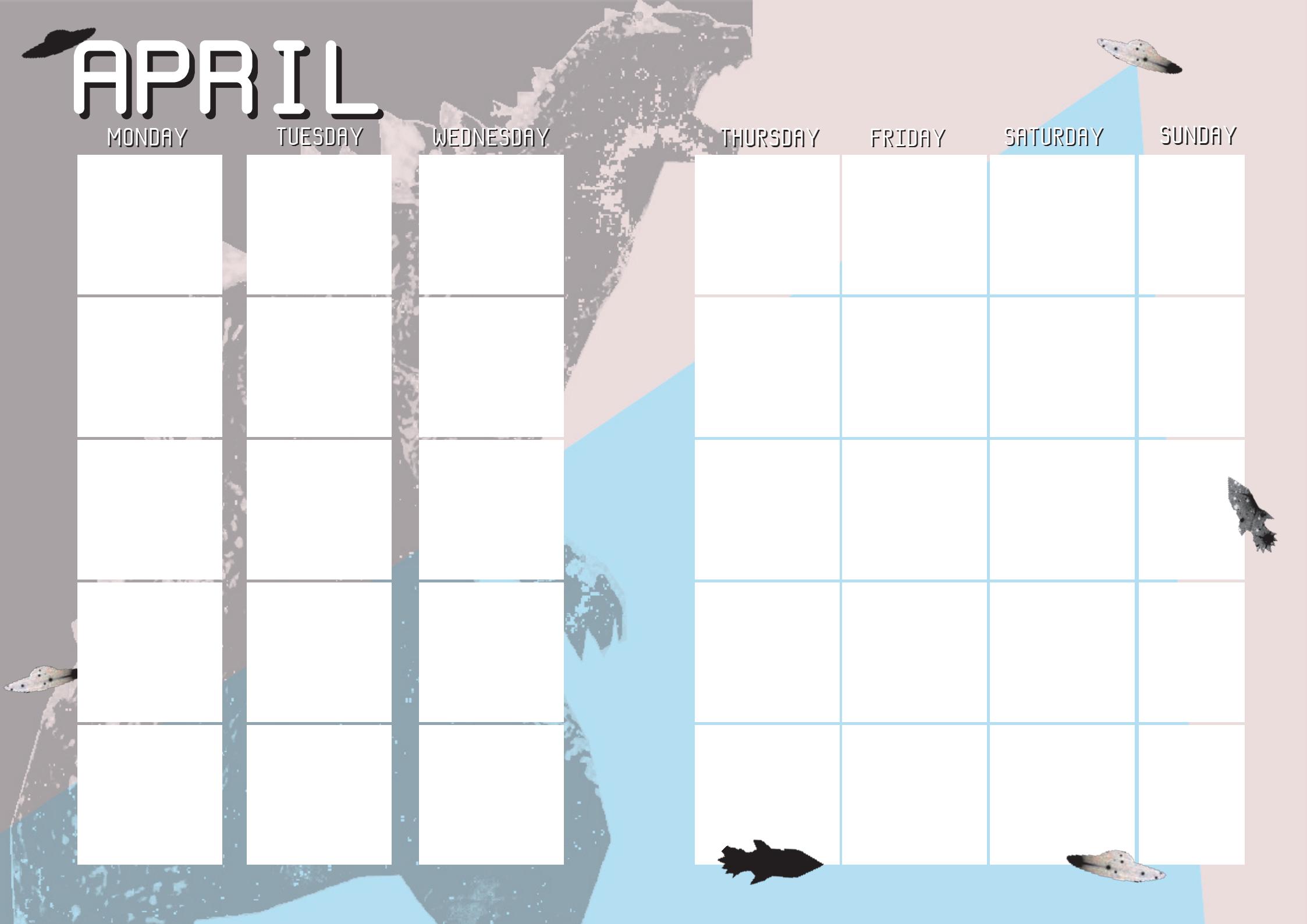
WEDNESDAY

THURSDAY

FRIDAY

SATURDAY

SUNDAY



```
$ if [[ x = y ]] # only works for bash shell
```

```
$ for n in {0..10} # loops from 0 to 10, works for POSIX shells
```

```
$ for i in $(seq 0 10), or $ for ((n=0; n < 10; n++))  
# c style loops only work for bash
```

\$ %s/foo/bar/gc% # replaces all instances of foo with bar and
requires confirmation

```
$ var="$?" # captures exit status at the end of a script
```

MAY

MONDAY

TUESDAY

WEDNESDAY

THURSDAY

FRIDAY

SATURDAY

SUNDAY

```
$ cat < - or cat ./-
# reads a file whose name is dashed
```

```
$ find -readable -size 4M # checks which files are human-  
readable from current dir and of 4M size
```

```
$ find -readable \! -executable  
# finds a non-executable, human-readable file from current dir
```

```
$ find / -group <user> -user <group> -type f 2> /dev/null  
# find files owned by user and group from dir /
```

```
$ cat file.txt | sort | uniq -u  
# finds unique lines in a file
```

JUNE

MONDAY

TUESDAY

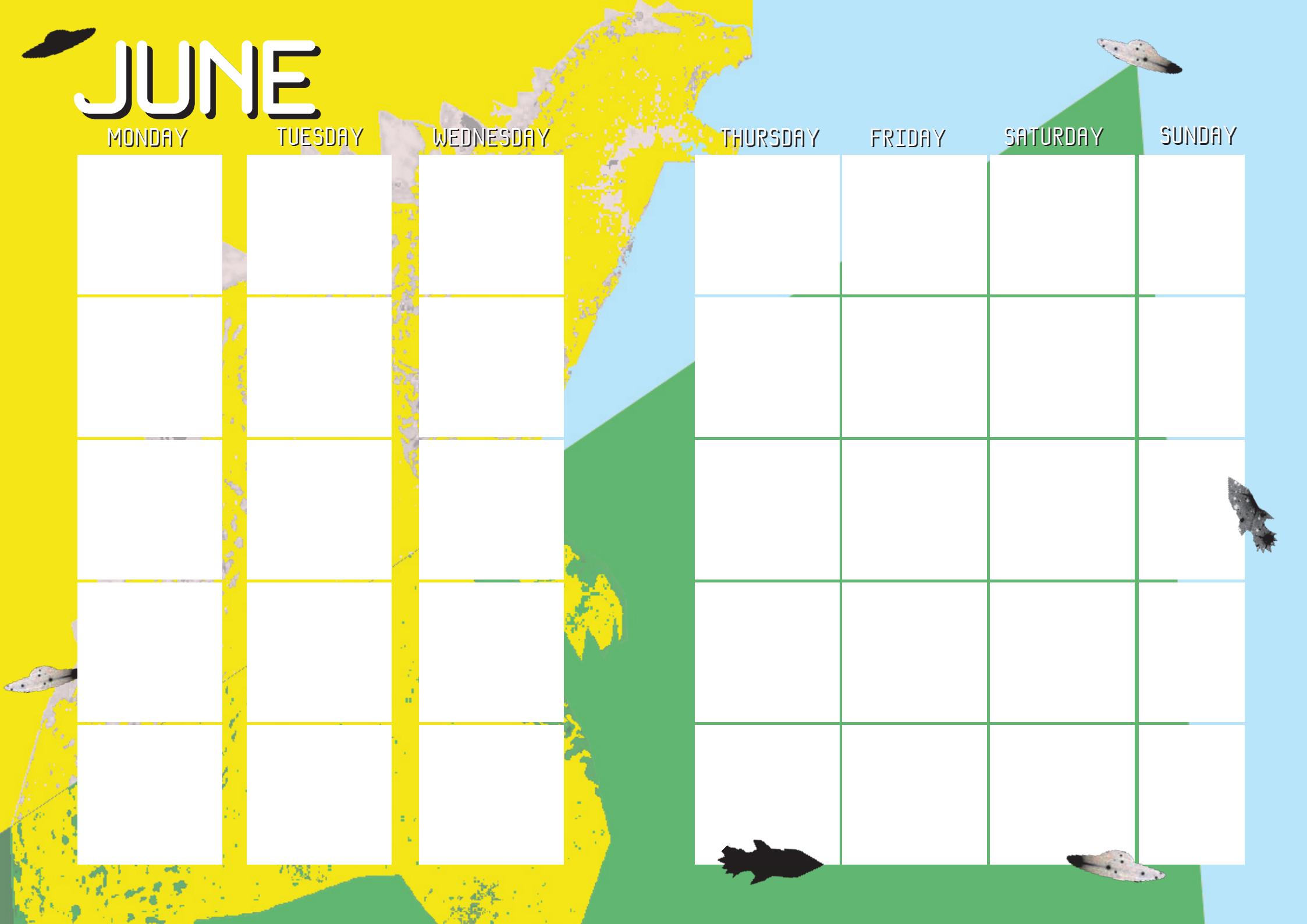
WEDNESDAY

THURSDAY

FRIDAY

SATURDAY

SUNDAY



```
$ echo "Phone number is 123456789" | tr -cd [:digit:]  
# deletes all chars but numbers
```

```
$ echo "Text has too many spaces" | tr -s " "
# replaces multiple spaces with single space
```

```
$ cat file.txt | tr 'a-z' 'A-Z'  
# converts from lower case to upper case
```

```
$ cat file.txt | tr -d 'is'  
# deletes i and s from file but not pattern 'is'
```

```
$ echo "<my string>" | tr '[A-Za-z]' '[N-ZA-Mn-za-m]'  
# encrypts a string by shifting letters 13 times
```

JULY

MONDAY

TUESDAY

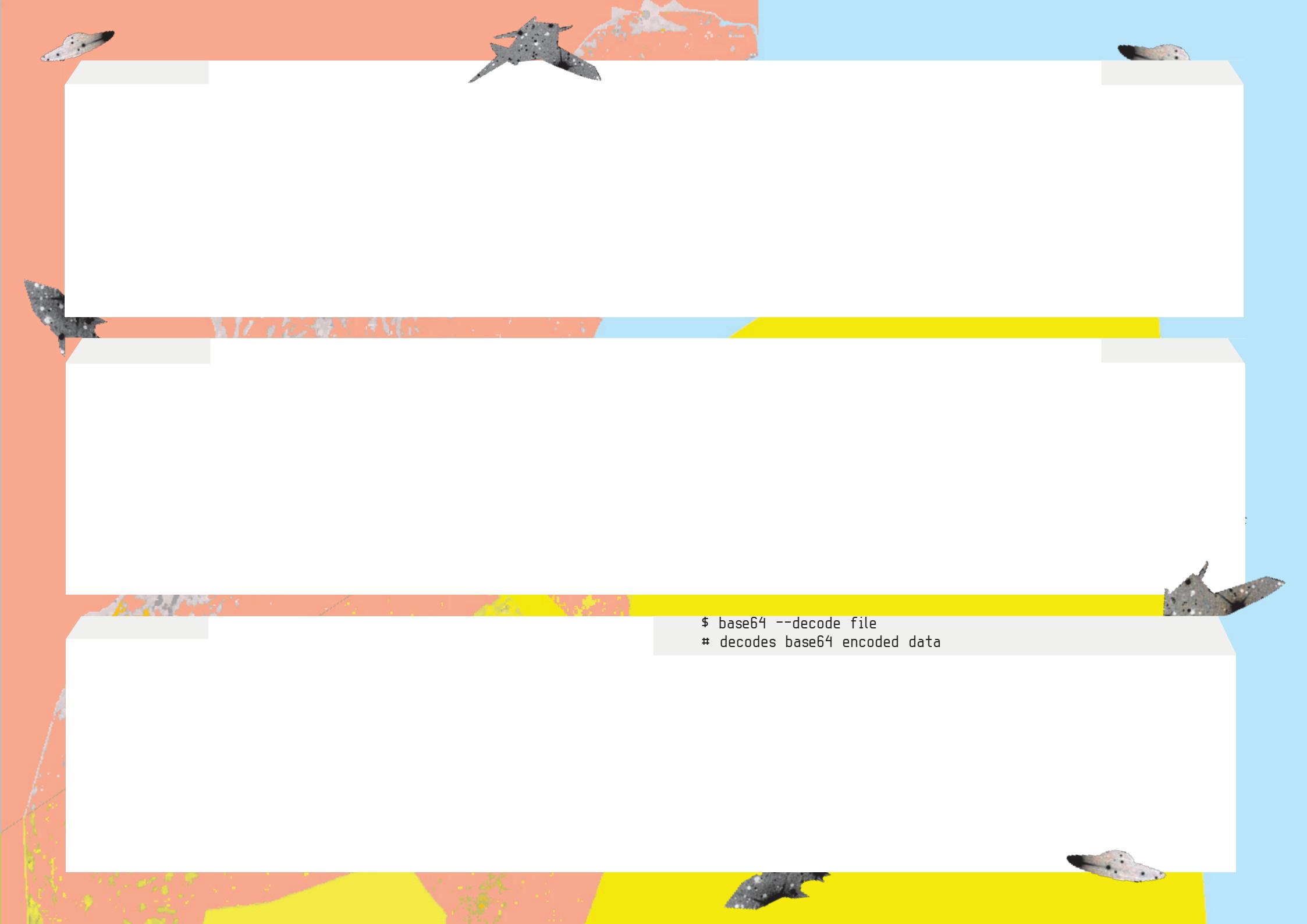
WEDNESDAY

THURSDAY

FRIDAY

SATURDAY

SUNDAY



```
$ base64 --decode file  
# decodes base64 encoded data
```

```
$ strings file | grep <pattern> # outputs printable characters  
from a binary file then finds a pattern
```

```
$ sudo strings /dev/mem | less  
# read through device's RAM
```

```
$ strings -f /bin/* | grep Copyright  
# finds all authors in the executable software
```

```
$ md5sum -b file  
# creates a checksum for a binary file
```

AUGUST

MONDAY

TUESDAY

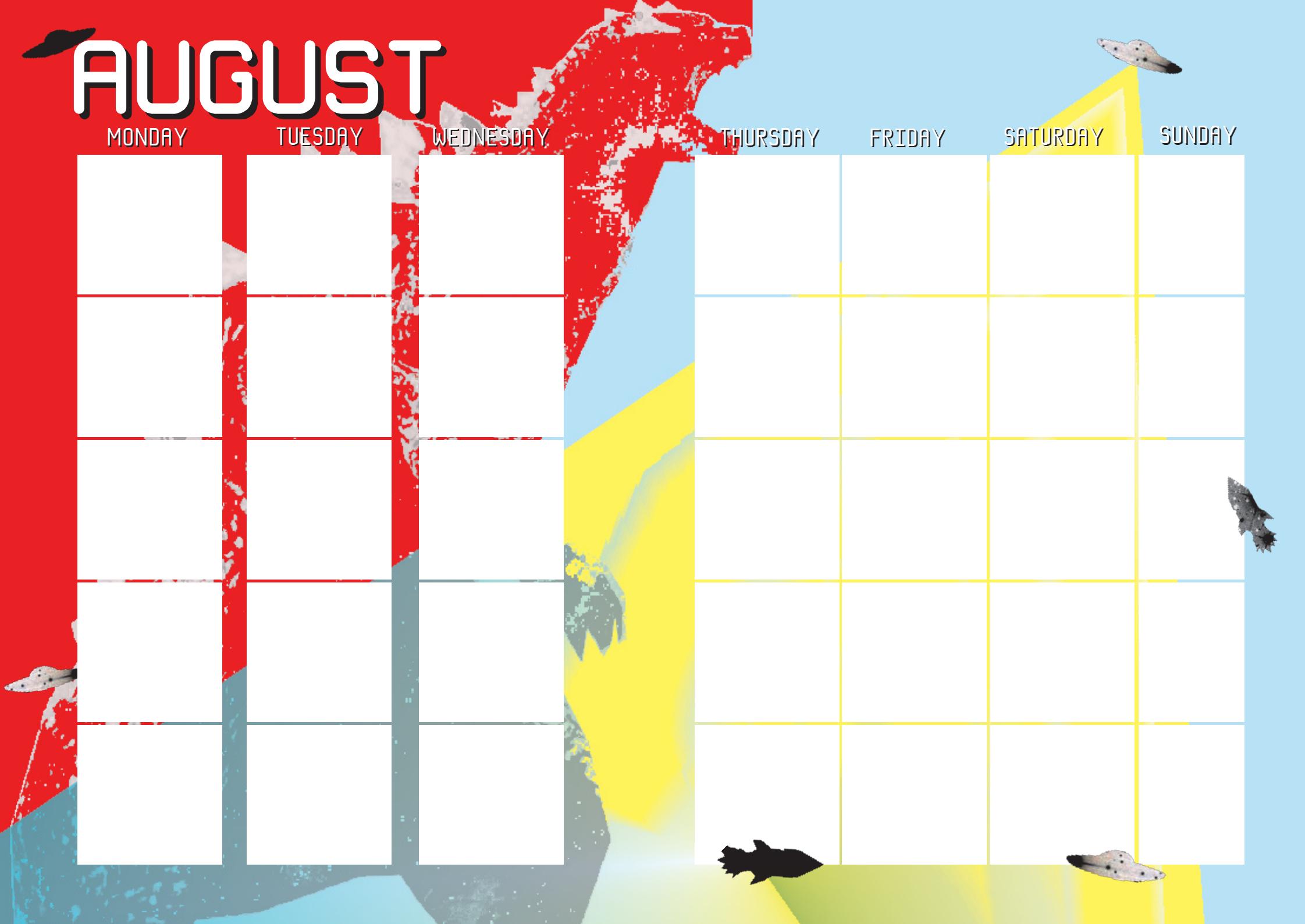
WEDNESDAY

THURSDAY

FRIDAY

SATURDAY

SUNDAY



```
$ du -sh /dir  
# checks the size of a dir and skips subdirs
```

```
$ du -h --max-depth 1 | sort -n  
# lists all directories and sorts by size
```

```
$ du -h --max-depth 1 | sort -n | head -10  
# lists top 10 by size directories
```

```
$ tail -n N <file1> <file2> <file3>
# outputs last n lines of the files
```

```
$ tail -f /var/log/file.log  
# monitor files in real time
```

SEPTEMBER

MONDAY

TUESDAY

WEDNESDAY

THURSDAY

FRIDAY

SATURDAY

SUNDAY

```
$ nl file | tail -3  
# outputs the lines with numbers
```

```
$ cut -c -14 file  
# outputs characters from 1 to 14 position (inclusive)
```

```
$ cut -f 5- -d',' file.csv | head # uses ',' as a column  
separation & outputs the columns from 5th position
```

```
$ awk 'a[$0]++' file  
# prints duplicate lines
```

```
$ awk '!a[$0]++' file  
# removes duplicate lines from stdout
```

OCTOBER

MONDAY

TUESDAY

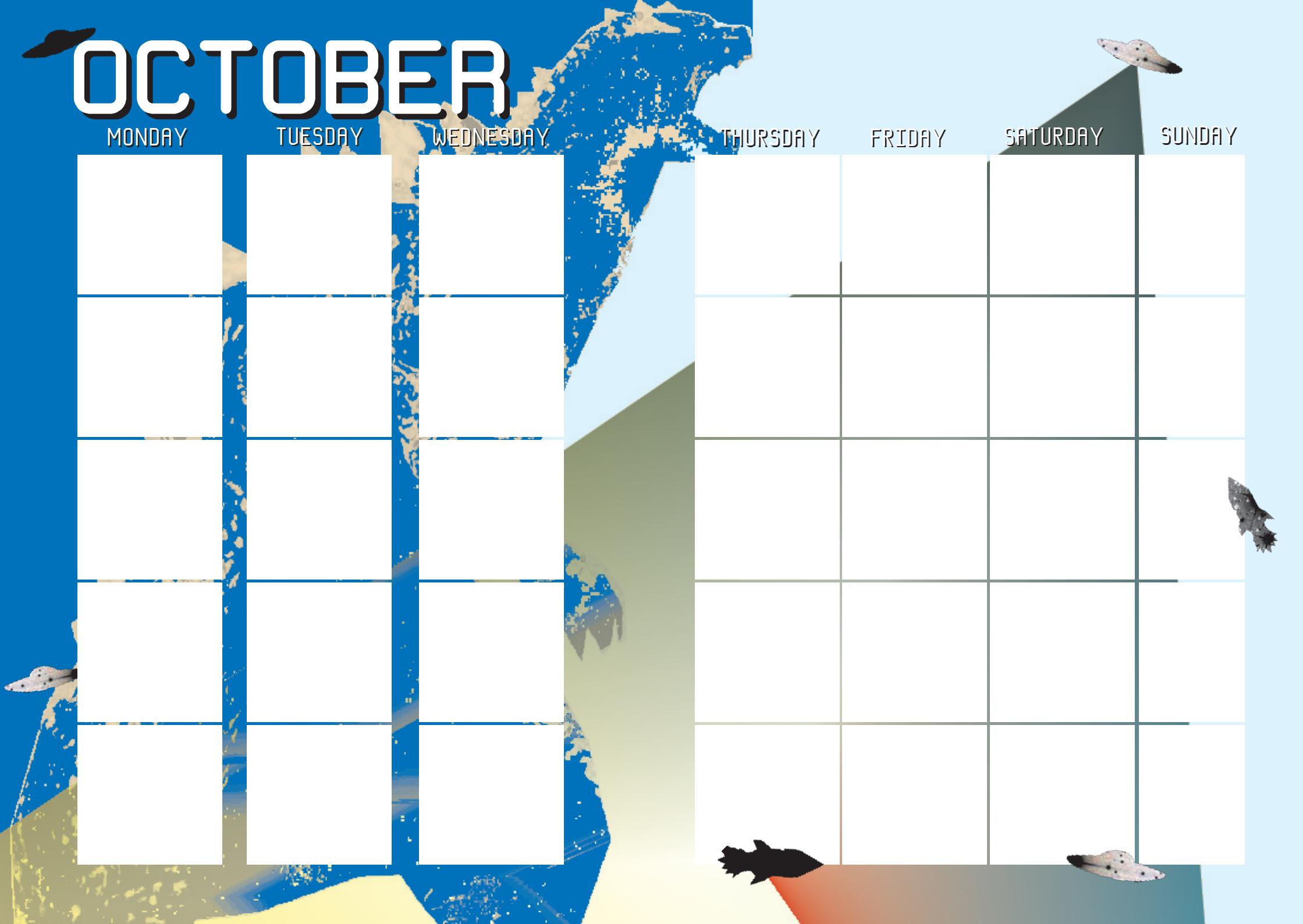
WEDNESDAY

THURSDAY

FRIDAY

SATURDAY

SUNDAY



```
$ awk '{ printf("%s\n ", $1) }' FS=: /etc/passwd; echo  
# lists all users in the system
```

```
$ sed -n -e '$p' file  
# prints the last line of the file
```

```
$ sed -n -e '1p' file1 file2 file3  
# prints first line of each file
```

```
$ sed -n -e '/systemd/p' file  
# prints only lines containing the string "systemd"
```

```
$ sed -En -e '/(www)|(mail)/p' file # option E extends POSIX  
regexpes & no need to escape symbols with '\'
```

NOVEMBER

MONDAY

TUESDAY

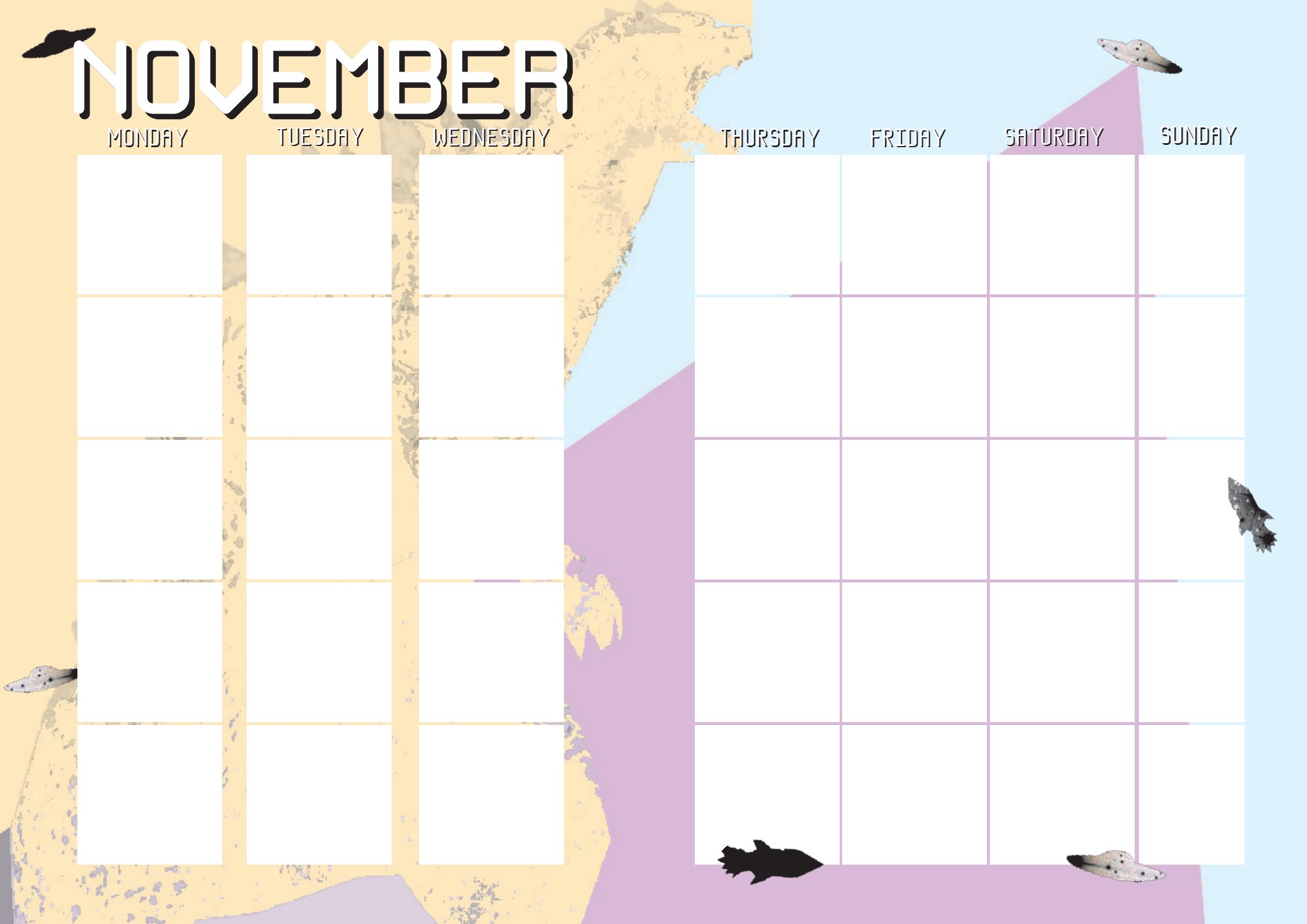
WEDNESDAY

THURSDAY

FRIDAY

SATURDAY

SUNDAY



```
$ sed -En -e '/.{5}$/{p}' file  
# Print lines containing exactly 5 characters
```

```
$ ngrep -d any port 80  
# prints packets passing through port 80
```

```
$ ngrep -d any "domain.com" port 80  
# prints packets containing domain.com
```

```
$ nmap -sn 192.168.1.0/24  
# scans for active hosts on a local network
```

```
$ nmap -p 80,443 192.168.1.1  
# scans for several ports
```

DECEMBER

MONDAY

TUESDAY

WEDNESDAY

THURSDAY

FRIDAY

SATURDAY

SUNDAY

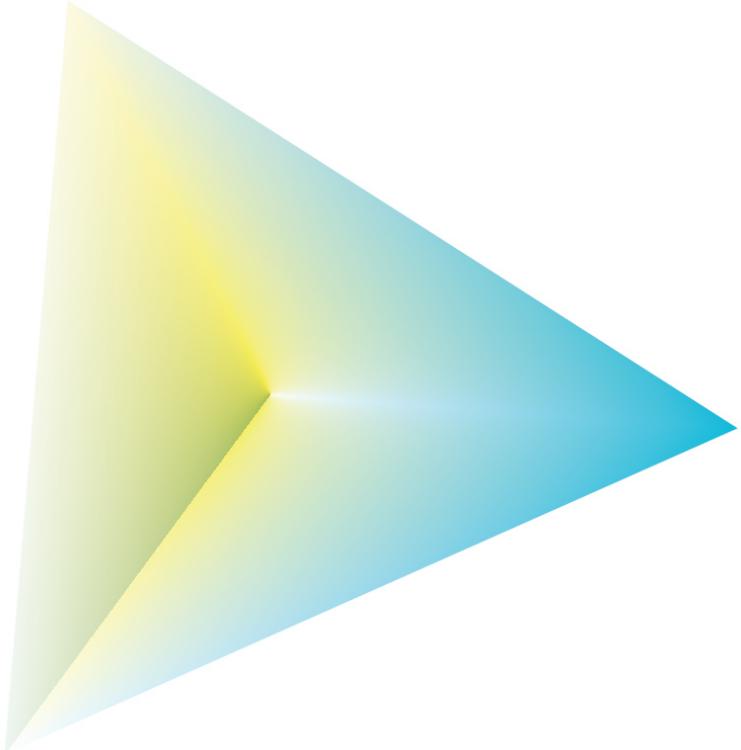
```
$ traceroute <domain or IP>
# finds the network path from current device to domain/IP
```

```
$ sudo tcpdump -n -i any port 53  
# prints packets of DNS queries
```

```
$ sudo tcpdump -w file.pcap dest port 1194  
# saves traffic that goes to openvpn in a file
```

```
$ dig -t ANY domain.com +noall +answer  
# prints all the DNS record types of the domain.com
```

```
$ sudo iptables -t nat -L --line-numbers # lists current rules  
regarding NAT
```



FIND MORE PRINTS & ZINES
TO DOWNLOAD
FROM PSAROSKALAZINES.GR
SUPPORT THE PROJECT by
PURCHASING PHYSICAL PRINTS
FROM CUCU.GR OR SEND
DONATIONS to
KO-FI.COM/PSAROSKALAPOT
THANKS!