

# **REPORT - INT 301**

## **Open Source Technologies**

**B.Tech. CSE**

**Submitted to - Rajeshwar Sharma Sir**



**L**OVELY  
**P**ROFESSIONAL  
**U**NIVERSITY

**PROJECT-** Using Open Source Software to \*System information utility , \*Display the Mac product key and ID, \*A list of installed softwares, \*and all the currently running processes , \*GITHUB-Repository , \*And save the Report in a simple text file.

**Submitted by-**

**1. Nishant kumar**

*Department of Computer science and engineering LPU*

**\* GITHUB-Repository -**

<https://github.com/dxenrique/Final-Report.git>

# **1.Introduction -**

## **1.1 Objective of the project**

- To develop a system information utility using open source software.
- To display the Windows product key and ID.
- To list installed softwares .
- To list all the currently running processes.
- To save the report in a simple text file.
- To upload the report to the GITHUB-Repository.

## **1.2 Description of the project**

The aim of this project is to develop a system information utility that can be used to display various information about a Windows computer. The utility will be developed using open source software to ensure that it is easily accessible and can be used by anyone.

The utility will be designed to display the Windows product key and ID, a list of all installed software on the computer, and a list of all the currently running processes. The information will be displayed in a user-friendly format that is easy to read and understand.

The utility will also include a feature to save the report in a simple text file. This will allow users to easily access the report later, without having to run the utility again.

## **1.2 Scope of the project**

The scope of this project is limited to developing a system information utility that can display the Windows product key and ID, a list of installed software, and all the currently running processes. The utility will be developed using open source software and will be designed to save the report in a simple text file.

# **2.SYSTEM DESCRIPTION -**

## **2.1 Target system description**

The scope of this project is limited to developing a system information utility that can display the Windows product key and ID, a list of installed software, and all the currently running processes. The utility will be developed using open source software and will be designed to save the report in a simple text file. The project does not aim to provide any additional functionality beyond the scope of the defined objectives.

## **2.2 Assumptions and dependencies**

- The project assumes that the user has basic knowledge of using command-line interfaces.
- The project depends on the Homebrew package manager for macOS to install and manage command-line software.
- The project depends on the "mas" command-line tool, which is a package provided by Homebrew, to access the Mac App Store and retrieve information about installed applications.

- The project assumes that the user has a stable internet connection to install and update the Homebrew package manager and the “mas” command-line tool.

### **2.3 Functional dependencies**

- The project relies on the "system\_profiler" command-line tool to gather system information utility, such as the Mac product key and ID and a list of installed software.
- The project relies on the "ps" command-line tool to retrieve information about all currently running processes on the Mac computer.

### **2.4 Non-Functional dependencies**

- The project should be efficient and responsive, providing system information utility and process information in a timely manner.
- The project should be user-friendly, providing clear and concise instructions for the user to access system information utility and process information.
- The project should be reliable and accurate, providing correct and up-to-date information about the system.

### **2.5 Data-set used in support of your project**

The project does not rely on any external datasets. However, it uses the information provided by the "system\_profiler" and "ps" command-line tools to gather and display system information utility and process information.

## 3. Analysis Report-

### STEP 1: Install Homebrew:

Homebrew is an open source software. It is a popular package manager for macOS that allows users to easily install and manage open source software packages and libraries from the command line. The Homebrew codebase is available on GitHub and is distributed under the MIT license, which means that it is free to use, modify, and distribute.

Homebrew is a package manager for macOS that provides a simple and efficient way to install, manage, and update open source software packages and libraries from the command line. With Homebrew, users can easily install and manage a wide range of software packages and libraries, including programming languages, web servers, databases, command-line tools, and more.

### Some of the benefits of using Homebrew include:

- Easy installation and management of open source software packages and libraries
- Automatic dependency resolution and installation
- Ability to update packages with a single command
- Support for multiple versions of the same package
- Simple uninstallation of packages
- Integration with macOS and the command line interface

Overall, Homebrew makes it easy for users to set up and maintain a custom software development environment on macOS, with access to a wide range of open source software packages and libraries.

Open Terminal and run the following command to install Homebrew:

```
/bin/bash -c "$(curl -fsSL  
https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

### STEP 2: Install the “mas” command-line tool using Homebrew:

```
'brew install mas'
```

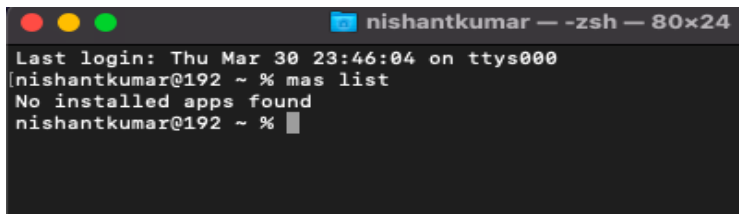
This command will install the "mas" tool, which is a command-line interface for the Mac App Store.

**STEP 3:** To list all the installed applications from the App Store, run the following command in Terminal: (**\*Apps installed from Apple store only**)

*'mas list'*

This will display a list of all the apps installed on your Mac that were downloaded from the App Store.

## SNAPSHOTS

A terminal window titled 'nishantkumar — -zsh — 80x24'. The output shows the last login time and the result of the 'mas list' command, which is 'No installed apps found'.

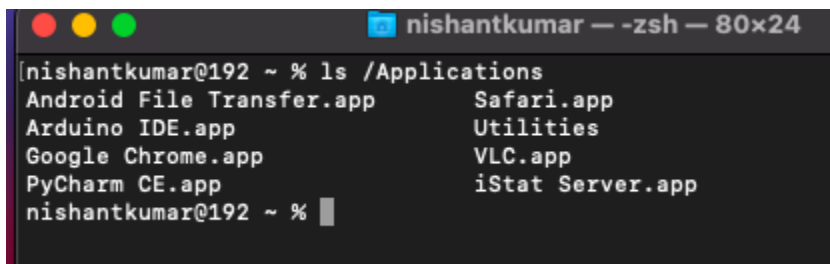
```
nishantkumar@192 ~ % mas list
No installed apps found
nishantkumar@192 ~ %
```

Since I have not installed any apps from the apple store it's showing no installed apps found.

**STEP 4:** To list all the installed applications installed on mac including both third-party apps and system apps run the following command in the terminal:

*'ls /Applications'*

## SNAPSHOTS

A terminal window titled 'nishantkumar — -zsh — 80x24'. The output shows a list of applications installed in the /Applications directory, including Android File Transfer.app, Safari.app, Arduino IDE.app, Utilities, Google Chrome.app, VLC.app, PyCharm CE.app, and iStat Server.app.

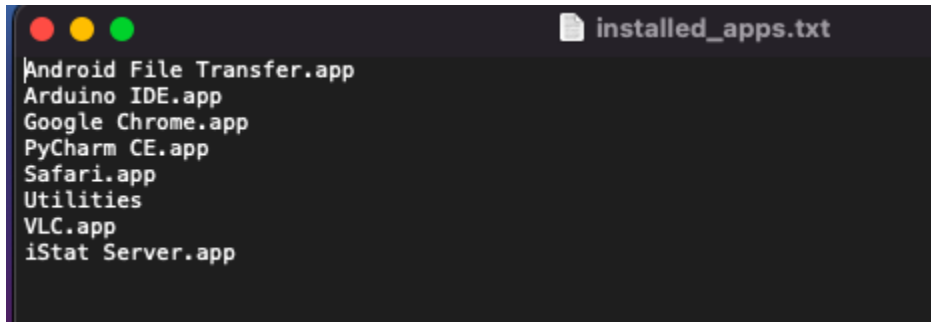
```
nishantkumar@192 ~ % ls /Applications
Android File Transfer.app  Safari.app
Arduino IDE.app           Utilities
Google Chrome.app        VLC.app
PyCharm CE.app           iStat Server.app
nishantkumar@192 ~ %
```

**STEP 5:** To save the list of installed applications to a text file run the following command in the terminal:

*'ls /Applications > installed\_apps.txt'*

This will save the output of the “ls /Applications” command to a file named “installed\_apps.txt.”

## SNAPSHOTS

A screenshot of a macOS Terminal window. The title bar shows three colored window control buttons (red, yellow, green) on the left and a document icon followed by the filename 'installed\_apps.txt' on the right. The terminal content displays the output of the 'ls /Applications' command, listing various applications: Android File Transfer.app, Arduino IDE.app, Google Chrome.app, PyCharm CE.app, Safari.app, Utilities, VLC.app, and iStat Server.app.

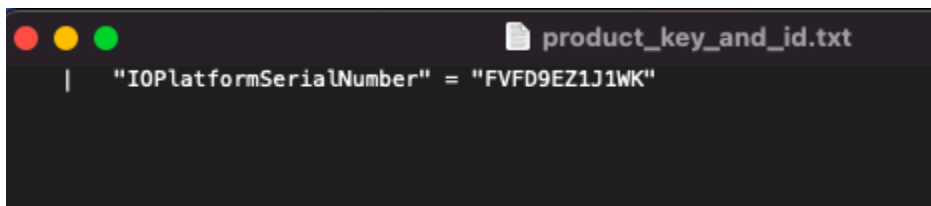
```
Android File Transfer.app
Arduino IDE.app
Google Chrome.app
PyCharm CE.app
Safari.app
Utilities
VLC.app
iStat Server.app
```

**STEP 6:** To get the product key and ID for your Mac and saving it in a file , run the following command in Terminal:

*‘ioreg -l | grep IOPlatformSerialNumber > product\_key\_and\_id.txt’*

This will save the product key and ID information to a file named "product\_key.txt" in the current directory.

## SNAPSHOTS

A screenshot of a macOS Terminal window. The title bar shows three colored window control buttons (red, yellow, green) on the left and a document icon followed by the filename 'product\_key\_and\_id.txt' on the right. The terminal content displays the output of the 'ioreg -l | grep IOPlatformSerialNumber' command, showing the IOPlatformSerialNumber as 'FVFD9EZ1J1WK'.

```
| "IOPlatformSerialNumber" = "FVFD9EZ1J1WK"
```

**STEP 7:** To list all the currently running processes, run the following command in Terminal:

*‘ps aux’*

This will list all the running processes and display them on the terminal.

## SNAPSHOTS

```
nishantkumar — zsh — 80x24
System/Library/
root      27755    0.0    0.0    4361944    3148    ??    Ss    8:36AM    0:00.04 /
usr/libexec/dis
nishantkumar 27740    0.0    0.0    4297412    2160    s000    S    8:34AM    0:00.10 -
zsh
root      27739    0.0    0.1    4329816    6124    s000    Ss    8:34AM    0:00.05 1
ogin -pf nishan
nishantkumar 27700    0.0    0.0    4334808    2900    ??    S    8:28AM    0:00.07 /
usr/libexec/nea
nishantkumar 27699    0.0    0.2    4367048    13576    ??    S    8:28AM    0:00.47 /
Library/Apple/S
nishantkumar 27696    0.0    0.0    4363332    3200    ??    S    8:27AM    0:00.07 /
System/Library/
nishantkumar 27693    0.0    0.1    4368028    8680    ??    Ss    8:27AM    0:00.09 /
System/Library/
root      27692    0.0    0.0    4362952    3548    ??    Ss    8:27AM    0:00.09 /
usr/libexec/con
nishantkumar 27680    0.0    0.3    4906112    25596    ??    Ss    8:25AM    0:02.78 /
System/Applicat
nishantkumar 27627    0.0    0.0    4330124    1652    ??    Ss    8:14AM    0:00.02 /
System/Library/
nishantkumar 27626    0.0    0.2    4907416    19548    ??    Ss    8:14AM    0:01.08 /
System/Library/
nishantkumar@192 ~ %
```

**STEP 8:**To save all the running processes to a txt file run the following command in the terminal:

*'ps aux > running\_processes.txt'*

This will save the list of running processes to a file named "running\_processes.txt" in the current directory.



## SNAPSHOTS

```
running_processes.txt
USER          PID  %CPU  %MEM    VSZ   RSS  TT  STAT  STARTED      TIME COMMAND
nishantkumar  27075  2.7   0.1  4338140 11000  ??  S    11:55PM    0:00.10 /System/Library/
Frameworks/CoreServices.framework/Frameworks/Metadata.framework/Versions/A/Support/mdworker_shared -s
mdworker -c MDSImporterWorker -m com.apple.mdworker.shared
nishantkumar  35944  1.9   3.1  39056428 262056  ??  S    10Mar23  360:48.10 /Applications/Google
Chrome.app/Contents/MacOS/Google Chrome
nishantkumar  16308  0.7   3.7  1190578372 310880  ??  S    6:40AM    10:32.06 /Applications/Google
Chrome.app/Contents/Frameworks/Google Chrome Framework.framework/Versions/110.0.5481.177/Helpers/
Google Chrome Helper (Renderer).app/Contents/MacOS/Google Chrome Helper (Renderer) --type=renderer --
lang=en-GB --num-raster-threads=2 --enable-zero-copy --enable-gpu-memory-buffer-compositor-resources
--enable-main-frame-before-activation --renderer-client-id=6397 --time-ticks-at-unix-
epoch=-1675283000259111 --launch-time-ticks=3773292744628 --shared-files --field-trial-
handle=1718379636,r,5317984445530243184,5913093049791061096,131072 --seatbelt-client=375
root          262    0.5   0.4   6080024 31104  ??  Ss   23Aug22  58:28.05 /System/Library/
Frameworks/CoreServices.framework/Frameworks/Metadata.framework/Versions/A/Support/mds_stores
root          96     0.4   0.2   4482628 13496  ??  Ss   23Aug22  80:35.11 /System/Library/
Frameworks/CoreServices.framework/Frameworks/Metadata.framework/Support/mds
nishantkumar  27054  0.1   0.1   4338140 11180  ??  S    11:53PM    0:00.12 /System/Library/
Frameworks/CoreServices.framework/Frameworks/Metadata.framework/Versions/A/Support/mdworker_shared -s
mdworker -c MDSImporterWorker -m com.apple.mdworker.shared
nishantkumar  313    0.1   0.0   4331228 2684  ??  S    23Aug22  12:39.74 /usr/sbin/cfprefsd
agent
nishantkumar  35956  0.1   1.0  38641220 86736  ??  S    10Mar23  789:52.20 /Applications/Google
Chrome.app/Contents/Frameworks/Google Chrome Framework.framework/Versions/110.0.5481.177/Helpers/
Google Chrome Helper (GPU).app/Contents/MacOS/Google Chrome Helper (GPU) --type=gpu-process --gpu-
preferences=UAAAAAAAAAAAgAAIAAAAAAAAAAAAAAAAAABGAAAAAAAAAwAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAALgFA
AAAAAAAAAuAAAAAAAAAQAAPAAAAAABAAAAAAAAA6AEAAAAAAAAADwAQAAAAAAAAAPgBAAAAAAAAAIAAAAAAAAAIAgAAAAAAAABACAAAAAA
AAGATIAAAAAAAAAAgAAAAAAAAACgCAAAAAAAAAAMATIAAAAAAAAAA4AgAAAAAAAAAEACAAAAAAAAASATIAAAAAAAAAABQAgAAAAAAAFgCAAAAAAAAYAI
AAAAAABoAgAAAAAAAHACAAAAAAAEATIAAAAAACAAGAAAAAAAIgCAAAAAAAAKATIAAAAAACYAgAAAAAAAKCAAAAAAAQATIAAAAA
AACwAgAAAAAAALgCAAAAAAAwATIAAAAAADIAgAAAAAAANCAAAAAAAATIAAAAAADgAgAAAAAAOgCAAAAAAA8ATIAAAAAAD4A
```

**STEP 9:**To display system information utility (**HARDWARE**) run the following command in the terminal:

*'system\_profiler SPHardwareDataType'*

This will display information about your system hardware, such as the processor, memory, and storage.

## SNAPSHOTS

```
Documents — -zsh — 80x24

Overview:
macOS
Protection:
[nishantkumar@192 Documents % system_profiler SPHardwareDataType
Hardware:

Hardware Overview:

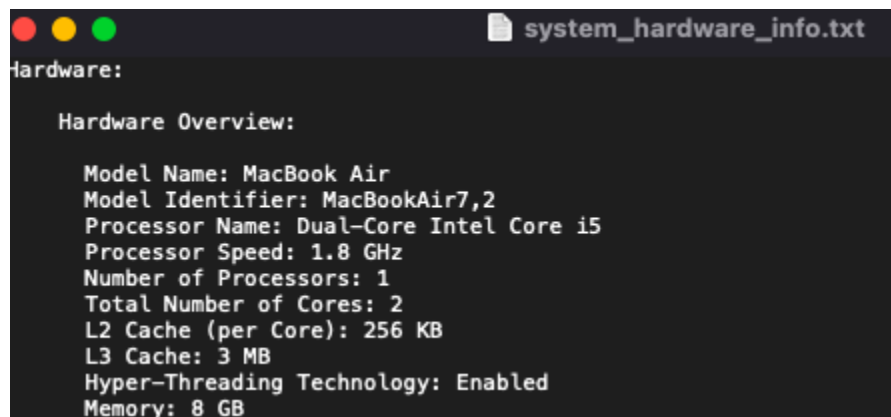
Model Name: MacBook Air
Model Identifier: MacBookAir7,2
Processor Name: Dual-Core Intel Core i5
Processor Speed: 1.8 GHz
Number of Processors: 1
Total Number of Cores: 2
L2 Cache (per Core): 256 KB
L3 Cache: 3 MB
Hyper-Threading Technology: Enabled
```

**STEP 10:**To save system information utility (**HARDWARE**) in a text file run the following command in the terminal:

```
'system_profiler SPHardwareDataType | tee system_hardware_info.txt'
```

This will save the output of the “system\_profiler” command to a file named “system\_hardware\_info.txt”

## SNAPSHOTS



```
Hardware:

  Hardware Overview:

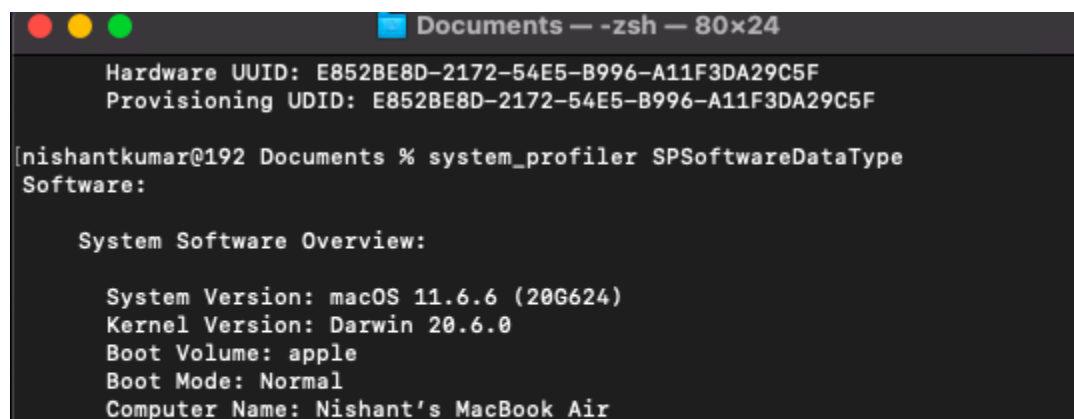
    Model Name: MacBook Air
    Model Identifier: MacBookAir7,2
    Processor Name: Dual-Core Intel Core i5
    Processor Speed: 1.8 GHz
    Number of Processors: 1
    Total Number of Cores: 2
    L2 Cache (per Core): 256 KB
    L3 Cache: 3 MB
    Hyper-Threading Technology: Enabled
    Memory: 8 GB
```

**STEP 11:**To display system information utility (**SOFTWARE**) run the following command in the terminal:

```
'system_profiler SPSoftwareDataType'
```

This will display information about your system software, such as the operating system version and installed applications.

## SNAPSHOTS



```
Hardware UUID: E852BE8D-2172-54E5-B996-A11F3DA29C5F
Provisioning UDID: E852BE8D-2172-54E5-B996-A11F3DA29C5F

[nishantkumar@192 Documents % system_profiler SPSoftwareDataType
Software:

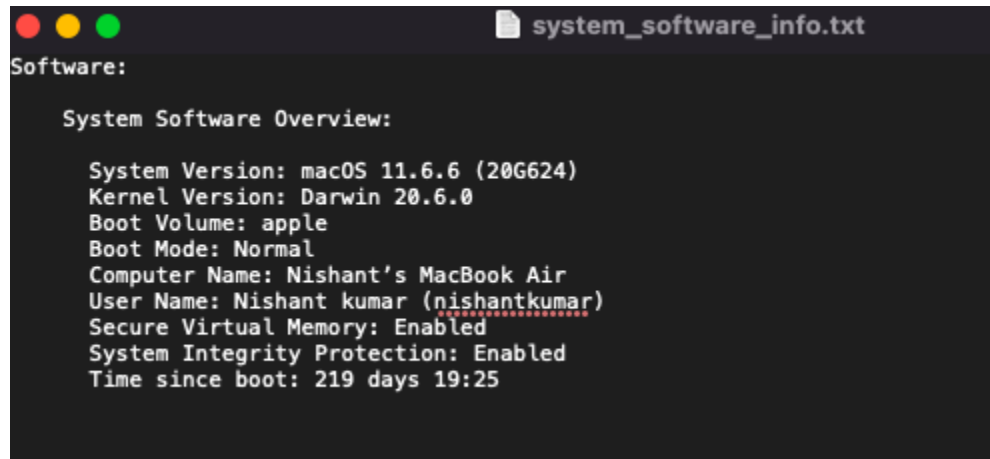
  System Software Overview:

    System Version: macOS 11.6.6 (20G624)
    Kernel Version: Darwin 20.6.0
    Boot Volume: apple
    Boot Mode: Normal
    Computer Name: Nishant's MacBook Air
```

**STEP 12:** To save system information utility (**SOFTWARE**) in a text file run the following command in the terminal:

```
'system_profiler SPSoftwareDataType | tee system_software_info.txt'
```

## SNAPSHOTS



```
Software:

  System Software Overview:

    System Version: macOS 11.6.6 (20G624)
    Kernel Version: Darwin 20.6.0
    Boot Volume: apple
    Boot Mode: Normal
    Computer Name: Nishant's MacBook Air
    User Name: Nishant kumar (nishantkumar)
    Secure Virtual Memory: Enabled
    System Integrity Protection: Enabled
    Time since boot: 219 days 19:25
```

**STEP 13:** To save all these processes in the final report txt file . Displaying system information utility , Displaying the mac product key and id , Displaying the list of installed from 3rd party aswell and DisplayING all the running processes.

```
'cat system_info.txt product_key.txt installed_apps.txt running_processes.txt > final_report.txt'
```

This command will concatenate the contents of the four text files I have created earlier into a single file named "final\_report.txt". The ">" symbol is used to redirect the output of the "cat" command to the file named "final\_report.txt".

## SNAPSHOTS

```
final_report.txt

Accessibility:

  Accessibility Information:

    Cursor Magnification: Off
    Display: Black on White
    Flash Screen: Off
    Mouse Keys: Off
    Slow Keys: Off
    Sticky Keys: Off
    VoiceOver: Off
    Zoom Mode: Full Screen
    Contrast: 0
    Keyboard Zoom: Off
    Scroll Zoom: Off

Applications:

  iStat Server:

    Version: 3.03
    Obtained from: Identified Developer
    Last Modified: 03/08/17, 3:02 AM
    Kind: Intel
    Signed by: Developer ID Application: Bjango Pty Ltd (Y93TK974AT), Developer ID Certification
    Authority, Apple Root CA
    Location: /Applications/iStat Server.app

  Python Launcher 3:
```

```
final_report.txt

  Signed by: Developer ID Application: JetBrains s.r.o. (2ZEFA8TH3), Developer ID Certification
  Authority, Apple Root CA
  Location: /Applications/PyCharm CE.app
  Get Info String: PyCharm 2022.2.1, build PC-222.3739.56. Copyright JetBrains s.r.o., (c)
  2000-2022

  XProtect:

    Version: 93
    Obtained from: Apple
    Last Modified: 17/03/23, 6:25 PM
    Kind: Universal
    Signed by: Software Signing, Apple Code Signing Certification Authority, Apple Root CA
    Location: /Library/Apple/System/Library/CoreServices/XProtect.app

  VLC:

    Version: 3.0.17.3
    Obtained from: Identified Developer
    Last Modified: 12/06/22, 4:39 PM
    Kind: Intel
    Signed by: Developer ID Application: VideoLAN (75GAHG3SZQ), Developer ID Certification
    Authority, Apple Root CA
    Location: /Applications/VLC.app

  Android File Transfer:

    Version: 1.0.12
    Obtained from: Identified Developer
    Last Modified: 16/10/18. 11:38 AM
```

**4.References-** [www.google.com](http://www.google.com)  
[www.youtube.com](http://www.youtube.com)

