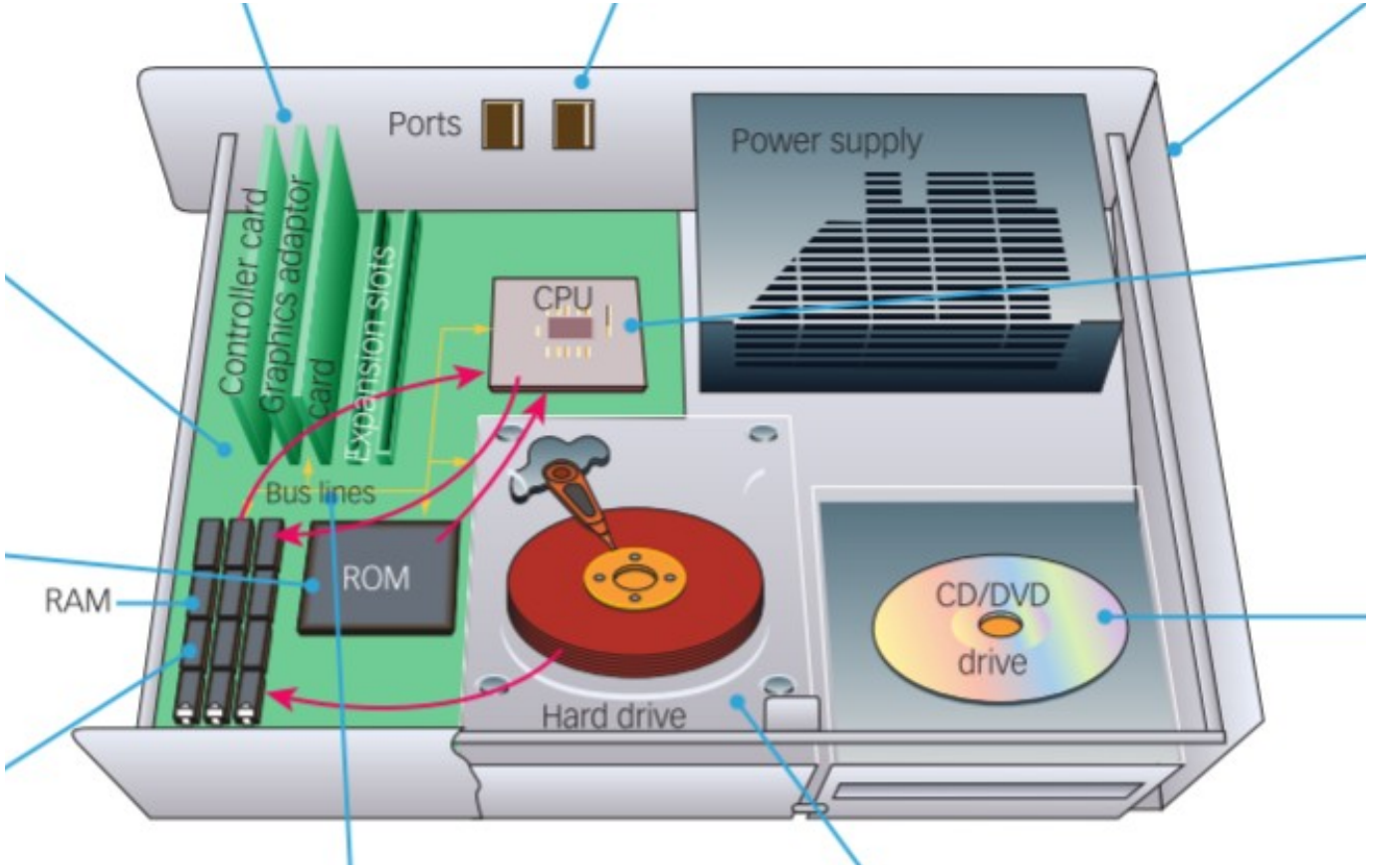
Day 1

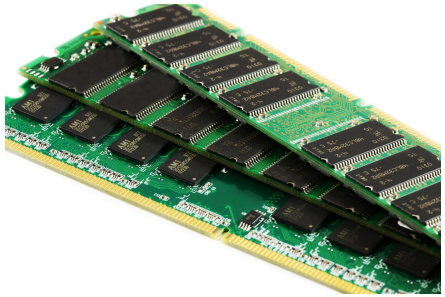
Basics

What’s inside your PC?

Simplified


Ports
Power supply
Controller card
Graphics adaptor
card
Expansion slots
CPU
Bus lines
ROM
RAM
CD/DVD
drive
Hard drive

What does it do?



Stores important information
Anything that is needed now

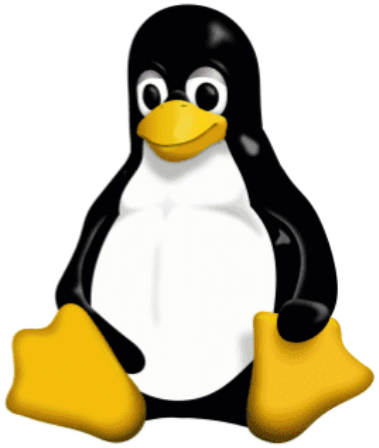
Runs calculations
Instructs other components

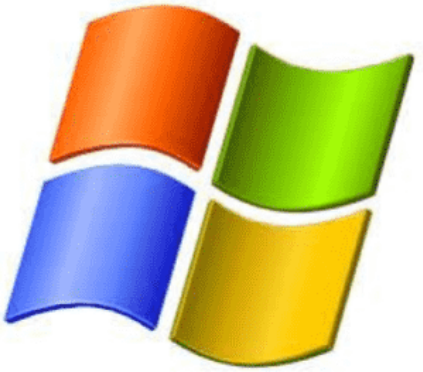
Output to user

Stores information:
Anything that may be needed later

Input from user

Operating Systems





But all OS’s do the same

Manage components

Assign CPU time to programs

Give user an interface: GUI

(Obvious differences in the latter)

Terminals

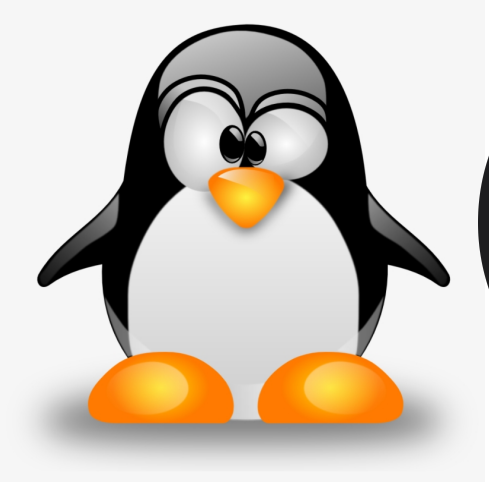

```
merlin@mls-pc  Beispiel  l                                              14:54:23
artikel/   artikel3/  artikel6@    artikel.iso  artikel.mp4  artikel.ps
artikel2/  artikel4@  artikel.doc  artikel.mp3  artikel.png  artikel.sh*

merlin@mls-pc  Beispiel  cd ~/.oh-my-zsh                                14:54:24

merlin@mls-pc  .oh-my-zsh  master ✔  touch test                         14:54:31

merlin@mls-pc  .oh-my-zsh  master ✔ ✱  rm test                          14:54:38
rm: reguläre leere Datei 'test' entfernen? y

merlin@mls-pc  .oh-my-zsh  master ✔  cd plugins/                        14:55:13
cache/      lib/        plugins/    themes/
custom/     log/        templates/  tools/
```

```
PS C:\> Get-ChildItem 'MediaCenter:\Music' -rec |
>>        where { -not $_.PSIsContainer -and $_.Extension -match 'wma|mp3' } |
>>        Measure-Object -property length -sum -min -max -ave
>>


Count    : 1307
Average  : 5491276.09563887
Sum      : 7177097057
Maximum  : 22905267
Minimum  : 3235
Property : Length


PS C:\> Get-WmiObject CIM_BIOSElement | select biosv*, man*, ser* | Format-List

BIOSVersion  : {TOSCPL - 6040000, Ver 1.00PARTIBL}
Manufacturer : TOSHIBA
SerialNumber : M821116H


PS C:\> ([wmiSearcher]@'
>> SELECT * FROM CIM_Job
>> WHERE Priority > 1
>> '@).get() | Format-Custom
>>

class ManagementObject#root\cimv2\Win32_PrintJob
{
```



```
Macintosh HD — top — 80×24
Processes: 210 total, 2 running, 9 stuck, 199 sleeping, 901 threads    23:30:03
Load Avg: 1.40, 1.75, 1.00  CPU usage: 4.15% user, 4.40% sys, 91.44% idle
SharedLibs: 1648K resident, 0B data, 0B linkedit.
MemRegions: 31278 total, 1892M resident, 117M private, 564M shared.
PhysMem: 5893M used (1191M wired), 10G unused.
VM: 523G vsize, 1026M framework vsize, 0(0) swapins, 0(0) swapouts.
Networks: packets: 12105/8925K in, 11907/1964K out.
Disks: 80156/2205M read, 21235/425M written.

PID  COMMAND      %CPU TIME     #TH  #WQ  #PORT MEM    PURG   CMPR PGRP PPID
592  screencaptur 0.0  00:00.02 7    5    55+   1952K+ 20K+   0B   262  262
590  mdworker     0.0  00:00.01 3    0    44    2032K  0B     0B   590  1
589  mdworker     0.0  00:00.01 3    0    44    1572K  0B     0B   589  1
588  top          1.7  00:00.51 1/1  0    22+   2860K  0B     0B   588  584
584  bash         0.0  00:00.00 1    0    15    588K   0B     0B   584  583
583  login        0.0  00:00.01 3    1    28    1228K  0B     0B   583  482
574  auditd       0.0  00:00.00 2    0    25    560K   0B     0B   574  1
567  System Prefe 0.0  00:03.23 3    0    270   39M    8364K  0B   567  1
561  systemstatsd 0.0  00:00.01 2    1    19    1040K  0B     0B   561  1
560  com.apple.We 0.0  00:01.42 9    0    229   25M    0B     0B   560  1
558  com.apple.We 0.0  00:05.07 15   3    224   151M   1716K  0B   558  1
555  bash         0.0  00:00.00 1    0    15    604K   0B     0B   555  554
554  login        0.0  00:00.01 3    1    28    1176K  0B     0B   554  482
550  bash         0.0  00:00.00 1    0    15    608K   0B     0B   550  549
```

Your terminal



Look for “Terminal”
Language: Bash (or usually similar)

Look for “Powershell”
Language: Powershell

Unix-Like vs Windows

Different file organization

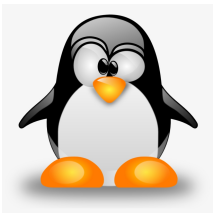
/home/user/file

C:\\User\folder\file

Different language

Bash (1989) is quite popular

Powershell (2006) is a newcomer



Change directories	cd {directory}	Set-Location {directory}, cd
List files in directory	ls	Get-ChildItem, ls, dir
Print current location	pwd	Get-Location
Make a new directory	mkdir {directory}	New-Item -Path '{directory}' -ItemType Directory
Make a new file	touch {file}	New-Item '{file}' -ItemType File
Copy a file	cp {file} {copy}	Copy-Item {file} {copy}
Remove an item (Be **very** careful!)	rm {file}	Remove-Item {file}
Move/Rename a file	mv {old} {new}	Move-Item {old} {new}

Try it out!

- Navigate around your file system
- Make some files, move them around, delete them
- Please don’t delete anything else! ;)

- Congratulations! You are writing code.

Starting out Python

- Python is a language, just like Bash
- There are different ways of using it
- Use the Python command (OS-dependent) to get to the interactive shell
- Type in “ print(‘Hello world!’) “
- Congratulations! You just wrote Python code.

The Interactive Shell

+ Quick

+ Easy

- Have to keep track of things

- No saving your code

Scripts

Good for *serious* programing

You save your script

Once script is done running, clean exit

Running a script

{Python} script.py will run the script

Executes line after line

Prints output to terminal

Editors

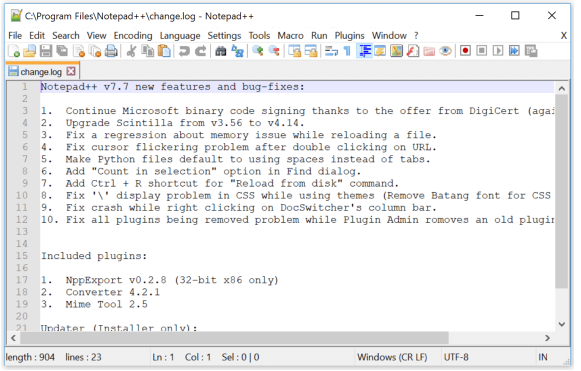
Are just tools to write text

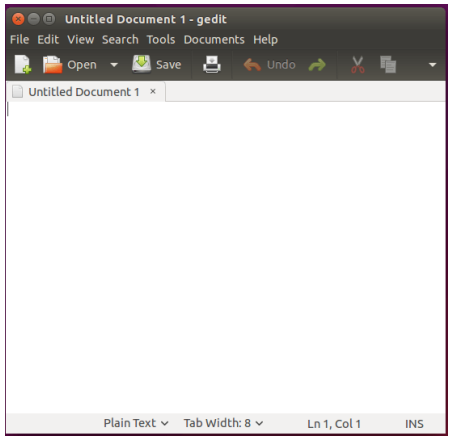
Plain text

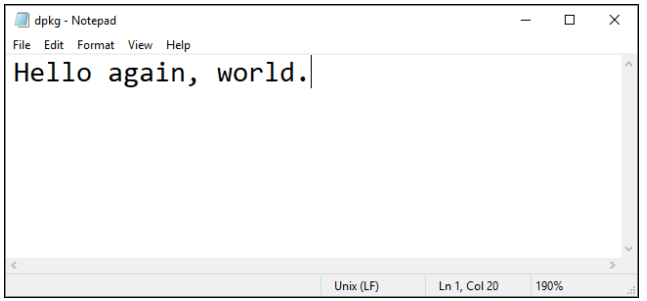
Often highlighting for language

Notepad (Windows)

Gedit or any of hundreds in Linux







IDEs

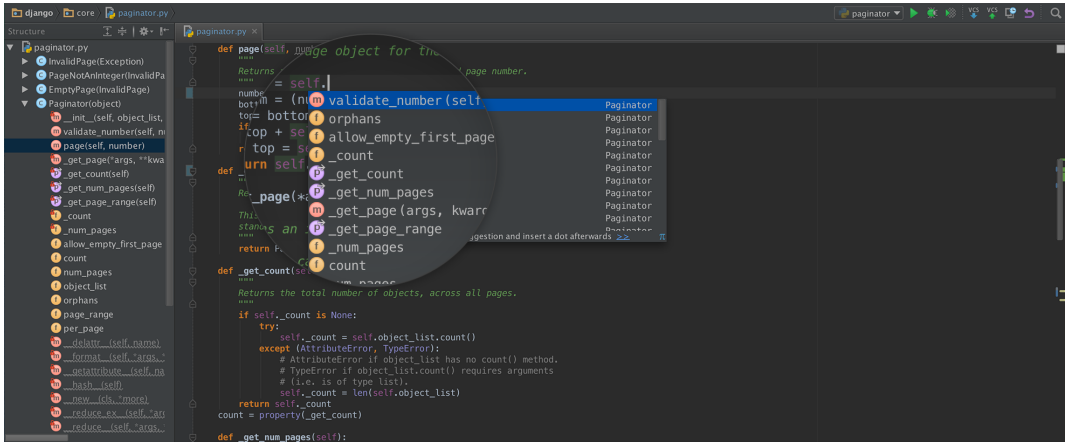
(Integrated Development Environments)

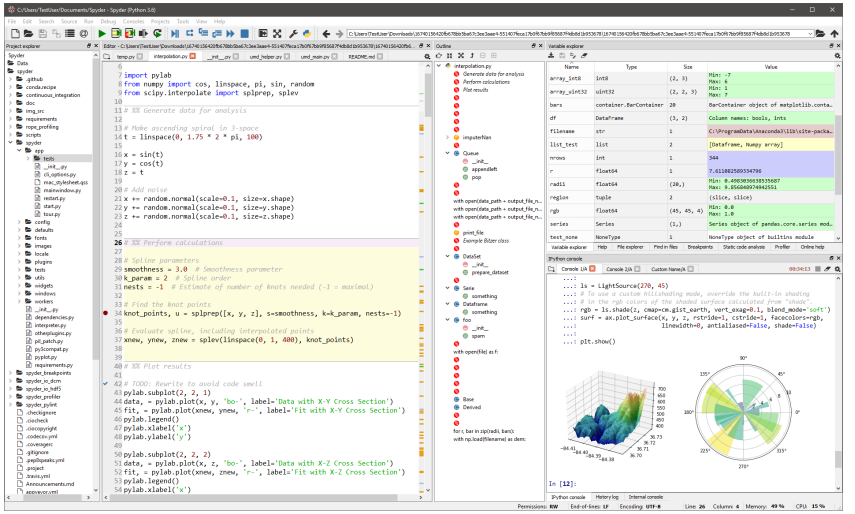
Basically an Editor with more features

Syntax highlighting

Code feedback

Often finds Errors





Your choice

Try out a few IDEs or editors.
Which do **you** like best?

A few suggestions:

IDEs: Spyder, PyCharm, VSCode, Ninja

Editors: Notpad++ (Windows), Geany (Linux), Vim