

Plan des sujets abordés en OC Informatique 4^{ème} année 2019 - 2020

Ce plan est souple et peut changer, en fonction de vos demandes et de vos intérêts.

1) Introduction

2) Les logiciels libres

- L'origine de GNU/Linux, (re)-naissance des logiciels libre.
- Divers classes de logiciels, propriétaires (ou privateur), shareware, freeware, open source et libre.
- Les 4 libertés fondamentales des logiciels libres.
- Les trois systèmes d'exploitations principaux et leur avantages / désavantages : GNU/Linux, Windows, Mac.
- Les interfaces graphiques de bases de GNU/Linux : KDE, Gnome, Xfce, Lxde, ...
- **L'environnement KDE sous GNU/Linux**. Une fois que vous aurez reçu votre clé USB.
- Information sur la structure des dossiers.
- Quelques logiciels libres.

3) Installation de GNU/Linux Ubuntu sur clé USB

4) Traitement d'images avec GIMP

5) Initiation au HTML et aux pages Web

6) Initiation au CSS, complément du HTML

7) Initiation à la programmation avec Javascript

Quelques informations

- Plusieurs dizaines de feuilles seront distribuées → ayez un **classeur** !
- Il y aura des feuilles de cours, comme celle-ci et des feuilles d'exercices nommées "Série ..."
- Page Web avec des informations supplémentaires :
<http://www.juggling.ch/gisin>
ATTENTION d'écrire cette adresse URL en minuscule.

Nous travaillerons généralement sur une clé USB 64GB, sur lequel le système d'exploitation GNU/Linux KUbuntu 18.04 64 bits est installé.

Je vous fournirai cette clé USB déjà installée, avec divers logiciels. Elle vous coûtera 15.-CHF.

Une partition de 16 GB sera utilisable sous tous les systèmes d'exploitations, GNU/Linux, Windows et Mac. Cela sera très utile pour transférer des données d'un ordinateur ou système à un autre.

En décembre, **la semestrielle** durera 160 minutes et comptera pour 50% de la note du semestre.

En juin, **la semestrielle** sera un oral de 20 minutes et comptera pour 1/3 de la note du semestre.

L'origine de GNU/Linux, (re)-naissance des logiciels libres.

Au début des années 1980, Richard Stallman, informaticien au MIT, frustré de ne pas avoir accès au code source qui pilote une imprimante qui ne fonctionne pas comme il le désirait, crée la Free Software Foundation (FSF) et le projet GNU (GNU Is not Unix).

L'objectif est de développer un système d'exploitation similaire à UNIX, qui respecte quatre libertés fondamentales selon Richard Stallman.

Sources :

"Richard Stallman et la révolution du logiciel libre, une biographie autorisée", de Richard M. Stallman, Sam Williams et Christophe Masutti, édition EYROLLES, 2010. ISBN: 978-2-212-12609-9.

http://fr.wikipedia.org/wiki/Richard_Stallman

Au début de l'informatique, il était courant de recevoir le code source des programmes qui pilotaient divers matériels informatiques. Puis est venue la notion de logiciels propriétaires (ou privateurs). La notion de logiciel libre a été formalisée par la FSF, créée par Richard Stallman.

Divers classes de logiciels, distinction entre :

1. Logiciel propriétaire (ou privateur), fermé (de source non disponible), payant.
 2. Logiciel shareware, fermé, payant, mais pouvant être testés gratuitement.
 3. Logiciel freeware, fermé, gratuit.
 4. Logiciel open source, ouvert, mais sans garantie qu'il restera ouvert.
 5. Logiciel libre, ouvert, avec garantie par la licence GPL qu'il restera ouvert. Source disponible.
- Il n'y a presque plus de différence entre « l'open source » et « le libre ».

Aucun des deux n'est imposé d'être gratuit, même si dans les faits, ces logiciels sont très souvent gratuits.

C'est par les services accompagnant le libre, que l'argent est gagné.

Les 4 libertés fondamentales des logiciels libres :

- la **liberté d'exécuter** le programme, pour tous les usages (liberté 0) ;
- la **liberté d'étudier** le fonctionnement du programme, et de le modifier pour qu'il effectue vos tâches informatiques comme vous le souhaitez (liberté 1),
l'accès au code source est une condition nécessaire ;
- la **liberté de redistribuer** des copies, donc d'aider votre voisin (liberté 2) ;
- la **liberté de modifier** et le programme et de distribuer des versions modifiées (liberté 3),
en faisant cela, vous donnez à toute la communauté une possibilité de profiter de vos changements ; l'accès au code source est une condition nécessaire.

Ces libertés sont associées à une **contrainte**. Si vous distribuez des copies, modifiées ou non, vous devez donner également le code source et les 4 libertés ci-dessus aux destinataires.

En 1990, Linus Torvalds, étudiant finlandais en informatique, crée un logiciel lui permettant d'accéder aux ordinateurs de l'université. Ce logiciel connu rapidement un grand succès auprès d'autres informaticiens. Il se combina avec le projet GNU de Richard Stallman, pour donner le système d'exploitation GNU/Linux.

Par la suite, de nombreux environnements graphiques furent écrits pour rendre ce système plus convivial.

Source : http://fr.wikipedia.org/wiki/Linus_Torvalds

Richard Stallman et Linux Torvald, deux acteurs fondamentaux des logiciels libres et open-source, ont des idéologies différentes. Pour le premier, il faut se battre pour obtenir les 4 libertés fondamentales et rejeter les logiciels non libres.

Pour le second, la fonctionnalité prime sur la liberté.

<http://www.gnu.org/philosophy/free-sw.fr.html> et <http://www.gnu.org/philosophy/words-to-avoid.html>

Les trois systèmes d'exploitations principaux et leurs avantages / désavantages :

- GNU/Linux
Libre, Open source, gratuit, trop de liberté, ce qui implique énormément de choix à faire, si une autre personne ne les a pas faites pour vous.
Très bien sécurité à la base, et pas encore suffisamment d'utilisateurs, donc pas de virus.
Malgré l'absence actuelle de virus sous Linux, il faut faire attention et ne pas installer n'importe quoi sur son ordinateur.
- Windows
Propriétaire, payant, peu de liberté donc moins de choix à faire, ce qui simplifie la vie de l'utilisateur en restreignant ses possibilités.
Mal sécurité à la base, donc sensible aux virus.
- Mac.
Propriétaire, payant, peu de liberté donc moins de choix à faire, ce qui simplifie la vie de l'utilisateur en restreignant ses possibilités.
Très similaire à Windows, avec encore moins de liberté, mais si on reste dans le système Mac, il fonctionne très bien, les utilisateurs en sont généralement très satisfait.
Mieux sécurité que Windows à la base, mais comme il se veut très simple d'utilisation, il devient sensible aux virus.

Les interfaces graphiques de bases de GNU/Linux :

- Gnome 3.0
- KDE
- Xfce
- Lxde, ...

L'environnement KDE sous GNU/Linux

En 2019, au Département de l'Instruction Publique, c'est l'environnement **KDE sur GNU/Linux Ubuntu16.04** qui est utilisé.

Nous utiliserons l'environnement **KDE sur GNU/Linux KUbuntu 18.04** 64 bits.

L'environnement KDE sous GNU/Linux

Nous verrons plus loin comment installer GNU/Linux et l'environnement KDE. D'autres installations sont similaires.

Un avantage de GNU/Linux est qu'il peut s'installer sur une clé USB. On peut aussi facilement cloner une clé USB. Cela permet d'effectuer de nombreux essais, sans craindre des erreurs, au pire, il faut cloner une nouvelle fois la clé ayant des erreurs irréparables.

L'installation que vous avez reçue sur une clé USB contient déjà de nombreux logiciels pré-installés, accessible depuis le menu en bas à gauche. L'image correspond à la tête d'un petit rat, symbole de vitesse.

Le "**Gestionnaire de paramètres**" permet de personnaliser votre système.

Sur le bureau, se trouve l'icône "**Répertoire personnel**". Double cliquez dessus.

Sur le bureau, se trouve l'icône "**Double commander**". Double cliquez dessus.

Les deux logiciels sont des gestionnaires de fichiers. Je préfère le second, qui est plus complet et donne plus d'options.

Information sur la structure des dossiers

/dos est le répertoire qui contient les fichiers que vous verrez si vous utilisez votre clé sous Windows ou sous Mac. Si vous branchez votre clé sur un autre ordinateur fonctionnant sous GNU/Linux, vous verrez deux répertoires, dont un qui correspondra à celui-ci.

/home/bg est votre répertoire principal. `/home` est standard, `"bg"` a été choisi par moi-même. Tous vos documents et paramètres personnels se trouveront dans ce répertoire.

/media contient le dossier correspondant à un CD, un DVD, une clé USB ou un média que vous aurez connecté à votre ordinateur.

/usr/bin contient les applications

/usr/share contient des données liées aux applications que vous avez installées. Souvent c'est là que l'on cherche une information particulière.

/usr/share/icons contient beaucoup d'icônes d'applications.

/boot contient les informations de démarrage du système et en particulier les "noyaux linux".

Il existe plusieurs manières d'installer des logiciels sous Ubuntu.

1) **Discover**, une liste de logiciels pour Ubuntu.

C'est probablement la méthode la plus simple, mais certains logiciels ne s'y trouvent pas par défaut. Il est possible d'ajouter des "dépôts", qui contiennent d'autres logiciels qui seront ainsi facile à installer.

2) **Système > Gestionnaire de paquets Synaptic**.

C'est une autre méthode, assez simple d'installation de logiciels, qui est plus complète que la précédente, mais moins clair. Les choix qu'on doit faire ne sont pas toujours simple.

3) **sudo apt install nom_du_paquet**

C'est la méthode la plus souple, la plus complète et aussi celle qui demande le plus de connaissances. Il faut utiliser un "Terminal" et écrire toutes les instructions en mode texte.

Il faut connaître le nom du paquet que l'on désire installer.

4) **En téléchargeant un paquet Debian : .deb**

Parfois, il est possible de télécharger un paquet `.deb`, de le sauvegarder sur le disque, puis de l'installer à l'aide de "Discover" d'Ubuntu, ou en tapant `"sudo apt ./nom_fichier.deb"` dans un Terminal.

5) **En téléchargeant directement un binaire exécutable**

Parfois, un binaire exécutable est directement disponible dans un fichier compressé. Il suffit de la décompresser dans un dossier. Un exemple est "Double Commander"

6) **En téléchargeant un installateur .sh**

Parfois, un fichier d'installation est téléchargeable et exécutable, pour qu'il se charge d'installer le logiciel. Un exemple est "NetBeans".

7) **En compilant les sources du logiciel**

C'est la méthode la plus compliquée, mais celle qui correspond à la dénomination "Logiciel Libre". On télécharge les sources d'un logiciel, souvent écrits en langage C, puis on compile ces sources. Généralement la procédure à suivre est indiquée. Il faut quelques connaissances pour qu'elle fonctionne.

Le gros avantage de cette méthode est qu'une personne ayant quelques connaissances en programmation, peut modifier le code source, corriger des erreurs, ajouter des fonctionnalités ou personnaliser le logiciel, avant de le compiler.

Citons **quelques exemples de logiciels libres**. Il en existe des dizaines de milliers !

L'évolution des logiciels libres est souvent beaucoup plus rapide que celle des logiciels propriétaires.

Ceux débutant par un + sont installés sur votre clé USB.

+x indique que le logiciel n'est pas libre, mais utilisable gratuitement et très utile.

Ceux en gras seront utilisés ou sont très utiles à mon avis.

+ **GNU/Linux** est un système d'exploitation libre.

Il existe en des centaines de distributions de ce système d'exploitation, accompagnés logiciels.

Quelques exemples : Ubuntu, Redhat, SUSE, Mandriva.

+ **KDE** est l'environnement graphique que nous utiliserons.

+ XFCE est un environnement graphique rapide que j'utilisais les années précédentes.

+ **Double Commander**, est un gestionnaire de fichiers, très recommandable.

Comme Gnome commander, Kruzader, et Midnight commander, ce sont des gestionnaires de fichiers. « La » référence est « Total Commander », qui fonctionne sous Windows, mais également sous GNU/Linux, par l'intermédiaire de WINE.

+ **Midnight Commander (MC)**, est un gestionnaire de fichiers en mode texte.

+x **Total Commander**, LA référence des gestionnaires de fichiers multi-fonction. Utilise WINE.

+ **Dolphin**, le gestionnaire de fichiers par défaut de KDE.

◦ Thunar, le gestionnaire de fichiers par défaut de XFCE.

◦ Nautilus, le gestionnaire de fichiers par défaut de Gnome.

+ **Libreoffice** est une suite libre comparable à la suite Microsoft Office.

+ **Firefox** est un navigateur Web.

◦ Thunderbird, est un gestionnaire d'e-mail et de carnet d'adresses.

+ **SeaMonkey**, est un cousin de Firefox et Thunderbird, qui intègre un navigateur Web, un gestionnaire d'e-mail, un carnet d'adresses et un éditeur de pages Web.

+ **BlueGriffon**, est un éditeur de pages Web. À malheureusement des défauts.

+ **Netbeans** est un éditeur de pages Web, très utile pour coder en HTML, CSS et javascript.

+ **Geany**, est un éditeur de textes très rapide, que je trouve utile pour voir du code HTML.

+ Kate, est l'éditeur de textes par défaut de KDE.

+ leafpad, gedit, sont d'autres éditeurs de textes.

+ Bluefish, un éditeur de pages HTML.

◦ Emacs, vi, sont deux éditeurs de textes standards sous Unix.

+ **Filezilla** pour transférer ses fichiers sur un serveur (FTP)

+ VLC permet la visualisation de vidéos.

+ **GIMP** est un éditeur d'images et de photos.

+ Mirage, pour visualiser des images.

+ Audacity est un éditeur de musiques et de sons.

+ Lame est utilisé pour sauvegarder des sons au format .mp3

+ Avidemux est un éditeur vidéo, qui permet des conversions de formats et des retouches.

◦ Cinelerra, Openshot, Pitivi sont des éditeurs vidéo, que je connais très mal.

◦ Blender pour l'édition et la réalisation de vidéos en 3D

◦ Inkscape pour du dessin technique

◦ Scribus pour la publication d'affiches

+ GeoGebra, pour des constructions géométriques et des fonctions.

+ SciLab pour des calculs scientifiques et des simulations. Cousin de MatLab.

◦ Octave est un autre cousin de MatLab, que je trouve plus compliqué que SciLab.

+ Maxima et wxMaxima pour du calcul symbolique. Cousin de Mathematica et Maple.

+ Stellarium, pour la visualisation des étoiles de notre galaxy.

◦ Celestia, pour une visite des astres de notre galaxy, à vitesse supra-lumineuse.

+ CodeBlocks, pour faire de la programmation en C++.

◦ Lazarus, pour faire de la programmation en pascal, style Delphi.

+ IDLE, pour la programmation en langage Python.

- + **WINE** (Wine Is Not an Emulator) pour exécuter des logiciels Windows XP sous GNU/Linux.
- PlayOnLinux est un complément à WINE, qui aide à installer des logiciels Windows.
- Crossover est une alternative payante, non libre, mais plus complète.
- + gParted pour gérer les partitions du disque dur.
- + Disques pour gérer les partitions du disque dur.
- + **Synaptic** pour gérer les paquets des logiciels.
- + HPLIP (HP Linux Imaging and Printing) pour l'aide à l'impression sur des imprimantes HP.
- + CUPS-pdf, pour imprimer dans des fichiers pdf.
- + Shutter, pour faire des copies d'écran ou de portions d'écran.
- + K3b, pour graver des CD et des DVD.
- + VirtualBox, pour émuler des ordinateurs, avec d'autres systèmes d'exploitation.

Pour plus de références et de descriptions, voici une de mes pages Web :

http://www.juggling.ch/gisin/linux/linux_logiciels.html

Le but de ce qui suit est d'indiquer comment **installer GNU/Linux XUbuntu 18.04 sur une clé USB**. L'installation sur un disque dur est similaire, voir plus simple. Cette version s'appelle une LTS (Long Term Support). Une telle version sort tous les 2 ans. "18.04" signifie que cette version est sortie en avril (04) 2018. L'avantage des versions LTS est qu'elles sont mises à jour durant 4 années, contre 9 mois pour les autres.

c.f.

- <https://www.youtube.com/> chercher "**Gisin Bernard**". J'explique dans deux vidéos comment faire l'installation sur une clé USB.
- <http://doc.ubuntu-fr.org/installation>
- http://doc.ubuntu-fr.org/tutoriel/installer_ubuntu_avec_le_live_cd
- <http://fr.openclassrooms.com/informatique/cours> voir Linux, puis 2) "Reprenez le contrôle à l'aide de Linux !"
- <http://doc.ubuntu-fr.org/debutant> pour l'utilisation de Ubuntu

Certaines clés sont beaucoup plus rapides que d'autres, ce qui a son importance ici. Vous pouvez voir certains tests ici : <http://www.juggling.ch/gisin/linux/usbkey.html>

Beaucoup de choix sont à notre disposition.

- Un choix standard est Ubuntu avec l'interface graphique Gnome.
- Le choix du DIP est **KUbuntu**, qui ressemble le plus à Windows d'un point de vue utilisateur.
- Edubuntu est un autre choix, qui utilise l'interface Gnome et propose de nombreux logiciels éducatifs.
- LUbuntu est un système minimal.
- XUbuntu était adapté à l'installation sur une clé USB, lorsqu'elles étaient plus lentes, car il est assez complet et relativement léger. Il s'exécute rapidement.

Il existe deux bibliothèques de logiciels sur lesquels la plupart des autres logiciels se basent.

- 1) GTK, qui est celle utilisée par Xfce et par Gnome et les logiciels standards associés.
- 2) Qt, qui est celle utilisée par KDE et les logiciels standards associés.

Lorsqu'il y a le choix, Qt est conseillé sous KDE, mais les deux bibliothèques fonctionnent sous KDE.

Pour débiter, il faut posséder un DVD bootable d'installation de XUbuntu 18.04.

Un fichier « gravable » sur DVD se trouve sous : <http://www.ubuntu-fr.org/telechargement>

Optez pour "Autres variantes" > KUbuntu

Choisissez l'emplacement de téléchargement, la Suisse par exemple.

"Plus d'options.."

Si votre ordinateur n'a pas plus de 4 Go de RAM (mémoire vive), utilisez le processeur 32 bits.

De nos jours, presque tous les ordinateurs ont plus de 4 Go, on installe donc une version 64 bits.

Une fois le fichier "kubuntu-18.04-desktop-amd64.iso" téléchargé, gravez-le sur un DVD.

Pour la suite, je suppose que vous avez le DVD en question, gravé.

Procédure d'installation :

Des compléments d'informations se trouvent sous : http://doc.ubuntu-fr.org/tutoriel/installer_ubuntu_avec_le_live_cd

- Démarrer (booter) sur le DVD
 - Dès qu'une image apparaît au bas de l'écran, presser sur la touche espace. Image = 
 - Insérer la clé USB
 - Choisir la langue, Français
 - Presser sur F3, pour choisir le clavier Swiss French
 - Sélectionner le menu : « Installer KUbuntu », presser ENTER ... attendre
 - Presser ENTER pour accepter la langue « Français » d'installation
 - Cocher la case « Installer ce logiciel tiers » en bas d'écran
 - Cocher la case « Télécharger les mises à jour pendant l'installation ».
 - Si la connexion Internet est par Wifi, il faut aller sur l'écran suivant, pour se connecter par Wifi, puis revenir en arrière, pour cocher la case de mises à jour.
 - Cocher la dernière puce « Autre chose » pour installer sur la clé USB, ENTER ... attendre
 - En bas de fenêtre, **sélectionner le lieu d'installation « /dev/sdb USB ... » !!!**
Il se peut que le lieu soit « /dev/sdc ... » s'il y a deux disques dur dans l'ordinateur.
 - Sélectionner « /dev/sdb1 » se trouvant sous « /dev/sdb » et cliquer sur le bouton « - » pour effacer la partition actuelle de la clé USB (sdb). S'il y a une deuxième partition sur la clé, l'effacer aussi.
 - Sélectionner « espace libre », sous « /dev/sdb » et cliquer sur le bouton « + » pour ajouter une première partition.
Taille = 1'000 Mo, Type = Primaire, Emplacement = Début,
Utiliser comme « système de fichiers FAT32 », Point de montage = « /dos ». VALIDER
Cela permettra de stocker des données lisibles sur les systèmes Linux, Windows et Mac.
 - Sélectionner « espace libre », cliquer sur le bouton « + » pour ajouter une deuxième partition.
Taille = inchangée, sera maximal, Type = Primaire, Emplacement = Début,
Utiliser comme « système de fichiers ext2 », Point de montage = « / ». VALIDER
C'est là que le système XUbuntu sera installé.
« / » correspond à la racine de l'installation, « /home/vos_initiales » sera le dossier de vos fichiers.
 - Vérifier que le **lieu d'installation est bien « /dev/sdb USB... » !!!**
Si ce n'est pas fait, le disque dur ne démarrera plus sans la présence de la clé USB !
Il faudrait donc réparer le Master Boot Record (MBR) du disque dur si l'erreur est faite !
 - Cliquer sur « Installer maintenant » pour démarrer l'installation. Elle peut prendre 1 heure ou plus.
 - > L'installation commence, plusieurs questions vous sont encore posées :
 - « Où vous trouvez-vous ? », « Zurich », « Paris », « Bern », « Genève » sont quatre réponses possibles, c'est utile pour régler l'horloge interne de l'ordinateur.
 - Presser ENTER pour accepter le clavier : Allemand (Suisse) - Français (Suisse) + CONTINUER
Vous pouvez tester le clavier en cas de doutes.
 - . « Votre nom » : écrivez votre nom
 - . « Le nom de votre ordinateur » : je conseille un nom court, tel que « usb ».
 - . « Choisissez un nom d'utilisateur » : je conseille vos initiales, tel que « bg » pour moi.
Ces deux noms apparaîtront souvent, donc s'ils sont courts, ils prendront moins de place.
 - . Mot de passe. Il sera souvent demandé, je conseille un mot court, tel que « 1234 ». Le répéter.
Trop de sécurité tue la sécurité, vu qu'il est demandé trop souvent, je suggère un mot court.
 - . « Ouvrir la session automatiquement », sélectionner cette puce, + CONTINUER
 - Le message indiquant qu'aucune partition swap n'a été définie, continuez ...
 - Parfois il est demandé : « Choisissez une image », peu important.
- Attendre le temps de l'installation, qui peut durer plus d'une heure !
Une fois l'installation terminée, il est bon de redémarrer sur la clé USB.
Des mises à jour seront proposées. Il est bon de les faire, mais cela prend de nouveau beaucoup de temps, jusqu'à une heure.
- Ensuite, grâce à la "Discover", installer d'autres logiciels.

Une liste de logiciels qui m'intéressent se trouve à cette adresse URL :

http://www.juggling.ch/gisin/linux/linux_logiciels.html

J'ai mis en fin de cette page, deux fichiers, qui sont des fichiers "batch" qui installent automatiquement plusieurs logiciels.

Il est recommandé de lire le contenu de ce fichier texte. Il est facile à comprendre et à modifier.

Il faut commencer par installer le logiciel le premier de la liste ci-dessous, ensuite, au lieu d'installer à la main une liste de logiciels, on peut aller dans le répertoire contenant le fichier

"apt_1804_get_kde.sh", ouvrir un terminal en tapant par exemple "super T", puis de taper dans ce terminal : `sudo sh ./apt_1804_get.sh`

L'installation peut prendre une heure (encore une fois), mais beaucoup de logiciels seront installés.

Quelques logiciels qu'il est fortement conseillé d'installer (depuis la Logithèque Ubuntu) :

- Suppléments restreints Kubuntu (Même s'il installe des logiciels non libres, c'est très utile.)
- `gparted` permet la gestion des partitions, sera certainement utile une fois, peut-être dans plusieurs mois.
- `synaptic` permet l'installation d'autres paquets, c'est plus compliqué, mais il y en a plus
- `gnome-commander` un gestionnaire de fichiers, spécifique à gnome. Krusader en est un spécifique à KDE
- `gedit` c'est l'éditeur de texte utilisé par défaut par `gnome-commander` et `gnome`, il est très pratique.
- `leafpad` c'est un autre éditeur de texte, pas indispensable, mais standard.
- `geany` c'est encore un éditeur de texte, que j'aime bien pour de petites modifications de fichiers HTML.
- `filezilla` pour transférer des fichiers par FTP sur un serveur.
- `wine` pour exécuter des applications Windows sous Linux
- `vlc` pour voir des vidéos, sans problèmes
- `gnome-disk-utility` pour gérer des disques et clés usb
- `cups-pdf` pour imprimer dans un fichier .pdf, aucune imprimante n'est nécessaire
- `shutter` pour des captures d'écran ou d'une zone d'écran
- `lame` pour sauver des sons au format .mp3
- `audacity` pour éditer des sons et de la musique
- `avidemux` pour éditer et convertir des vidéos
- `mirage` pour visionner toutes les images d'un dossier
- `stellarium` un simulateur de planétarium, permettant de voir le ciel depuis la Terre
- `geogebra` pour faire de la géométrie et de l'algèbre interactive
- `maxima` et `wxMaxima` pour faire du calcul symbolique
- `libreoffice` avec `libreoffice-110n-fr` et `libreoffice-help-fr` l'office libre de référence
- `libreoffice-nlpsolver` pour faire de l'optimisation non linéaire dans `libreoffice Calc`
- . . .

Les logiciels ci-dessus représentent un tout petit échantillon des logiciels disponibles sous Linux.

Il existe aussi de nombreux jeux, dans le domaine du libre.

La vitesse de croissance du libre et de l'open source est remarquable.

Les logiciels libres se développent également pour Windows et Mac OSX.

GNU/Linux n'est pas parfait, mais remplace largement Windows, avec plusieurs avantages et quelques désavantages.

Traitement d'images avec GIMP

Le but de ce qui suit est de donner quelques informations de bases concernant les images en informatique et de s'initier au traitement d'images avec le logiciel GIMP.

En informatique, on distingue deux grandes classe d'images.

Les images **vectérielles** et les images **bitmap**.

Dans la suite, nous nous intéresserons exclusivement aux images **bitmap**.

Dans une page Web, de même que dans un document Libreoffice Writer, on insère souvent des images. Leur taille est souvent trop grande et la qualité inutilement bonne. Ceci a pour désavantage que la taille du fichier contenant l'image est grande. Le chargement d'une page Web contenant une telle image sera donc plus lent.

Quelques formats de fichiers stockant des images :

- .bmp** est un format standard sous Windows, qui est ancien et n'a plus beaucoup d'intérêt.
- .gif** (Graphics Interchange Format) est un format qui est beaucoup utilisé pour les images sur le Web. Il ne permet d'avoir que 256 nuances de couleurs, mais elles peuvent être choisies parmi les $16'777'216 (=256^3)$ nuances usuelles de couleurs en informatique.
- .png** (Portable Network Graphics) est similaire au format **.gif** et le remplace de plus en plus souvent, car il provient du monde du logiciel libre, il permet d'avoir plus de nuances de couleurs et les fichiers qu'il génère sont plus petits.
- .jpg** ou **.jpeg** (Joint Photographic Experts Group) est le format utilisé pour les photos.

Avantage et désavantage de divers formats :

Le format **.jpg** a l'avantage de compresser l'information contenue dans une image, ce qui rend le fichier correspondant plus petit. Le désavantage est que la compression se fait avec une perte de qualité d'image. L'utilisateur a le choix de la qualité qu'il désire conserver, en sachant que « meilleure qualité » signifie aussi « plus grand fichier ».

Ce format s'utilise pour stocker des photos. Il n'est pas adéquat pour stocker du texte, ni des dessins contenant de grandes zones de couleurs uniforme, comme certains logos.

Le format **.png** comprime aussi l'information, mais sans perte de données. Pour une même *photo*, celle stockée au format **.png** utilisera un fichier de plus grande taille que celle stockée au format **.jpg**. Ce format s'utilise pour stocker des images contenant du texte et/ou des dessins contenant de grandes zones de couleurs uniforme, comme certains logos.

Lorsqu'on écrit un texte sur une photo, il y a un conflit et souvent on stocke l'image au format **.jpg** avec une grande qualité, typiquement de 90 %.

Le format **.gif** est similaire au format **.png** et s'utilise dans les mêmes situations. En principe le format **.png** lui est préférable.

Le format **.bmp** ne comprime pas les images par défaut, il doit être évité.

Notion de dimension d'une image, de couleur et de résolution.

Une image **bitmap** est définie par un tableau de points, appelés « pixels » pour « PicturE LEment ». Typiquement une image peut contenir 800 lignes de 1200 points chacune, ou autrement dit, elle contient 1200 colonnes de 800 points chacune.

Chaque point est un mélange de trois couleurs : du Rouge, du Vert et du Bleu.

C'est le codage **RVB** ou en anglais : **RGB** (Red Green Blue).

Chaque pixel est défini par trois nombres entre 0 et 255. Le premier indique la quantité de **Rouge**, le second la quantité de **Vert** et le dernier la quantité de **Bleu**.

Rouge = `rgb(255, 0,0)` ; Vert = `rgb(0,255,0)` ; Bleu = `rgb(0,0,255)` ; cyan = `rgb(0,255,255)`.

Notation hexadécimale. `#ff0000` = `rgb(255,0,0)` ; `#4080c0` = `rgb(64, 128, 192)`.

Lorsqu'on imprime une image, il faut spécifier le nombre de points (=nombre de pixels) qu'il faut imprimer par centimètre. Vu que l'informatique est dominée par les États-Unies, on spécifie le nombre de points par pouce, ou « **dpi** » « Dot Per Inch ».

Ce nombre de points par pouce (ou par centimètre) s'appelle **la résolution de l'image**.

Il est utile de savoir qu'**un pouce (= un Inch) = 2,54 centimètres**.

Une résolution très courante est de 300 dpi.

Une résolution de 600 dpi est de très (trop) bonne qualité.

Les moniteurs d'ordinateur ont généralement une résolution légèrement inférieure à 100 dpi.

Question 1.

Si une image est de 1200x800 pixels et qu'on l'imprime avec une résolution de 300 dpi, quelle sera sa taille en centimètres ?

Manipulation 1.

- Chercher une photo, de dimension entre 600x400 et 1200x800 pixels et sauvegardez-la.
- Lancez le logiciel **GIMP**, ouvrez la photo.
- **Exportez-la** au format **.jpg** en ajoutant « _000 » à la fin de son nom, sauveez-la avec une qualité = 0.
- Recommencez en **exportant** la même photo avec les qualités suivantes : 10 ; 20 ; 30 ; 50 ; 70 ; 90 ; 95 ; 100. Nommez chaque fois le nom du fichier de manière adéquate.
- Regardez, dans le dossier contenant ces images, leur taille et visualisez-les pour voir leur qualité.

Manipulation 2.

- Dans **GIMP**, avec la photo précédente, allez dans « Image » > « Échelle et taille de l'image... »
- Définissez la « Largeur » et la « Hauteur » à 50 %, peu importe la résolution.
- Exportez votre nouvelle image avec une qualité de 90 %.
- Comparez la taille du fichier contenant la plus petite image à celle sauvée précédemment avec une qualité de 90 %.

Lorsque vous incluez des images dans vos pages Web, **il est important de diminuer la taille de l'image** pour l'ajuster à celle désirée. De plus, si c'est une photo, réduisez la qualité en fonction de ce qui est utile. Souvent on met une petite photo, de faible qualité, qui envoie vers la même photo, de plus grande taille et de meilleure qualité.

Manipulation 3.

- Cherchez une image contenant de grandes zones de couleur uniforme.
- Si l'image est au format **.svg**, elle est stockée sous un format vectoriel. Vous pouvez quand-même l'ouvrir avec GIMP, mais il vous sera demandé la largeur, la hauteur, les ratios et la résolution. Vous pouvez garder les dimensions par défaut.
- Exportez cette image au format **.pgn** et au format **.jpg** (qualité 90%) et comparez la taille des fichiers, ainsi que la qualité des images.

Manipulation 4.

- Dans **GIMP** créer une nouvelle image de taille 300x200.
- Faites un dessin quelconque.
- Effacer une partie du dessin, la couleur sera celle du fond de l'image, généralement blanc. On veut souvent que le fond soit transparent. Pour cela :
- Allez dans « Calques et brosses », fait un clique droit sur l'image miniature et ajoutez un « canal alpha »
- Sélectionnez la gomme et effacez une partie du dessin, vous voyez que même le fond s'efface. La région effacée est transparente.
- Toujours avec la gomme sélectionnée, « effacez » la partie transparente, en pressant en même temps sur la touche « Alt ». Le dessin qui était effacé réapparaît !

Une première page Web

Lorsqu'un navigateur reçoit une page Web, il reçoit en réalité un fichier contenant une série d'instructions qu'il interprète pour afficher la page telle que l'utilisateur la voit.

On nomme "**le client**", celui qui reçoit les pages Web.

Voici le fichier codant une page Web minimaliste :

```

1 <!DOCTYPE HTML>
2 <html lang="fr">
3 <head>
4 <meta charset="utf-8">
5 <title>Première page Web</title>
6 </head>
7 <body>
8 Ceci est ma première page Web.
9 </body>
10 </html>
```

Des numéros de lignes ont été ajoutés pour s'y référer, ils ne font pas partie du fichier.

La première ligne est toujours la même et indique que ce qui suit est de l'HTML 5.

HTML signifie "Hyper Text Markup Language", c'est un langage de description de pages Web.

Tous les textes qui sont entre les symboles < et > sont des **balises**, qui codent la manière d'afficher et de traiter le fichier, pour afficher la page Web.

Lignes :

2 <html> et 10 </html> codent le début et la fin du codage.

Si l'on veut spécifier la langue on peut remplacer <html> par <html lang="fr">

3 <head> et 6 </head> codent l'en-tête.

4 indique que le codage des caractères se fait en UTF-8.

Deux autres codages courant sont le iso-8859-15 et le windows-1252 utilisé sous Windows.

5 <title> ... </title> indique le titre de la fenêtre contenant la page web

6 </head> indique la fin de l'en-tête

7 <body> indique le début du contenu (du corps) de la page Web

8 "Ceci est ..." contient le contenu de la page Web, ici, il se limite à une seule ligne.

9 </body> indique la fin du contenu (du corps) de la page Web

10 </html> indique la fin du codage.

Les lignes 9 et 10 se trouvent toujours en fin de fichier.

Il est possible de taper les lignes qui précèdent dans n'importe quel éditeur de texte, qui permet de sauver les caractères en utilisant le codage utf-8. Sous Linux et Mac, c'est le standard.

Il est possible de taper une page Web avec un éditeur WYSIWYG (What You See Is What You Get), qui cache le code et montre directement le contenu qui sera affiché. Cela a des avantages de simplicité, mais également des désavantages de perte de contrôle de structure de codage.

Voici quelques logiciels d'édition WYSIWYG :

- **BlueGriffon**. Il modifie le code source HTML d'une page que l'on modifie en WYSIWYG.
- **Seamonkey**, qui est à la fois un navigateur Web, un gestionnaire d'e-mail et d'adresse et un éditeur de page Web. Il code de l'HTML 4.
Il modifie le code source HTML d'une page que l'on modifie en WYSIWYG.
- Kompozer, était très prometteur, mais n'est plus développé depuis mai 2009, dépassé.
- Dreamweaver, est développé par Adobe. Il est propriétaire, payant et assez cher, mais a une très bonne réputation.

Installation de Seamonkey

Seamonkey a l'avantage de pouvoir être installé facilement sur un ordinateur, sans avoir les droits d'administrations.

- 1) Télécharger du site Web : <https://www.seamonkey-project.org/> le fichier (l'archive) seamonkey-2.49.4.tar.bz2
- 2) En faisant un cliquer droit sur le fichier, ouvrez-le avec le gestionnaire d'archive.
- 3) Vous pouvez maintenant l'extraire dans le dossier Documents.
- 4) En allant dans le dossier Documents/seamonkey, vous pouvez double cliquer sur le fichier seamonkey, pour exécuter ce programme.

Votre site Web

Chacun d'entre vous a un petit site Web visible depuis le monde entier, sur lequel il peut stocker ces diverses créations.

! Le nom du site Web est : http://www.juggling.ch/xvotre_prénom
où *votre_prénom* est à remplacer par votre prénom.

Pour transférer des données sur votre site Web, vous avez besoin d'un logiciel et de trois données.

Le logiciel que je vous conseille s'appelle **Filezilla**, il est installé par défaut sur les ordinateurs du collègue.

Les trois données sont :

- 1^{ère} donnée est l'**Hôte**, qui est : `borj.ftp.infomaniak.com`
- ! 2^{ème} donnée est l'**Identifiant**, qui est : `borj_x` suivie de votre prénom, comme pour le nom de votre site web.
- 3^{ème} donnée est le **mot de passe**.

La donnée du Port n'est pas essentielle, on peut la laisser vide.

Tout doit être tapé en **minuscule**.

Un site pour le transfert ftp : <https://manager.infomaniak.com/ftp/>

Précédemment, une page Web minimaliste a été présentée. Le but de ce qui suit est de mettre en place un début de structure de site Web et d'écrire quelques pages Web, avec des liens, des images et une certaine mise en forme.

Structure de votre site Web

Un site Web est constitué de nombreux fichiers et dossiers.

Une page Web est créée à partir d'un fichier texte dans les langages **HTML** et **CSS**, qui décrivent le **contenu** et le **formatage** de la page Web.

Donc une page Web ne contient pas d'images ou de sons ou de vidéos, mais que du texte. Ce texte indique entre autre où se trouve chaque image.

HTML signifie "Hyper Text Markup Language"

CSS signifie "Cascading Style Sheet" « Feuille de Style en Cascade »

Pour développer et tester votre site Web, vous devez respecter une structure. Les fichiers que vous déposerez sur le serveur devront être placés dans le **dossier "web"** ou dans un de ses sous-dossiers.

Vous reproduirez cette structure sur votre ordinateur. Le **dossier "website"**, se trouvant sur votre ordinateur, contiendra l'image de ce qui se trouve sur le serveur.

Ce dossier doit contenir un sous-dossier **"images"** qui contiendra les images. (nom en **minuscules** !)

D'autres sous-dossiers ("sons", "styles", "videos" etc.) viendront plus tard.

Le dossier "website" doit aussi contenir un fichier "index.html".

Les noms de fichiers et dossiers sont constitués de lettres, chiffres et de "-" et de "_".

Ne mettez aucun espace ou accent dans ces noms de fichiers !

La page : index.html

Chaque site Web possède une page d'accueil qui se nomme généralement "index.html". C'est le point d'entrée dans un site Web, qui redirige vers d'autres sections du site.

Vous devrez donc avoir une page "index.html" qui aura pour but de diriger le visiteur vers d'autres pages de votre site Web.

Création de votre site Web

- Lancez « SeaMonkey » et allez dans le composeur (icône en bas à gauche) pour écrire votre première page Web. Une autre possibilité est de lancer « Bluegriffon ».
 - Sauvez votre première page Web dans « /home/bg/website » et nommez-la « index.html ». Vous devrez définir le titre de la page, qui est celui qui apparaît en haut de fenêtre.
 - Le fichier « index.html » est celui de la page d'accueil.
 - Écrivez une deuxième page Web, que vous nommerez « page2.html » et que vous stockerez dans le même dossier.
 - Créez un lien qui envoie de « index.html » sur « page2.html ».
 - Créez un lien dans l'autre sens.
 - Testez votre page Web, soit dans Firefox, soit sans SeaMonkey.
 - En cliquant en bas sur « <HTML> Source », vous pouvez voir le code source HTML. (Hyper Text Markup Language).
 - Dans le dossier "website/images", placez une image que vous voudrez voir dans "page2.html".
 - Dans « page2.html », insérez l'image du point précédent. Indiquez que le lien doit être relatif.
 - Regardez le code source HTML, pour vous rendre compte que l'image n'a pas été insérée, mais uniquement un lien sur l'image. En conséquence, les images qui apparaissent dans une page Web, doivent être stockées dans le sous dossier "images" du site Web et pas n'importe où dans l'ordinateur.
 - Sauvez, testez et regardez le code source.
 - Créez une "page3.html". Ayez des liens entre ces pages.
 - Pour s'y retrouver, il est utile d'indiquer en bas de votre site Web, sa structure et la date de la dernière modification de la page. Référez-vous à : http://www.juggling.ch/zgisin/a2019_oc3/index_oc4_2019.html
- Toutes vos pages doivent avoir l'indication de cette structure !

Transfert de votre site Web sur le serveur et tests

- Lancez « Filezilla », qui servira à transférer une copie de vos fichiers sur le serveur.
Hôte : juggling.ch
Identifiant : borj_x *suivit de votre prénom*
Mot de passe : c'est celui que je vous ai donné
Port : laissez-le blanc. (le port par défaut est 21. Il n'y a pas besoin de l'indiquer)
 Cliquez sur "Connexion rapide"
 À gauche, allez dans le dossier "website", à droite dans le dossier "web"
- La première fois, transférez tous les fichiers et dossier de "website" dans "web".
 Les fois suivantes, seules les modifications devront être reportées.
- Allez sur le site : [http://www.juggling.ch/x_suivit de votre prénom](http://www.juggling.ch/x_suivit_de_votre_prenom)
 Ce site est visible depuis le monde entier !
- **À la fin, DÉCONNECTEZ-VOUS, sinon l'accès FTP sera bloqué pour tout le monde !**

Balises HTML

Les balises HTML sont placée entre des « crochets » « < » et « > ».

Elles donnent des indications sur la manière d'organiser et partiellement d'afficher une page Web.
 Nous verrons plus loin que c'est par des instructions CSS que la forme de présentation sera codée.

-> Ouvrez votre page web « page2.html », éditez-la et visualisez le code « <HTML> Source ».

La structure minimale est :

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>Titre de la page.</title>
</head>
<body>
Texte de la page...<br>
</body>
</html>
```

Le texte en *italique* est arbitraire.

Votre texte diffère de celui qui se trouve ci-dessus. Pour l'instant, ce n'est pas important.

La première ligne informe votre navigateur que le document est de type « html », et quelle est la version utilisée. Par défaut, comme ci-dessus, c'est la dernière version qui est utilisée. Actuellement, en 2019, c'est de l'html 5 qui est utilisé.

<html> indique le début de codage en html. La dernière ligne, </html> indique la fin de la page.

<head> indique le début de l'en-tête, </head> indique la fin de l'en-tête.

<meta ...> donne des informations supplémentaires au navigateur concernant cette page Web.

L'information « charset="utf-8" » indique le type de codage des caractères.

<title> ... </title> indique qu'on donne le titre de la page Web, celui qui est affiché en haut de fenêtre.

<body> indique le début du contenu de la page Web et </body> la fin.

 indique un retour à la ligne.

<p> ... </p> indique un début et une fin de paragraphe.

 ... indique un lien sur une autre page ou dans une autre partie de la page.

 indique que l'image décrite doit être affichée dans la page Web.

Beaucoup d'autres balises existent, généralement pour la mise en forme et sont associée à un style CSS.

c.f. : <http://www.w3schools.com/html/default.asp> pour des balises html des attributs.

c.f. : <http://www.w3schools.com/tags/default.asp> pour la liste de toutes les balises de l'html.

Mise en forme avec le CSS

« CSS » signifie « Cascading Style Sheet », « Feuille de Style en Cascade »

L'idée est la suivante :

Dans le code HTML, on code l'information que l'on désire afficher dans la page Web et dans les instructions CSS on stocke la manière de présenter cette information.

Nous verrons 3 manières de définir le style (format) du texte dans une page Web avec le CSS.

- > Lancez SeaMonkey, dans : Édition > Préférence, Compositeur, cochez la case :
« Utiliser les styles CSS plutôt que les éléments et attributs HTML »
- > Éditez une de vos pages Web, ou créez-en une nouvelle et écrivez un petit texte, en mettant quelques mots en gras, d'autres en italique, une partie en souligné, certains en gras et italique.
 - Changez la taille de caractères.
 - Écrivez des titres (Format > paragraphe > titre 1) etc.

Ensuite, visualisez le code « <HTML> Source ».

Vous pourrez constater que pour mettre un texte en gras, l'instruction est :

```
<span style="font-weight: bold;">texte est en gras</span>
```

L'attribut « style » est la **première manière** de formater le texte.

Une ancienne manière est d'écrire `ce texte est en gras`, qui utilise la balise ``.

Il est conseillé de la combiner avec la deuxième manière de formater le texte en CSS.

Définition d'une feuille de style CSS pour modifier la présentation de la page Web

La **deuxième manière** utilise la balise `<style>` et terminent par `</style>`.

Elles se placent dans la section `<head> ... </head>`.

Les commentaires sont entre `/*` et `*/`, donc

```
<style> /* Ceci est un commentaire */ </style>
```

On peut définir le style d'un certain type de balise, exemple :

```
<style>
p { color:red } /* indiquera que le texte de tous les paragraphes sont en rouge (red) */
b { font-weight: bold; } /* c'est bien de l'écrire, même si c'est le comportement par défaut */
</style>
```

On peut définir une classe, définir son style et indiquer qu'une balise est de la classe donnée. Exemple :

```
<style>
.MonNomDeClasse { text-align:center; font-size:120%; font-weight:bold ; }
</style>
```

L'indication suivante, indiquera la classe de la balise :

```
<p class="MonNomDeClasse"> blablabla ... </p>
```

Donc le texte blablabla ... sera centré, de taille 120% et en gras.

c.f. <http://www.juggling.ch/gisin/bgweb/exemples.html> pour des exemples.

c.f. <http://www.w3schools.com/cssref/default.asp> pour la référence.

3^{ème} manière de définir les styles :

Si on désire définir plusieurs pages avec le même type de style, on peut définir les styles dans un fichier et l'inclure avec l'instruction :

```
<link rel="stylesheet" type="text/css" href="styles/nom_du_fichier_de_style.css">
```

en supposant que ce fichier se trouve dans le sous-répertoire « styles ».

Remarque :

Commentaires dans du code html : `<!-- commentaires -->`

Commentaires dans du code CSS : `/* commentaires */`

Commentaires dans du code javascript : `// commentaires sur une seule ligne.`

Allez sur ma page Web : http://www.juggling.ch/zgisin/a2019_oc3/index_oc4_2019.html

Allez sur ma "**page060_....html**", regardez son code source et sauvegardez-le chez vous sous le nom : "ex001_styles.html"

Lancez Netbeans et ouvrez ce fichier "ex001_styles.html".

Lisez et étudiez le code html de cette page.

Chaque balise peut avoir un attribut "style", qui permet de mettre en forme le texte.

Deux balises ont été introduites, juste pour mettre en forme des portions de pages.

1) La balise **** ne modifie en rien votre page Web. C'est le style qu'on lui donne qui modifiera la page. Exemples :

a) `rien ne change !`

b) `texte est en gras`

c) `celui-ci en italique`

2) La balise **<div>** crée un nouveau paragraphe. Elle impose un retour à la ligne.

Exemples :

`<div style="text-align:center">`

Autre essai de définition de style avec " text-align:center " dans un div.

`</div>`

`<div style="width:660px; border-style:solid; text-align:left; margin-top:5px; margin-right:8px; margin-bottom:5px; padding:5px 8px 5px 8px; ">`

Cette section est centrée et cardée. `
`

Référez-vous au code html du fichier que vous venez de sauver.

`</div>`

Créez une page `ex002_styles.html` pour faire d'autres essais !

Par exemple, ajouter un style à la balise `<body>` :

`<body style="background-color:rgb(255, 255, 128);">`

"rgb" indique la quantité de couleurs Red, Green, Blue utilisée. 0 = minimum, 255 = maximum.

Ce même style peut être ajouté à une balise ``, `<div>` ou n'importe quelle autre.

Exemple :

`spécialement`

Regardez les exemples de votre fichier "`ex001_styles.html`" et testez-en d'autres dans votre fichier : "`ex002_styles.html`"

Les quatre pages qui suivent donne de nombreux styles possibles et de nombreux styles possibles.

Netbeans vous aide en indiquant des suggestions, des erreurs et les positions des balises correspondantes à celle sur laquelle se trouve le curseur.

Mémento des propriétés CSS Source : "Réalisez votre SITE WEB avec HTML5 et CSS3", de Mathieu Nebra.
 Issu du Site OpenClassRooms, <http://fr.openclassrooms.com/>
 Pour beaucoup d'exemples, voir : <http://www.juggling.ch/gisin/bgweb/exemples.html>
 Pour une liste complète, c.f. : <http://www.w3schools.com/cssref/default.asp>

Propriétés de mise en forme du texte (et des images). C'est-à-dire de tout ce qui touche à la présentation du texte proprement dit : le gras, l'italique, le souligné, la police, l'alignement, etc.

Propriété	Description
font-family	Nom de police (Arial, Helvetica, Times, Courier, etc.)
@font-face	Police personnalisée
font-size	Taille de texte, typique est 12pt; ou font-size:150%; augmente de 50% la taille.
font-weight	font-weight:400; <=> normal, font-weight:700; <=> gras
font-style	font-style:normal; ou <i>italic</i> ; ou oblique;
text-decoration	text-decoration:none; ou <u>overline</u> ; ou <u>underline</u> ; ou line-through ;
font-variant	font-variant:normal; ou SMALL-CAPS;
text-transform	:none; :capitalize; :uppercase; :lowercase; change minuscules - majuscules
font	Super propriété de police, Combine : font-weight, font-style, font-size, font-variant- font-family.
text-align	:left; :center; :right; :justify; alignement horizontal
vertical-align	:top; middle; :bottom; :baseline; :sub; :super; :text-top; :text-bottom;
line-height	hauteur de ligne, typique est line-height:24pt;
text-indent	Indentation de la première ligne
white-space	Césures
word-wrap	Césures forcées
text-shadow	Ombre de texte

Propriétés de couleur et de fond

Propriété	Description
color	Couleur du texte, color:rgb(<i>rouge, vert, bleu</i>); color:#ffffff; color:blue;
background-color	Couleur du fond
background-image	Image de fond, ex. background-image:url(images/L_intrus.png);
background-attachment	Fond fixe
background-repeat	Répétition du fond
background-position	Position du fond
background	Super propriété, qui regroupe les 4 dernières propriétés ci-dessus.
opacity	Défini l'opacité et donc la transparence, par un nombre entre 0 et 1.

Propriétés des boîtes (et de ligne horizontale <hr>)

Propriété	Description
width	Largeur, typique en pixels "px" ou en relatif "%"
height	Hauteur, typique en pixels "px" ou en relatif "%"
min-width	Largeur minimale
max-width	Largeur maximale
min-height	Hauteur minimale
max-height	Hauteur maximale
margin-top	Marge du haut, position du haut du cadre relativement à ce qui précède.
margin-right	Marge à droite
margin-bottom	Marge du bas
margin-left	Marge à gauche, position de la gauche du cadre.
margin	Super propriété qui combine les 4 précédentes.
padding-top	Marge intérieure en haut, position du texte, dans le cadre, relativement au haut
padding-right	Marge intérieure à droite
padding-bottom	Marge intérieure en bas
padding-left	Marge intérieure à gauche
padding	Super propriété qui combine les 4 précédentes.
border-width	Épaisseur de la bordure
border-color	Couleur de la bordure
border-style	Type de bordure, none; solide; double; dotted; dashed; hidden; groove; ridge; ...
border	Super propriété qui combine les 3 précédentes.
border-radius	Bordure arrondie
box-shadow	Ombre de boîte

Propriétés de positionnement et d'affichage

Propriété	Description
display	none; (non affiché et ne prend pas de place) inline-block; est affiché en un petit bloque, block; inline;
visibility	visible; hidden; (caché, mais prend de la place), collapse
clip	Affichage d'une partie de l'élément, mais pas ce qui sort du "cadre"
overflow	Défini le comportement en cas de dépassement
float	Avec "display:inline-block", permet un affichage au-dessus du reste de la page.
clear	Arrêt d'un flottant
position	Positionnement, static; relative; absolute; fixed;
top	Positionnement par rapport au haut
bottom	Positionnement par rapport au bas
left	Positionnement par rapport à la gauche
right	Positionnement par rapport à la droite
z-index	Ordre d'affichage en cas de superposition, très utile avec le javascript et déplacement d'images.

Utilisé avec "hover", permet de faire apparaître un texte, c.f. http://www.juggling.ch/gisin/bgweb/a2css1090_liste_exemples_styles.html

Propriétés des listes

Propriété	Description
list-style-type	Type de liste, disc; circle; square; decimal; ...
list-style-position	Position en retrait, en px; ou en %
list-style-image	Puce personnalisée par une image.
list-style-	Super propriété, qui combine les 3 précédentes

Propriétés des tableaux `<table style="..."> ... </table>`

Propriété	Description
border-collapse	Fusion des bordures de tableaux, très souvent utilisé : <code><table style="border-collapse:collapse;"></code>
empty-cells	Affiche les cellules vides
caption-side	Position du titre du tableau

Propriétés du curseur de la souris

Propriété	Description
cursor	Définit l'apparence du curseur de la souris, c.f. la fin de : http://www.juggling.ch/gisin/bgweb/a2css1010_liste_exemples_style.html

Quelques balises de structuration du texte

`` est une balise générique, de type inline, qui suit donc le cours de la ligne.

`<div>` est l'autre balise générique, de type block, qui passe à la ligne suivante.

Ces deux balises sont très souvent utilisées pour définir des styles particuliers de portions de pages Web.

`<sup>` Exposant `<sub>` Indice `` Mise en valeur forte `` Mise en valeur normale

`<h1>` à `<h6>`, divers niveaux de titre

`` pour insérer une image

`<audio ...>` pour insérer un son, avec contrôle du son.

`<video ...>` pour insérer une vidéo, avec contrôle de la vidéo.

`<source ...>` indique la source pour les formats son et audio.

`` pour les liens hypertextes

`
` pour les sauts de lignes

`<p>` pour les paragraphes

`<hr>` pour afficher une ligne horizontale, séparatrice.

`<pre>` pour un texte formaté, les sauts de lignes et espaces sont affichés comment dans la source.

Balises de listes

`` Liste à puce, non numérotée

`` Liste numérotée

`` Élément de la liste à puces

`<dl>` Liste de définitions

`<dt>` Terme à définir

`<dd>` Définition du terme

Balise de tableau

Balise	Description
<table>	Définition du début d'un tableau
<caption>	Titre du tableau, optionnel
<tr>	Ligne de tableau, qui contient plusieurs <td> (cellules) et éventuellement un <th>
<th>	En-tête de cellule, optionnel
<td>	Cellule, se trouvant dans une ligne de tableau
<thead>	Section de l'en-tête du tableau, optionnel
<tbody>	Section du corps du tableau, optionnel
<tfoot>	Section du pied du tableau, optionnel

Balise de formulaire

Balise	Description
<form>	Définition du début d'un formulaire
<fieldset>	Groupe de champs
<legend>	Titre d'un groupe de champs
<label>	Libellé d'un champ
<input>	Champ de formulaire, <input name="nom" value="777" size="5" type="text"> ou <input name="nom" value=" en accord !" type="button">
<textarea>	Zone de saisie multiligne
<option>	Élément d'une liste déroulante
<optgroup>	Groupe d'éléments d'une liste déroulante

class="nom" indique le nom de la classe CSS à utiliser. Elle est définie par *.nom*
 id="nom" indique le nom de l'id CSS à utiliser. Il est **unique**. Il est défini par *#nom*

Pour plus d'information sur les formulaires, voir les pages :
http://www.juggling.ch/gisin/bgweb/a3js1040_formulaires.html
http://www.juggling.ch/gisin/bgweb/a3js1050_formulaires_avec_code.html

Pour plus d'information sur Javascript, voir la page :
<http://www.juggling.ch/gisin/bgweb/exemples.html>

Le lien ci-dessus donne des informations sur la gestion de son avec la balise <audio>.

Pour plus d'information sur la gestion audio, voir la page :
<http://www.juggling.ch/gisin/javascript/Videos/videos.html>

Initiation à Javascript

Javascript est un langage de programmation qui prend de plus en plus d'importance dans le monde d'Internet.

Le but de ce cours est de vous **initier aux bases de l'algorithmique et des langages de programmation**. Nous utiliserons le langage Javascript, car il est simple d'approche, permet d'obtenir rapidement des résultats intéressants. L'approche par **exemples et expérimentation** sera privilégiée par rapport à la théorie.

Tout langage de programmation contient :

- des variables, pour stocker de l'information ;
- des opérateurs (+ - x / = && || ==) pour manipuler le contenu des variables ;
- des tests (if) pour exécuter des instructions seulement sous certaines conditions ;
- des boucles (for while do) pour exécuter plusieurs fois certaines instructions ;
- des fonctions, pour clarifier et simplifier le code ;
- des "inputs" pour accepter des entrées d'un utilisateur, d'un micro, d'une caméra, etc. ;
- des "outputs" pour afficher des résultats ou agir sur l'écran, l'imprimante, des haut-parleurs, etc. ;
- des gestions d'événements, pour réagir à certains événements, tels que des clics de la souris ;
- des bibliothèques de fonctions et d'objets, pour ne pas devoir réinventer la roue.

L'essentielle de ce cours se fera dans des séries d'exercices sur le site :

http://www.juggling.ch/gisin/coursinfo_ref/coursinfo4oc/info4oc.html

et des exemples sur le site :

http://www.juggling.ch/zgin/a2019_oc4/index_oc4_2019.html

Pour un **micro résumé de Javascript**...

c.f. http://www.juggling.ch/gisin/bgweb/adocs/Javascript_syntaxe_resume.pdf

et : http://www.juggling.ch/gisin/bgweb/adocs/Javascript_syntaxe_resume.odt