Betton Yaël
Sacristan Thurian
Bastide Rémi

Compte rendu SAE S1.03

Sommaire :

1. Voici le guide d'installation de la VM:..... 2
A. Installation de Oracle VirtualBox..... 2
B. Installation graphique de la machine virtuelle..... 4
Scripts :..... 11
Présence dans tous les scripts :..... 11
Comptemps.bash :..... 11
1. Lecture et construction de la date..... 11
2. Validation de la date..... 12
3. Utilisation dans les comparaisons..... 12
Demande.bash :..... 12

1. Voici le guide d'installation de la VM:

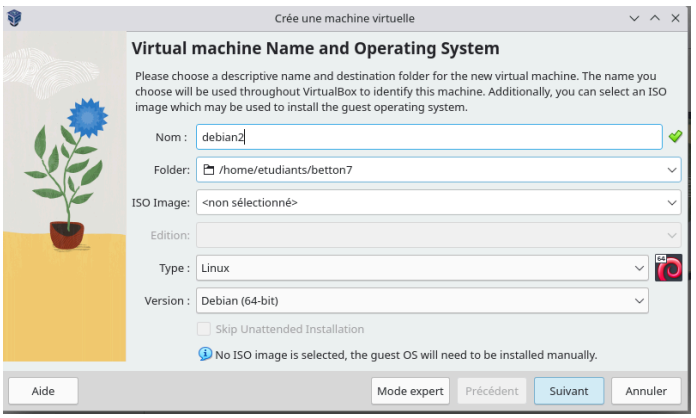
A. Installation de Oracle VirtualBox

Tout d'abord nous allons télécharger l'image iso du fichier Debian (en 64 bit) sur cette adresse là : https://www.debian.org/distrib/index.fr.html

Après l'avoir télécharger, nous ouvrons l'application **Oracle VirtualBox**:

Ensuite cliquons sur le bouton "nouvelle" représenté avec un logo bleu :

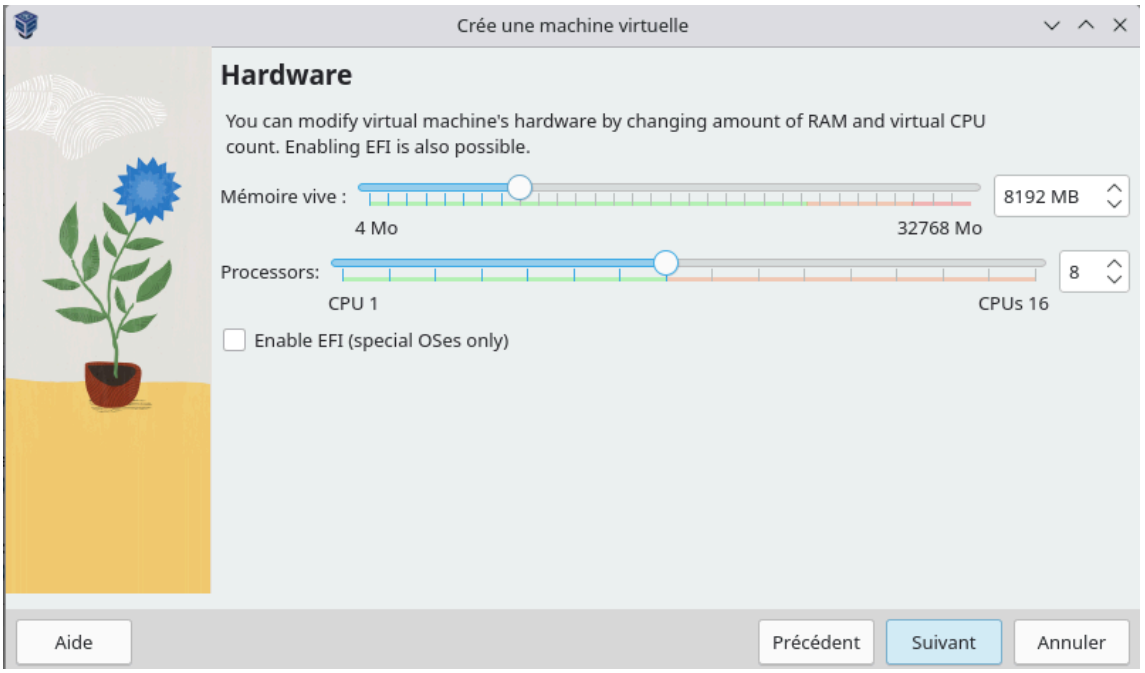
Nous arrivons donc sur cette interface là :



Ensuite, nous allons nommer cette machine virtuelle "Debian" et vérifions que le type est bien "Linux" et que la version, qui est normalement détecté automatiquement, est bien "Debian (64-bit)" :

Par la suite, après avoir cliquer sur suivant une fenêtre suivante s'ouvre en demandant la taille de la mémoire vive et le nombre de processeur que l'on souhaite allouer à la machine virtuelle.

Nous avons choisi d'allouer 8 Go de RAM (mémoire vive) et 8 Processeurs :



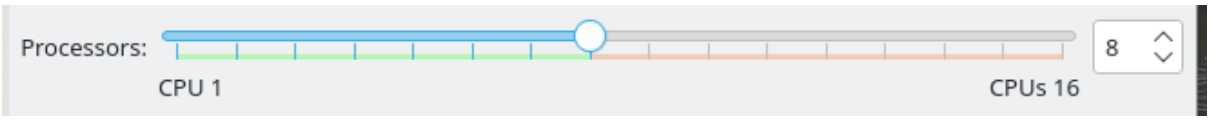
Pourquoi avons-nous choisi d'allouer ces spécifications ?

En suivant la vidéo suivante :

COMMENT INSTALLER DEBIAN 10 SUR VIRTUALBOX - [TUTO]

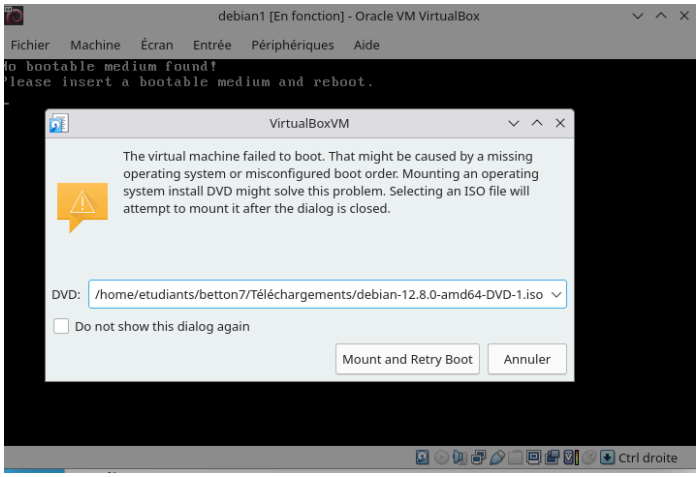
Le youtubeur nous conseille d'allouer ¼ de la mémoire vive de l'ordinateur qui possède la machine virtuelle, l'ordinateur contenait 32 Go de mémoire vive donc ¼ de 32 est égale à 8 :

Pour les processeurs il nous conseille d'allouer autant de cœur de processeur que nous pouvons, tant que nous ne dépassons pas la partie rouge, et que nous ne sommes pas collé à cette partie. Nous avons attribué 8 coeurs (cette image n'a pas été prise sur la vraie machine) :



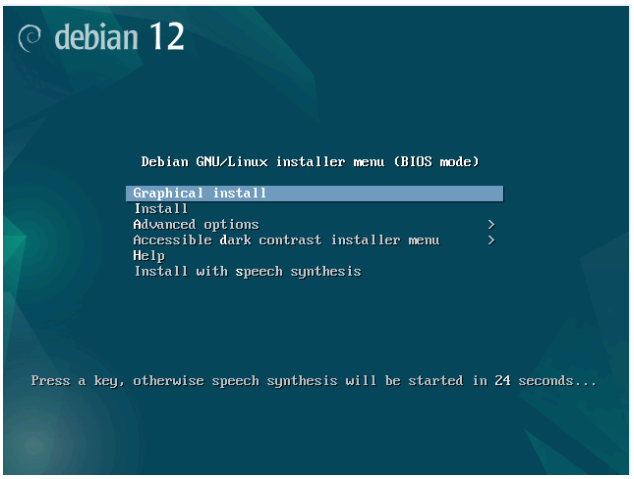
La fenêtre suivante nous demandes combien de place nous voulons allouer au disque dur virtuel, et nous avons choisi 16 Go .

Ensuite lançons la machine virtuel en la sélectionnant et cliquons sur démarrer, une fenêtre s'ouvre en nous demandons quel disque de démarrage faut-il utiliser, sélectionnez votre fichier de debian que vous avez télécharger : debian-12.2.0-amd64-netinst.iso

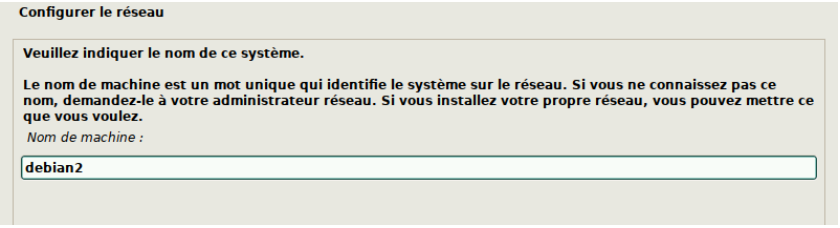


B. Installation graphique de la machine virtuelle

Après avoir lancé la machine virtuelle, vous devez avoir l'image suivante :

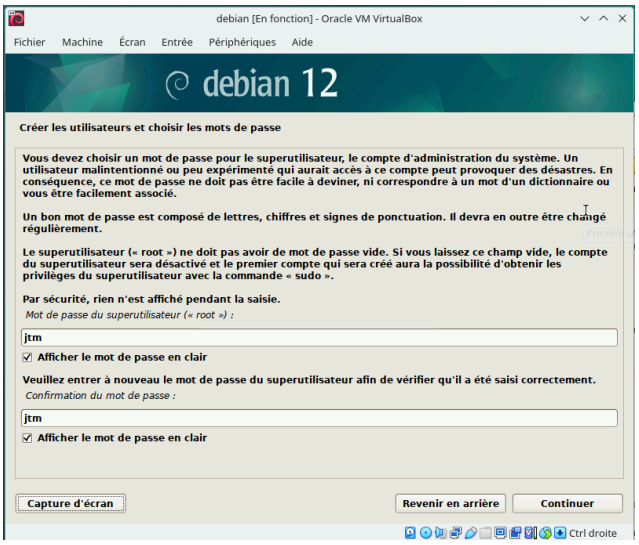


Sélectionnez avec les flèches directionnel "graphical install" puis tapez "entrez"
Sélectionnez le langage avec les flèche et choisissez "Français", puis choisissez "France", puis sélectionner votre disposition clavier soit, pour nous, "Français".
On nous demande ensuite le nom de n'autre machine :



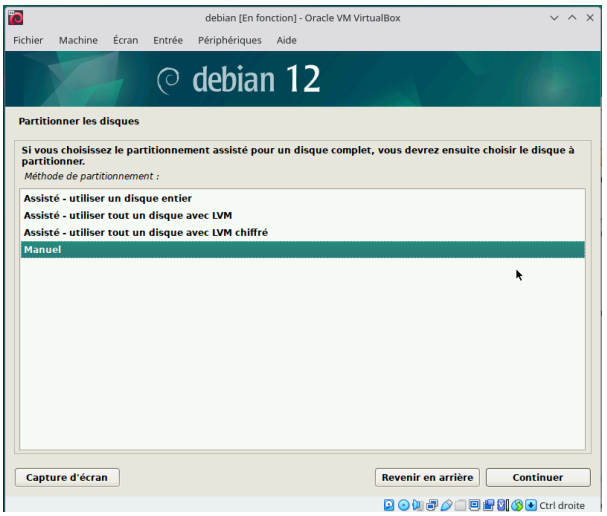
On laisse vide le nom de domaine puisque nous n'en avons pas.

On nous demande le mot de passe super-utilisateur on as mis "jtm"

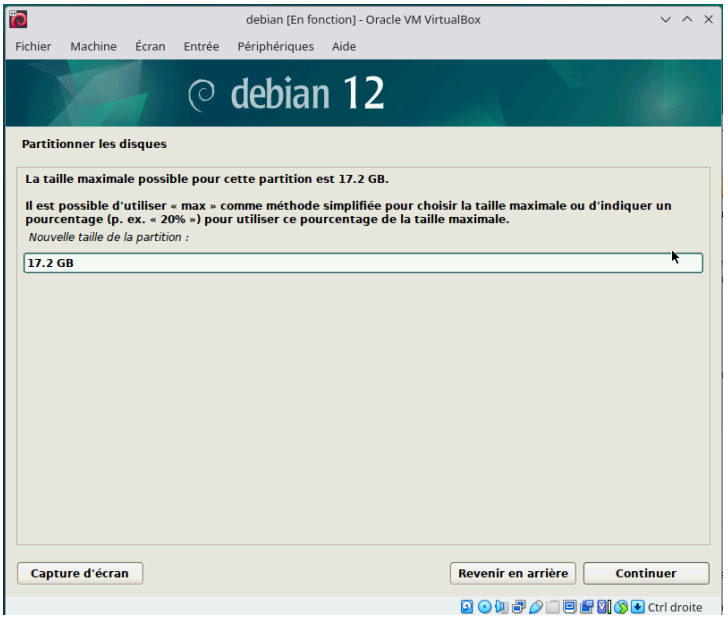


Puis on nous demande le nom d'utilisateur par défaut, on mettra : "y_r_t", puis on nous demande un login on laissera le même que précédemment et son mot de passe qui est demandé à la page d'après sera "jtm"

Passons à la partie partitionner les disques, nous sélectionnons une partition manuelle.

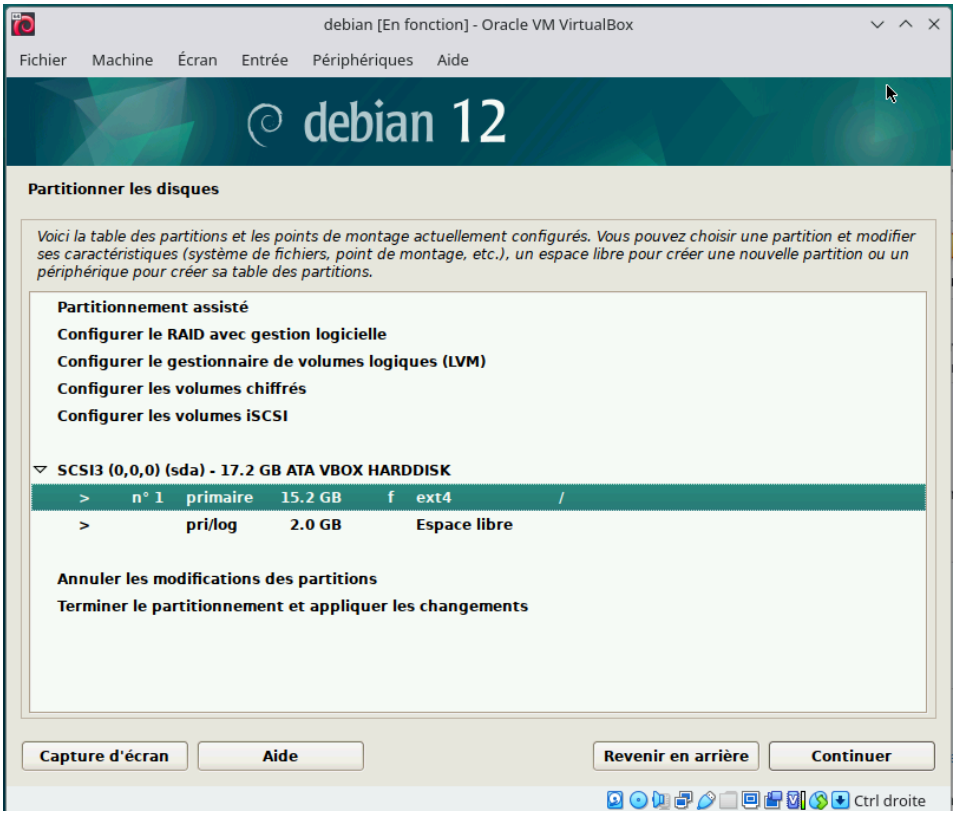


Puis sélectionnons le disque alloué :
Sélectionnez “oui” pour “Faut-il créer une nouvelle table de partitions sur ce disque ?”
Sélectionnez ensuite l’espace libre, créez une nouvelle partition. Et ensuite, on nous demande la taille maximale et nous mettrons le nombre de notre volume - 2 Go.

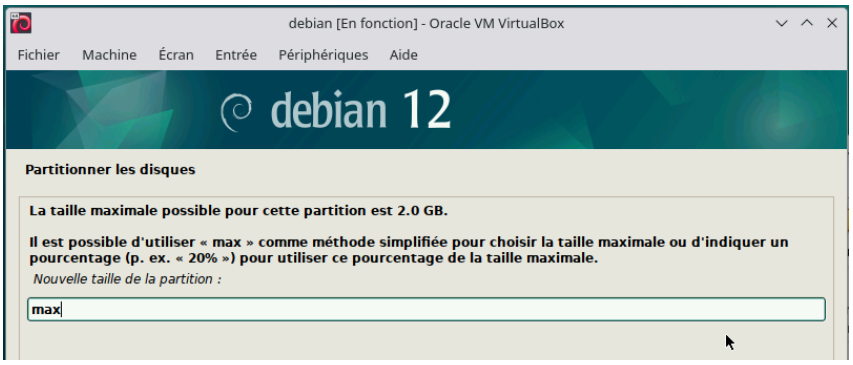


Sélectionnez primaire et début, puis vérifier bien que la partition est utilisée comme “système de fichiers journalisés ext 4” et que le point de montage est bien “/”

Si tout est bon, faites fin du paramétrage de cette partition, puis sélectionnez l’espace de 2 Go qu’il reste.

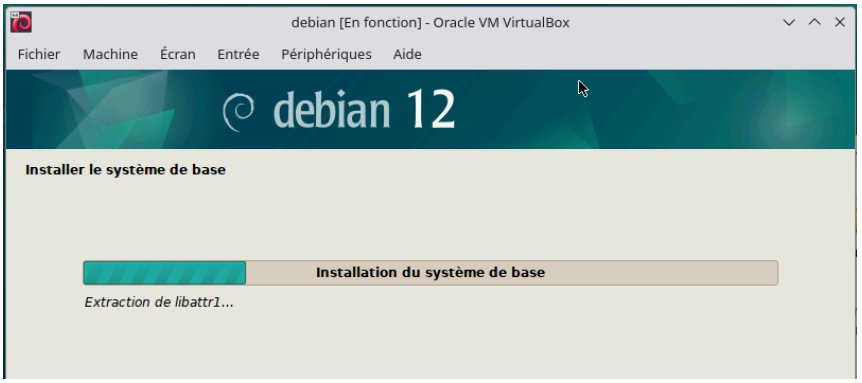


Créez ensuite une nouvelle partition et tapez “max” pour bien prendre toute la taille restante puis sélectionnez logique.



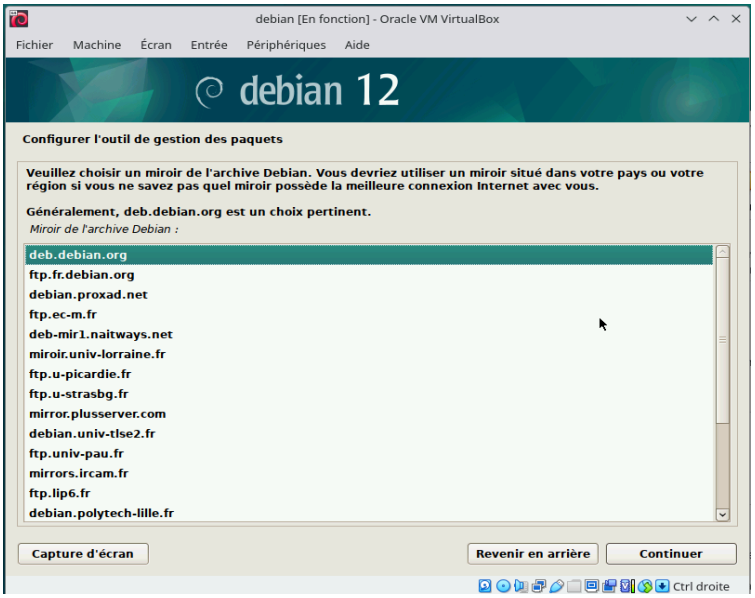
Vérifiez ensuite que la partition est utilisé comme “espace d’échange (“swap”)”
Si tout est bon, alors cliquez sur fin du paramétrage de cette partition. Puis sélectionnez sur fin sur “Terminer le partitionnement et appliquer les changements”, puis sélectionnez “oui”.

Attendez la fin de l’installation :

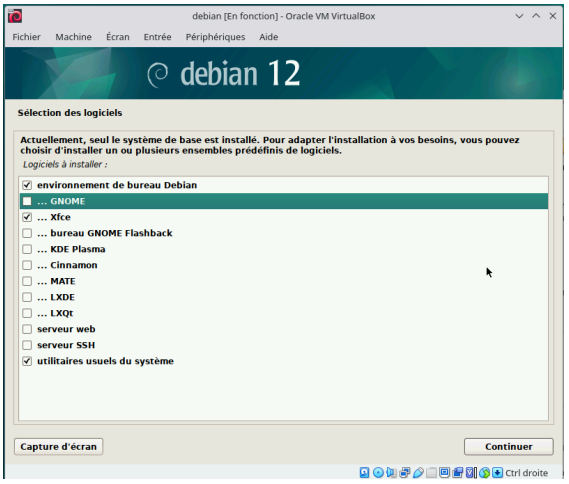


Sélectionnez “non” pour ajouter un autre disque CD ou DVD.

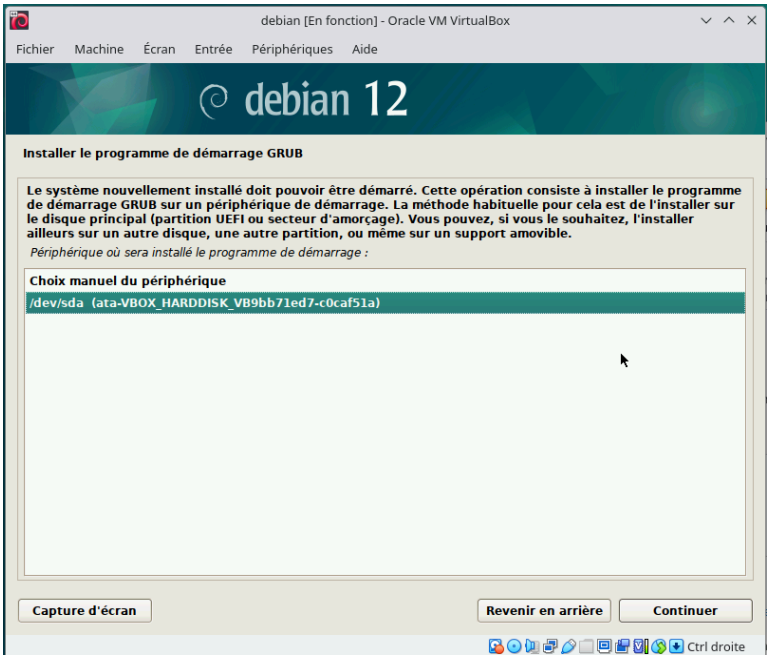
Sélectionnez “France” puis sélectionnez “deb.debian.org”, on ne souhaitera pas utiliser de proxy donc cliquez simplement sur continuer.
On laissera le choix d’envoyer les statistiques ou non.



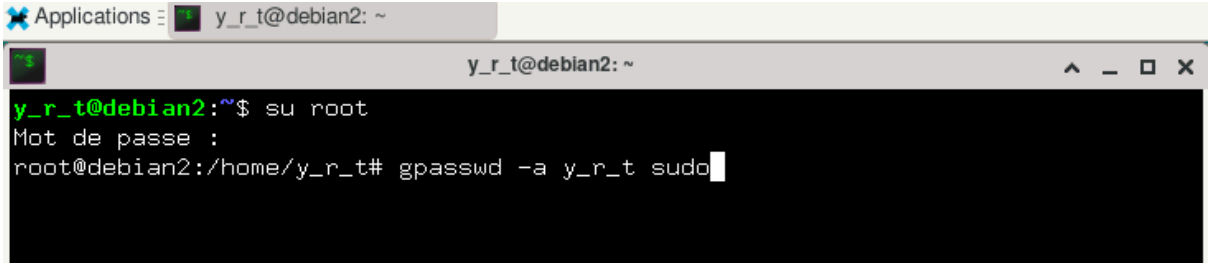
Nous utiliserons “xfce” comme interface Debian car il est peu gourmand en ressource.



On sélectionnez “oui” pour utiliser “Grub”, puis on sélectionne le seul périphérique disponible :

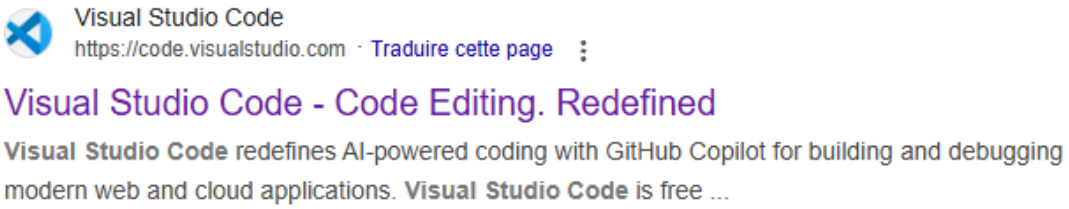


Puis lançons Debian GNU/Linux avec les flèches directionnelles et la touche “entrée”.
Debian se lance donc et on rentre le nom d’utilisateur et le mot de passe définit plutôt. On apparaît sur l’interface de Debian mais faisons certaines manipulations pour pouvoir afficher Debian en plein écran :
Aller dans “application” puis “éditeur de terminal”, tapez la commande “su root” et rentrer le mot de passe du superutilisateur, vous vous connecterez en tant que root et vous taperez ensuite la commande suivante : “gpasswd -a y_r_t sudo” puis faites entrée et fermez la fenêtre et déconnectez vous de la session pour vous reconnectez et que les informations soit bien prise en compte.



Après vous être reconnectez, ouvrez un terminal puis tapez la commande suivante : “sudo apt install make gcc dkms linux-source linux-headers-$(uname -r)”, on nous demandera de rentrer le mot de passe du super utilisateur. Cette commande nous permet de mettre en plein écran la machine virtuelle et d’implémenter le copié collé.

Après avoir exécuté ces codes, lancez le navigateur internet nous allons installer visual studio code, tapez “visual studio code” dans la barre de recherche :



Une page doit apparaître avec un rectangle bleu où il y a marqué “Download”, Cliquez dessus.

Un exécutable va se télécharger, ouvrez le terminal, et exécuter ces commandes :
Il faut se déplacer dans le répertoire téléchargement donc :

```
cd Downloads/
```

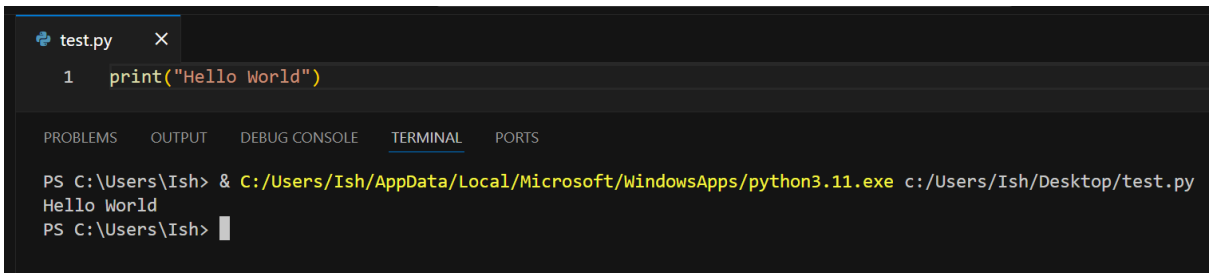
Ensuite il faut installer visual studio code :
sudo dpkg -i “nom du fichier télécharger”

On vous demandera le mot de passe super utilisateur entrez le et attendez la fin du téléchargement.

Vous pouvez fermer le terminal.

Pour terminer, vous avez simplement à rechercher l’éditeur visual studio code dans la barre de recherche.

Lancez visual studio code, nous allons essayer si les extensions python et HTML fonctionne, pour cela recopier le même code qu’il ya sur les captures d’écrans :



```html
<!DOCTYPE html>
<head>
    <h1>Bonjour poulet</h1>
</head>
<p>
    burger
</p>
</html>
```

Bonjour poulet

burger

Si le développement en python ne fonctionne pas, installez l'extension Python :

L'installation est donc terminée.

Scripts :

Présence dans tous les scripts :

- Erreur sur le fichier texte :

```bash
# Vérifie si le fichier se termine par un retour à la ligne sinon ça crée une erreur
if [ "$(tail -c 1 "$file")" != "" ];
then
    # Ajoute un retour à la ligne à la fin du fichier
    echo >> "$file"
fi
```

```
1;quatre_mousquetaires;dumas
2;robinson_crusoe;defoe
3;de_la_terre_a_la_lune;vernes
4;lequipe;anonyme
```

```
$ ./demande.bash dupont lequipe
Le livre n'est pas dans la bibliotheque
```

Ce test est présent dans tous les scripts, car sans un retour à la ligne, le programme ne pourra pas lire la dernière ligne du fichier texte.

Cette commande utilise "tail" pour lire le dernier caractère du fichier (-c 1 signifie "lire 1 caractère à partir de la fin"). Ensuite, le programme vérifie si ce caractère correspond à une chaîne vide, si c'est le cas alors on rajoute une ligne au fichier texte.

Comptemps.bash :

1. Lecture et construction de la date

```
# Conversion de la date en timestamp
date_arg=$(date -d "$annee-$mois-$jour $heure" +%s 2>/dev/null)
```

- date : La commande date permet de manipuler et de convertir des dates.
- -d : Cette option indique que l'argument suivant est une date à interpréter.
- "$annee-$mois-$jour $heure" : C'est la date construite à partir des arguments passés au script. Elle est au format "YYYY-MM-DD HH:MM:SS". (Exemple : si les arguments sont 08 01 2025 15:30:00, la date sera interprétée comme "2025-01-08 15:30:00".)
- +%s : Cette option de date convertit la date en timestamp UNIX (le nombre de secondes écoulées depuis le 1er janvier 1970 à minuit UTC)..
- 2>/dev/null : Cela redirige les messages d'erreur éventuels (par exemple, si la date est invalide) vers /dev/null, les rendant invisibles.

2. Validation de la date

```
if [ -z "$date_arg" ];
then
    echo "Date invalide"
    exit 4
fi
```

- -z : Cette condition vérifie si la variable est vide.
- Si date_arg est vide, cela signifie que la date fournie est invalide (par exemple, une date incorrecte comme "2025-13-08 15:30:00".
- Dans ce cas, le script affiche "Date invalide" et termine avec un code de sortie 4.

3. Utilisation dans les comparaisons

```
if [ "$date_arg" -ge "$date_act" ];
```

- La comparaison détermine si la date fournie est :
    - Postérieure ou égale à la date actuelle (-ge).
    - Antérieure à la date actuelle

Demande.bash :

```
# Vérifie si le numéro d'exemplaire est le bon et si il est bon alors il regarde s'il est disponible
if [[ "$num_livre" == "$id_livre_exemplaire" ]] && [[ "$disponible" == "oui" ]];
then
    echo "Le livre a été emprunté avec succès"

    date_formate=$(date | tr -s ' ' ':')
    heure=$(echo $date_formate | cut -d':' -f4-6)
    jour=$(echo $date_formate | cut -d':' -f3)
    mois=$(date +"%m")
    annee=$(echo $date_formate | cut -d':' -f7)

    echo "$num_membre;$num_livre;$num_exemplaire;$jour $mois $annee $heure" >> emprunts.txt
    # Change le oui par non dans le fichier exemplaires.txt
    sed -i "/^$num_livre;$num_exemplaire;/s/oui/non/" exemplaires.txt
    exit 0
fi
```

Ce passage permet de vérifier si l'ID du livre dans “livres.txt” correspond à l’ID du livre dans “exemplaires.txt” et on vérifie aussi si la disponibilité est “oui”. Si ces vérifications sont passées alors on prend la date actuelle et on la coupe en 4 parties (heure,jour,mois,année).

```
mois=$(date +"%m")
```

La commande “%m” permet de mettre le mois en chiffre et pas en lettre (affichage de base : Jan / affichage après le changement : 01). Ensuite on enregistre les informations dans le fichier “emprunts.txt” du livre empruntés, l’Id du membre, l’Id de l’exemplaire, et pour finir la date. Pour terminer, on remplace oui par non avec la commande “sed”.

Groupe.bash :

```
echo "Création des groupes pour les étudiants..."
# Création des groupes
groupadd A1
groupadd A2
groupadd A3
echo "Groupes créés : A1, A2, A3"

echo "Création des utilisateurs et affectation aux groupes..."
# Création des utilisateurs pour chaque groupe
useradd -m -G A1 etudiant1 -s /bin/bash
echo "etudiant1:password1" | chpasswd
useradd -m -G A2 etudiant2 -s /bin/bash
echo "etudiant2:password2" | chpasswd
useradd -m -G A3 etudiant3 -s /bin/bash
echo "etudiant3:password3" | chpasswd

echo "Création d'un compte 'test'..."
# Création du compte "test"
useradd -m test -s /bin/bash
echo "test:testpassword" | chpasswd

echo "Création des répertoires pour chaque groupe..."
# Création des répertoires et attribution des droits
mkdir -p /home/A1
chown :A1 /home/A1
chmod 740 /home/A1

mkdir -p /home/A2
chown :A2 /home/A2
chmod 740 /home/A2

mkdir -p /home/A3
chown :A3 /home/A3
chmod 740 /home/A3
```

On initialise les groupes grâce à la commande “groupadd”, ensuite nous créons 1 utilisateur pour chaque groupe.Ici -m permet de lui créer directement un répertoire personnel et -G lui permet de lui affecter le groupe.
Pour finir nous créons les répertoires de chaque groupe en les affectant à ceux-ci et leurs affectons les droits 740. Nous donnons les droit read,write et exécuté pour l’utilisateur, et seulement read et exécuté pour les groupes afin que les utilisateurs du même groupe puissent voir les fichiers sans pouvoir les modifier pour autant. Enfin pour les others nous ne donnons aucun droits.