

Exercice 1 : Jouons avec les encodages

- Q1. Créez le dossier `~/R1.03/TP03` et placez-vous dedans.
- Q2. Dans un terminal, entrez la commande `export LANG=C.UTF-8`, puis saisissez, avec `emacs` et dans un fichier nommé `texte.utf8`, la ligne suivante «ABCé€».
- Q3. Déterminez l'option de `wc` permettant de compter les octets d'un fichier.
- Q4. Déterminez l'option de `wc` permettant de compter les caractères d'un fichier.
- Q5. Déterminez la taille du fichier `texte.utf8` en octets puis en caractères. Sont-elles identiques ?
- Q6. Utilisez la commande `od` pour afficher le contenu binaire de ce fichier (sous la forme qui vous est la plus agréable) et déterminer les octets codant le caractère «é».
- Q7. Déduisez le point de code de la table UCS du caractère «é». Le slide 9 du Cours 2 peut vous aider.
- Q8. De la même manière, déduisez le point de code du caractère «€».

Exercice 2 : UTF-8 vs ISO-8859-15

- Q1. Copiez le fichier `/home/public/baste/R1.03/TP03/lettres-utf-8` dans votre dossier de travail.
- Q2. Regardez le contenu du fichier `lettres-utf-8` avec la commande `cat`.
- Q3. En étudiant le contenu du fichier `lettres-utf-8` avec `hexedit` ou `od` déterminez les codes UTF-8 de chacune des lettres présentes dans le fichier en spécifiant le nombre d'octets utilisés par chacun des codes.
- Q4. Unicode affecte le numéro `0x0041` au caractère **A**, et le numéro `0x00E9` au caractère **é**. Vérifiez que le codage que vous avez calculé dans l'exercice précédent est bien correct ?
- Q5. Ouvrez ce fichier avec `emacs`.
- Q6. Le contenu semble-t-il correctement affiché ?
- Q7. En utilisant la commande «`iconv -f utf-8 -t iso-8859-15 lettres-utf-8 -o lettres-latin15`», convertissez `lettres-utf-8` en un fichier équivalent, nommé `lettres-latin15`, codé en ISO-8859-15.
- Q8. Affichez le contenu de ce fichier dans votre terminal grâce à la commande : `cat lettres-latin15`
- Q9. L'affichage semble-t-il correct ?
- Q10. Via les menus du terminal, modifiez le codage de votre terminal pour passer en ISO-8859-15.
- Q11. Affichez de nouveau le contenu du fichier `lettres-latin15`.
- Q12. L'affichage semble-t-il désormais correct ?
- Q13. Affichez le contenu du fichier `lettres-utf-8` dans votre terminal grâce à la commande : `cat lettres-utf-8`
- Q14. L'affichage semble-t-il correct ?
- Q15. Modifiez le codage de votre terminal pour repasser en UTF-8.
- Q16. En étudiant le contenu du fichier `lettres-latin-15` avec `hexedit` ou `od` déterminez les codes ISO-8859-15 de chacune des lettres présentes dans le fichier en spécifiant le nombre d'octets utilisés par chacun des codes.
- Q17. Comparez les tailles des 2 fichiers grâce à la commande `stat`. Lequel est le plus gros ? Pourquoi ?
- Q18. **Juste en observant cette phrase (et sans utiliser votre ordinateur)**, déterminez combien il faut d'octets pour coder le texte du premier article de la déclaration universelle des droits de l'homme en UTF-8 :
Tous les êtres humains naissent libres et égaux en dignité et en droits.
- Q19. En remplaçant `NN` par le nombre déterminé dans la question précédente, exécutez la commande suivante :

```
dd if=/dev/zero of=declaration bs=1 count=NN
```
- Q20. En utilisant `hexedit` et la table de codage UTF-8 déterminée plus tôt, saisissez le texte du premier article de la déclaration universelle des droits de l'homme, dans le fichier `declaration`.
- Q21. Quelle commande vous permet de vérifier que votre saisie est correcte ?

Exercice 3 : Couverture des jeux de caractères

Q1. Construisez le fichier `extrait-utf8` contenant le texte suivant :

Chiffre "0"
Minuscule "a"
Capitale "A"
eacute "é"
Copyright "©"
euro "€"
Yen "¥"
pound "£"
pi "π"

Q2. Exécutez les lignes de commandes suivantes et interprétez le résultat :

```
iconv extrait-utf8 -f UTF-8 -t ASCII -o tmp -c  
iconv tmp -f ASCII -t UTF-8 -o extrait-utf-8
```

Q3. Complétez le tableau suivant en indiquant si chaque caractère est encodable pour les encodages respectifs.

Caractère	UTF-8	ASCII	ISO_8859-15	WINDOWS-1252
A				
é				
©				
€				
¥				
£				
π				