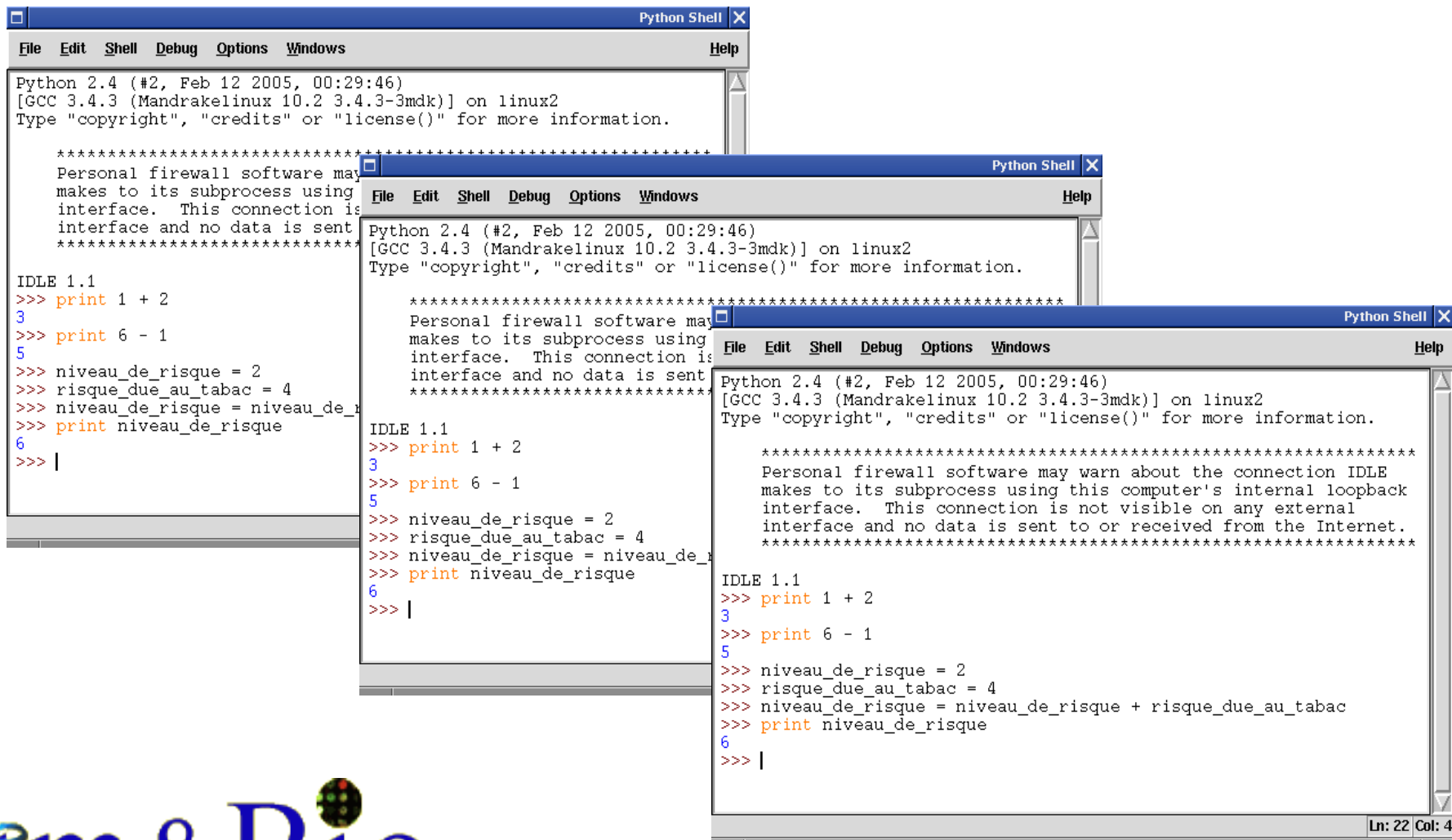


# Cours 3 : Python, les conditions



The image displays three overlapping screenshots of a Python Shell and IDLE 1.1. The top-left screenshot shows the Python Shell window with the following text:

```
Python 2.4 (#2, Feb 12 2005, 00:29:46)
[GCC 3.4.3 (Mandrakelinux 10.2 3.4.3-3mdk)] on linux2
Type "copyright", "credits" or "license()" for more information.

*****
Personal firewall software may warn about the connection IDLE
makes to its subprocess using this computer's internal loopback
interface. This connection is not visible on any external
interface and no data is sent to or received from the Internet.
*****
```

The middle-left screenshot shows the IDLE 1.1 window with the following code:

```
IDLE 1.1
>>> print 1 + 2
3
>>> print 6 - 1
5
>>> niveau_de_risque = 2
>>> risque_due_au_tabac = 4
>>> niveau_de_risque = niveau_de_risque + risque_due_au_tabac
>>> print niveau_de_risque
6
>>> |
```

The middle-right screenshot shows the Python Shell window with the following text:

```
Python 2.4 (#2, Feb 12 2005, 00:29:46)
[GCC 3.4.3 (Mandrakelinux 10.2 3.4.3-3mdk)] on linux2
Type "copyright", "credits" or "license()" for more information.

*****
Personal firewall software may warn about the connection IDLE
makes to its subprocess using this computer's internal loopback
interface. This connection is not visible on any external
interface and no data is sent to or received from the Internet.
*****
```

The bottom-right screenshot shows the IDLE 1.1 window with the following code:

```
IDLE 1.1
>>> print 1 + 2
3
>>> print 6 - 1
5
>>> niveau_de_risque = 2
>>> risque_due_au_tabac = 4
>>> niveau_de_risque = niveau_de_risque + risque_due_au_tabac
>>> print niveau_de_risque
6
>>> |
```

Ln: 22 Col: 4

# Conditions

- Permet d'exécuter des ordres dans certaines conditions :
  - if condition1:**
    - code exécuté si condition1 est vraie*
    - code exécuté si condition1 est vraie...*
  - elif condition2:**
    - code exécuté si condition1 est fausse et condition2 est vraie*
  - else:**
    - code exécuté si condition1 et condition2 sont fausses*
- Attention à l'indentation !!!

# Conditions

- Exemple :

```
fumeur = raw_input("patient fumeur (oui ou non) ? ")
if fumeur == "oui":
    niveau_de_risque = 3
else:
    niveau_de_risque = 0
print niveau_de_risque
```

# Conditions

- Exemples : égal à, différent de, supérieur, incluse dans une liste

```
if niveau_de_risque == 0:  
    print "Le risque est nul !"
```

```
if niveau_de_risque != 0:  
    print "Il y a un risque !"
```

```
if niveau_de_risque >= 3:  
    print "Risque élevé !"
```

# Conditions

- Exercice :
  - Demander l'âge de la personne
  - Si l'âge est supérieur à 60, afficher «c'est une personne âgée»
  - Si l'âge est inférieur à 15, afficher «c'est un enfant»
  - Sinon, afficher «c'est un adulte»

# Conditions

- Exercice :
  - Demander l'âge de la personne
  - Si l'âge est supérieur à 60, afficher «c'est une personne âgée»
  - Si l'âge est inférieur à 15, afficher «c'est un enfant»
  - Sinon, afficher «c'est un adulte»

```
age = int(raw_input("âge de la personne ?"))
```

```
if age > 60:  
    print "c'est une personne âgée"  
elif age < 15:  
    print "c'est un enfant"  
else:  
    print "c'est un adulte"
```

# Conditions

- Exercice :
  - Demander l'âge de la personne et afficher «la personne a X an(s)», où X est la valeur entrée, et en ne mettant un «s» à «ans» que si nécessaire

# Conditions

- Exercice :
  - Demander l'âge de la personne et afficher «la personne a X an(s)», où X est la valeur entrée, et en ne mettant un «s» à «ans» que si nécessaire

```
age = int(raw_input("âge de la personne ?"))
```

```
if age > 1:
```

```
    print "la personne a", age, "ans"
```

```
else:
```

```
    print "la personne a", age, "an"
```

# Conditions

- Opérateur logique ET, OU, NON

```
if (fumeur == "oui") and (age > 60):
```

```
    print "le patient est une personne âgée qui fume !"
```

```
if (fumeur == "oui") or (age > 60):
```

```
    print "le patient est une personne âgée ou un fumeur!"
```

```
if not(fumeur == "oui"):
```

```
    print "le patient est non fumeur!"
```

- Condition imbriquée

```
if fumeur == "oui":
```

```
    if age > 60:
```

```
        print "le patient est une personne âgée qui fume !"
```

# Exercice

- Écrire un programme qui demande à l'utilisateur d'entrer une séquence d'ADN, et qui affiche «C'est un gène» si celle-ci correspond à un gène, et «Ce n'est pas un gène» dans le cas contraire.
- On considérera qu'une séquence est un gène si celle-ci commence par un codon méthionine (ATG) et se termine par un codon STOP (TAA, TAG, TGA)

# Exercice

```
adn = raw_input("entrez une séquence d'ADN: ")
if (adn[:3] == "ATG") and ((adn[-3:] == "TAA") or (adn[-3:] == "TAG")
                           or (adn[-3:] == "TGA")):
    print "C'est un gène !"
else:
    print "Ce n'est pas un gène !"
```

# Exercice

- Nous allons réaliser un programme qui calcule un niveau de risque cardiovasculaire simplifié.
- Le niveau de risque de base est de 0
- Il est augmenté de 2 chez les fumeurs
- Il est diminué de 1 chez les personnes faisant du sport
- Il est augmenté de 1 chez les hommes de plus de 50 ans
- Il est augmenté de 1 chez les femmes de plus de 60 ans
- Le risque cardiovasculaire est élevé si le niveau de risque est supérieur à 1
- Faire un programme qui demande si l'utilisateur fume, quel est son sexe, son âge et s'il fait du sport, qui calcule le niveau de risque et indique si celui-ci est élevé ou non

```
niveau_de_risque = 0

fumeur = raw_input("Êtes-vous fumeur ? (oui ou non) ")
if fumeur == "oui":
    niveau_de_risque = niveau_de_risque + 2

sport = raw_input("Faîtes-vous du sport ? (oui ou non) ")
if sport == "oui":
    niveau_de_risque = niveau_de_risque - 1

sexe = raw_input("Quel est votre sexe ? (h ou f) ")
age = int(raw_input("Quel est votre âge ? "))
if (sexe == "h") and (age > 50):
    niveau_de_risque = niveau_de_risque + 1
if (sexe == "f") and (age > 60):
    niveau_de_risque = niveau_de_risque + 1

if niveau_de_risque > 1:
    print "Le niveau de risque est élevé"
else:
    print "Le niveau de risque est faible"
```

# Exercice

- Jeu : l'utilisateur doit penser à un animal, et le programme doit essayer de deviner de quel animal il s'agit en posant des questions auxquelles l'utilisateur répond par «oui» ou «non»
- Pour simplifier, on se limite à 5 animaux : un oiseau, un poisson, un chien, un tyranosaure, un diplodocus
- (démonstration du programme)

```
existe = raw_input("Est-ce que l'animal existe encore aujourd'hui ?")
if existe == "oui":
    vole = raw_input("Est-ce que l'animal vole ?")
    if vole == "oui":
        print "l'animal est un oiseau !"
    else:
        nage = raw_input("Est-ce que l'animal nage ?")
        if nage == "oui":
            print "l'animal est un poisson !"
        else:
            print "l'animal est un chien !"
else:
    carnivore = raw_input("Est-ce que l'animal est carnivore ?")
    if carnivore == "oui":
        print "l'animal est un tyranosore !"
    else:
        print "l'animal est un diplodocus !"
```

# Exercice

- La clairance rénale peut être calculée par la formule :

Pour l'homme :

$$\text{Clairance (ml/min)} = 1,24 \times (140 - \text{age}) \times \text{poids (kg)} / \text{créatinine } (\mu\text{mol/l})$$

Pour la femme :

$$\text{Clairance (ml/min)} = 1,04 \times (140 - \text{age}) \times \text{poids (kg)} / \text{créatinine } (\mu\text{mol/l})$$

- Écrire un programme qui demande les différents paramètres, calcule la clairance et affiche le résultat
- Quels sont les paramètres ? Quels sont leur type ?

```
sexe = raw_input("Sexe du patient (H ou F) ? ")
age = int(raw_input("Âge du patient ? "))
poids = float(raw_input("Poids du patient (Kg) ? "))
creatinine_plasmatique = float(raw_input("Créatinine plasmatique (µmol/l) ? "))

if sexe == "H":
    clairance = (140 - age) * poids * 1.24 / creatinine_plasmatique
else:
    clairance = (140 - age) * poids * 1.04 / creatinine_plasmatique

print "Clairance de la créatinine :", clairance, "ml/mn"
```

# Exercice

- La clairance rénale peut être calculée par la formule :

Pour l'homme :

$$\text{Clairance (ml/min)} = 1,24 \times (140 - \text{age}) \times \text{poids (kg)} / \text{créatinine } (\mu\text{mol/l})$$

Pour la femme :

$$\text{Clairance (ml/min)} = 1,04 \times (140 - \text{age}) \times \text{poids (kg)} / \text{créatinine } (\mu\text{mol/l})$$

- Limites de validité de la formule :
  - Âge : 18 à 110 ans
  - Poids : 35 à 120 kg
  - Créatinine : 50 à 800  $\mu\text{mol/L}$  (si le patient a une créatinine basse, cela peut conduire à une surestimation de la clairance)
- Écrire un programme qui demande les différents paramètres, calcule la clairance et affiche le résultat, ou une erreur si les limites de validité ne sont pas respectées

```
sexe = raw_input("Sexe du patient (H ou F) ? ")
age = int(raw_input("Âge du patient ? "))
poids = float(raw_input("Poids du patient (Kg) ? "))
creatinine_plasmatique = float(raw_input("Créatinine plasmatique (µmol/l) ? "))

if age < 18 : print "La formule n'est pas valable en dessous de 18 ans !"
elif age > 110 : print "La formule n'est pas valable au dessus de 110 ans !"
elif poids < 35 : print "La formule n'est pas valable en dessous de 35 kg !"
elif poids > 120 : print "La formule n'est pas valable au dessus de 120 kg !"
elif creatinine_plasmatique < 50 : print "La formule n'est pas valable en
                                     dessous de 50 µmol/l !"
elif creatinine_plasmatique > 800 : print "La formule n'est pas valable au
                                     dessus de 800 µmol/l !"
else:
    if sexe == "H":
        clairance = (140 - age) * poids * 1.24 / creatinine_plasmatique
    else:
        clairance = (140 - age) * poids * 1.04 / creatinine_plasmatique
    print "Clairance de la créatinine :", clairance, "ml/mn"
```