

Les éléments à télécharger sont disponibles à l'adresse suivante :
<http://www.editions-eni.fr>
Saisissez la référence ENI de l'ouvrage **RIPYTCN** dans la zone de recherche et validez. Cliquez sur le titre du livre puis sur le bouton de téléchargement.

Avant-propos

Chapitre 1

Nombres, opérations et fonctions dans Python

1. Nombres et opérations	17
1.1 Entiers et décimaux	17
1.2 Les variables numériques	18
1.3 L'opérateur d'affectation =	18
1.4 Les opérations disponibles dans Python	19
1.5 Les expressions numériques	19
1.6 Les opérateurs de comparaison	20
1.7 Le module « fractions »	21
1.8 Deux autres instructions du module « fractions »	22
2. Représentation des nombres	23
2.1 Historique	23
2.2 La représentation binaire des entiers naturels	24
2.3 La représentation binaire des entiers relatifs	25
2.4 Les nombres dyadiques	25
2.5 La représentation des nombres à virgule à l'aide des « flottants »	26
2.6 La précision des calculs	27
3. Fonctions disponibles dans Python	29
3.1 Les fonctions usuelles	29
3.2 Les fonctions numériques du module « math »	30
3.3 Comment définir ses propres fonctions ?	31

3.4	Définir une fonction par une suite d'actions	32
3.5	Le mot réservé « lambda »	33
4.	La récursivité des fonctions	35
4.1	Les factorielles	35
4.2	Une application de la récursivité : les tours de Hanoï	37

Chapitre 2

Suites de nombres réels

1.	Suites et racines carrées	41
1.1	La méthode d'Archytas de Tarente	41
1.2	La méthode de Héron d'Alexandrie	43
1.3	Le calcul d'une racine cubique	45
2.	Comment définir une suite ?	47
2.1	Définition	47
2.2	Suites définies par $u_n = f(n)$	47
2.3	Suites récurrentes	48
3.	Quand n devient de plus en plus grand	51
3.1	Une suite peut être convergente	51
3.2	Une suite peut ne pas avoir de limite	53
4.	Une suite célèbre : la suite de Fibonacci	55
4.1	Historique	55
4.2	Le problème des lapins	56
4.3	L'étude du rapport de deux termes successifs de la suite	58
4.4	L'étude du nombre $ r_n - \varphi $	59
4.5	La formule de Binet	59
5.	Suites définies par des sommes	61
5.1	Historique	61
5.2	La somme des carrés et des cubes des entiers naturels de 1 à n	62
5.3	Les séries géométriques	63
5.4	La série de Swineshead	64
5.5	La série harmonique et la série harmonique alternée	65

5.6 Le problème de Bâle 66
 5.7 Le nombre e 67

Chapitre 3

Fonction exponentielle et fonctions logarithmes

1. La fonction exponentielle 69
 1.1 Historique 69
 1.2 Définition de la fonction exponentielle par Euler 70
 1.3 Dérivée de la fonction $x \rightarrow \exp(x)$ 71
 1.4 Autre définition de $\exp(x)$ 72
 1.5 Instructions $\exp(x)$ et $e^{**}x$ 73
 1.6 Représentation graphique de la fonction exponentielle $x \rightarrow \exp(x)$ 74
 2. Les logarithmes décimaux 75
 2.1 Historique 75
 2.2 Logarithme décimal d'un nombre strictement positif 76
 2.3 Fonction logarithme décimal dans Python 76
 2.4 Représentation graphique de la fonction logarithme décimal. . 77
 3. L'algorithme de Briggs 79
 3.1 Historique 79
 3.2 Un programme Python 80
 4. Les logarithmes népériens 83
 4.1 Historique 83
 4.2 Définition et calcul de $\ln(x)$ pour $x \geq 0$ 85
 4.3 Programme pour calculer un encadrement de $\ln(x)$ 86
 4.4 Fonction logarithme népérien dans Python 87

Chapitre 4

Dérivation numérique et équations différentielles

1. Dérivée d'une fonction numérique	89
1.1 Historique	89
1.2 Dérivées à droite et dérivées à gauche	90
1.3 Calculs approchés de $f'_d(x)$ et de $f'_g(x)$	92
1.4 Calcul approché de $f'(x)$ à l'aide de $f'_g(x_0)$ et de $f'_d(x_0)$	94
2. Calcul approché de $f'(x)$ et de $f''(x)$	95
2.1 Administration par un polynôme	95
2.2 Calcul d'une valeur approchée de $f'(x)$	96
2.3 Application à la fonction exponentielle	97
2.4 Approximation de $f(x)$ au voisinage de x_0	98
2.5 Calcul approché de $f''(x)$	98
3. Qu'est-ce qu'une équation différentielle ?	101
3.1 Historique	101
3.2 Les équations du type $y'=f(x)$	103
3.3 Les équations du type $y'=ay$	103
3.4 Les équations du type $y'=ay+b$	104
3.5 Les équations linéaires du type $ay'+by=z$	105
3.6 La notation différentielle de Leibniz	106
4. La méthode d'Euler	109
4.1 Principe de la méthode d'Euler.	109
4.2 Un programme pour calculer $y(x)$	110
4.3 Influence du choix de n sur la précision des résultats	112
4.4 Construction d'une fonction Euler(x_0, y_0, x, n)	113
4.5 Un cas particulier : l'équation différentielle $y'=y$	114
5. Les méthodes de Runge-Kutta	117
5.1 Historique	117
5.2 Cas d'une équation différentielle du premier ordre	117
5.3 Cas d'une équation différentielle du second ordre	120

Chapitre 5
Résolution approchée des équations

- 1. La recherche d'une solution par dichotomie 123
 - 1.1 Historique 123
 - 1.2 Deux programmes 124
- 2. La méthode des approximations successives 129
 - 2.1 Historique 129
 - 2.2 Étude d'un exemple 130
 - 2.3 Deux programmes pour calculer r 133
- 3. La méthode de Newton 135
 - 3.1 Historique 135
 - 3.2 Extension de la méthode 136
 - 3.3 Représentation graphique de la méthode de Newton 138
 - 3.4 Deux programmes 138

Chapitre 6
Calcul infinitésimal et intégration numérique

- 1. Longueur d'un arc de courbe 141
 - 1.1 Principe du calcul 142
 - 1.2 Un programme de calcul 142
 - 1.3 Application : calcul du nombre π 143
- 2. Aire du disque et calcul de π 145
 - 2.1 Historique 145
 - 2.2 Méthode d'Archimède 145
 - 2.3 Avec le calcul infinitésimal 147
 - 2.4 Calcul de π par la méthode de Monte-Carlo. 148
- 3. Volume d'une boule 151
 - 3.1 Principe du calcul 151
 - 3.2 Le calcul. 151
 - 3.3 Programme de calcul. 153

4.	Intégration approchée par la méthode des rectangles	155
4.1	Historique	155
4.2	Principe de la méthode des rectangles	156
4.3	Programme pour calculer l'intégrale d'une fonction continue .	157
4.4	Cas des fonctions monotones	158
5.	Intégration approchée par la méthode des trapèzes	161
5.1	Rappel	161
5.2	Principe de la méthode des trapèzes	162
5.3	Programme pour calculer l'intégrale d'une fonction continue .	162
6.	Intégration approchée par la méthode de Simpson	165
6.1	Historique	165
6.2	Méthode de Simpson	165
6.3	Cas d'une intégrale avec une borne infinie	166
7.	Intégration approchée par la méthode de Gauss	169
7.1	Historique	169
7.2	Principe de la méthode de Gauss	169
7.3	Un programme de calcul	170
7.4	Deux remarques	171

Chapitre 7 Nombres complexes

1.	Les nombres complexes dans Python	173
1.1	Historique	173
1.2	Construction moderne des nombres imaginaires	175
1.3	Nombre complexe i et ses propriétés	176
1.4	Représentation algébrique d'un nombre complexe	176
1.5	Opérations sur les complexes dans Python	177
1.6	Forme trigonométrique d'un nombre complexe	178
1.7	Forme exponentielle d'un nombre complexe	179

- 2. Résolution dans \mathbb{C} des équations du second degré 181
 - 2.1 Cas d'une équation à coefficients réels 181
 - 2.2 Cas d'une équation du second degré à coefficients complexes 182
- 3. Les suites de nombres complexes 185
 - 3.1 Suites récurrentes 185
 - 3.2 Partie réelle et partie imaginaire d'une suite complexe 185
 - 3.3 Convergence d'une suite. 186
 - 3.4 Une suite géométrique 188
 - 3.5 Représentation graphique d'une suite 189
- 4. Aperçu sur les fonctions d'une variable complexe 193
 - 4.1 Fonctions nouvelles 193
 - 4.2 La fonction $z \rightarrow z+a$ 193
 - 4.3 La fonction $z \rightarrow az$ avec $|a|=1$ 195
 - 4.4 La fonction $z \rightarrow az$ avec a réel 195
 - 4.5 Les fonctions homographiques complexes 196
 - 4.6 Les transformations homographiques du plan complexe 197

Chapitre 8
Éléments de statistiques

- 1. Les paramètres d'une série statistique 199
 - 1.1 Un exemple 199
 - 1.2 Construction du tableau des effectifs 200
 - 1.3 Calcul de la médiane. 201
 - 1.4 Calcul des quartiles Q_1 et Q_3 et
de l'écart interquartile Q_3-Q_1 202
 - 1.5 Calcul de la moyenne 203
 - 1.6 Calcul de la variance et de l'écart-type 203
- 2. Covariance et coefficient de corrélation 205
 - 2.1 Historique 205
 - 2.2 Définitions 206
 - 2.3 Un programme de calcul de $\text{cov}(x,y)$ et de r 206

3.	Ajustements linéaires et autres	209
3.1	Historique	209
3.2	Ajustement linéaire	209
3.3	Ajustement par une exponentielle	211
3.4	Ajustement par une fonction puissance	212

Chapitre 9 Combinatoire et échantillonnage

1.	Factorielles et combinaisons	215
1.1	Premières recherches	215
1.2	L'invention des factorielles	216
1.3	Les combinaisons de n objets pris p à p	217
1.4	Le calcul du nombre $\binom{n}{p}$	218
2.	Échantillonnage	221
2.1	Historique	221
2.2	Fabrication expérimentale d'un échantillon	221
2.3	Un calcul direct	222
3.	Échantillonnage et fréquences	225
3.1	Fluctuations d'échantillonnage	225
3.2	Intervalle de fluctuation de la fréquence d'un échantillon	226
3.3	Estimation de la fréquence d'un caractère dans une population	227
3.4	Quelques remarques	229

Chapitre 10 Les probabilités

1.	Les probabilités conditionnelles	231
1.1	Une simulation pour conjecturer	231
1.2	Le calcul confirme la conjecture	234
1.3	Une formule pour définir une probabilité conditionnelle	234

1.4	Un exemple	235
2.	La formule de Bayes	237
2.1	Historique	237
2.2	La formule de Bayes	237
2.3	Une première simulation	238
2.4	Une deuxième simulation	239
3.	L'espérance et l'écart-type d'une variable aléatoire discrète	243
3.1	Variables aléatoires discrètes	243
3.2	Variables aléatoires et lois de probabilité	243
3.3	Espérance mathématique d'une variable aléatoire	243
3.4	Variance et écart-type d'une variable aléatoire X	244
3.5	Un programme pour calculer $E(X)$, $V(X)$ et (X)	245
4.	La loi binomiale.	247
4.1	Expériences et schémas de Bernoulli	247
4.2	Étude d'un exemple	248
4.3	Une généralisation : la loi binomiale	248
4.4	Un programme pour calculer $P(X=k)$	249
4.5	Un programme pour calculer $P(X \leq k)$	250
4.6	Espérance et écart-type	251
5.	La loi de Poisson	253
5.1	Historique	253
5.2	Expression de la loi de Poisson.	253
5.3	Exemples	254
5.4	Loi de Poisson et loi binomiale.	255
6.	Les variables aléatoires continues	257
6.1	Historique	257
6.2	Qu'est-ce qu'une variable aléatoire continue ?	258
6.3	Comment définir une loi de probabilité continue ?	259
6.4	Espérance et écart-type d'une variable aléatoire continue	260
7.	La loi exponentielle.	261
7.1	À quoi sert cette loi ?	261
7.2	Définition	261

7.3	Espérance et variance d'une loi exponentielle.	262
7.4	Calcul de la probabilité $P(a < X < b)$	262
7.5	Application à la physique.	263
7.6	Usure et vieillissement	264
8.	La loi normale	265
8.1	Définition	265
8.2	Loi normale réduite	266
8.3	Calcul de $P(X < a)$	267
8.4	Calcul inverse	268
8.5	Exemple d'utilisation de la loi normale	269
9.	Loi normale et jugements statistiques	271
9.1	Intervalle de fluctuation d'une moyenne	271
9.2	Intervalle de fluctuation d'une fréquence	272
9.3	Intervalle de confiance d'une moyenne.	274
9.4	Intervalle de confiance d'une fréquence	275

Chapitre 11

Arithmétique et cryptographie

1.	La division euclidienne des entiers	277
1.1	Deux fonctions de Python	277
1.2	La division euclidienne des entiers relatifs	278
2.	Les diviseurs d'un entier naturel	281
2.1	Recherche des diviseurs d'un entier naturel	281
2.2	Somme des diviseurs propres d'un entier	283
2.3	Nombres parfaits	284
2.4	Nombres amicaux.	285
3.	Les nombres premiers.	287
3.1	Les nombres premiers sont en nombre infini	287
3.2	Le crible d'Ératosthène	287
3.3	Comment savoir si un entier donné est premier ?	288
3.4	Des listes de nombres premiers	289

3.5	La conjecture des nombres premiers jumeaux	290
3.6	La conjecture de Goldbach	291
4.	Le PGCD de deux entiers	293
4.1	L'algorithme d'Euclide	293
4.2	La méthode des divisions successives	294
4.3	La fonction pgcd dans Python	295
5.	Les factorisations d'un entier naturel	297
5.1	Décomposition en facteurs premiers	297
5.2	Décomposition en facteurs premiers et recherche d'un PGCD	298
5.3	Une autre méthode de factorisation	299
5.4	Méthode de Fermat	301
6.	Le théorème de Bezout	305
6.1	Historique	305
6.2	Deux exemples	306
6.3	Recherche des coefficients de Bezout avec Python	307
6.4	Conséquence du théorème de Bezout, le théorème de Gauss	309
7.	Introduction aux équations diophantiennes	311
7.1	Historique	311
7.2	Un exemple d'équation diophantienne	312
7.3	Un autre exemple	313
7.4	Un programme pour résoudre l'équation $ax+by=c$	314
7.5	Une équation diophantienne du second degré	315
8.	La congruence des entiers relatifs	317
8.1	Le terme « modulo »	317
8.2	Calcul des restes modulo n	318
8.3	Calculs modulo n et calculs dans l'anneau $\mathbb{Z}/n\mathbb{Z}$	319
8.4	Résolution de l'équation $ax+b=c$ dans l'anneau $\mathbb{Z}/n\mathbb{Z}$	322
9.	Le code secret de Jules César	323
9.1	Historique	323
9.2	Les instructions <code>ord()</code> et <code>chr()</code> de Python	323
9.3	Un programme pour coder un texte	324

9.4	Un programme pour décoder un texte quand on connaît le décalage	325
9.5	Décodage avec une analyse des fréquences des lettres.	325
9.6	Un programme de décodage quand on ne connaît pas le décalage employé	326
10.	Le chiffre de Vigenère	329
10.1	Historique	329
10.2	Principe du chiffre de Vigenère	330
10.3	Un programme de chiffrement et de déchiffrement	331
11.	Les codages affines	335
11.1	Une convention	335
11.2	Un programme pour coder un texte	335
11.3	Comment choisir les entiers a et b ?	336
11.4	Décodage d'un texte codé par une fonction affine avec a et b connus	337
11.5	Décodage d'un texte codé par une fonction affine avec a et b inconnus	338
11.6	Un programme général de décodage	338
12.	Le chiffrement de Hill	341
12.1	Principe du chiffrement	341
12.2	Un exemple	341
12.3	Principe du déchiffrement	342
12.4	Un programme pour coder	343
12.5	Un programme pour décoder	344

Chapitre 12

Matrices 2x2 et matrices 3x3

1.	Matrices carrées et applications linéaires	349
1.1	Historique	349
1.2	Une matrice représente une application linéaire	350
1.3	Représentation d'une matrice avec Python.	352
1.4	Image d'un vecteur par une matrice carrée 2x2 ou 3x3	353

2.	Opérations sur les matrices	355
2.1	Addition, soustraction et multiplication par un réel	355
2.2	Multiplication des matrices carrées de taille 2	356
2.3	Propriétés particulières de la multiplication des matrices	357
2.4	Un programme pour multiplier des matrices 2x2	358
2.5	Un programme pour multiplier des matrices 3x3	359
2.6	Multiplication de deux matrices de tailles différentes	360
2.7	Opérations avec des matrices carrées remarquables	361
3.	Déterminant d'une matrice carrée 2x2 ou 3x3	363
3.1	Déterminant d'une matrice 2x2	363
3.2	Déterminant d'une matrice 3x3	364
3.3	Déterminant d'un système de vecteurs	366
4.	Inversion des matrices carrées 2x2 et 3x3	367
4.1	Qu'est-ce qu'une matrice inversible ?	367
4.2	Inverse d'une matrice carrée 2x2	368
4.3	Inverse d'une matrice carrée 3x3	369
4.4	Méthode du pivot	370
5.	Résolution d'un système linéaire d'équations	373
5.1	Un exemple historique	373
5.2	Écriture matricielle des systèmes linéaires d'équations	374
5.3	Un programme pour résoudre les systèmes de deux équations à deux inconnues	375
5.4	Un programme pour résoudre les systèmes de trois équations à trois inconnues	376
5.5	Remarque sur l'emploi des déterminants	377
6.	Puissances d'une matrice 2x2 ou 3x3	379
6.1	Puissance d'une matrice 2x2	379
6.2	Puissance d'une matrice 3x3	380
6.3	Cas des matrices diagonales	381
7.	Diagonalisation d'une matrice 2x2	383
7.1	Les matrices diagonalisables	383
7.2	Étude d'un exemple	383

7.3	Diagonalisation d'une matrice 2×2	385
7.4	Un programme pour calculer les valeurs propres d'une matrice 2×2	386
8.	Matrices et suites récurrentes	389
8.1	Rappel : les nombres de Fibonacci	389
8.2	Calcul des nombres de Fibonacci à l'aide d'une matrice 2×2 . .	389
8.3	Les relations de Binet	390

Chapitre 13

Géométrie analytique

1.	Équation réduite d'une droite dans le plan	393
1.1	Historique	393
1.2	Détermination de l'équation réduite d'une droite	394
1.3	Intersection de deux droites	397
1.4	Distance d'un point à une droite	398
2.	Équation cartésienne d'une droite dans le plan	401
2.1	Historique	401
2.2	Recherche de l'équation cartésienne d'une droite dont on connaît deux points	401
2.3	Recherche de l'équation cartésienne d'une droite dont on connaît un vecteur directeur et un point	403
2.4	Intersection de deux droites	404
2.5	Droites parallèles	405
2.6	Vecteurs orthogonaux, vecteur normal à une droite	406
2.7	Droites perpendiculaires	407
3.	Droites dans l'espace	409
3.1	Vecteurs colinéaires dans l'espace	409
3.2	Points alignés	409
3.3	Représentation paramétrique d'une droite	410
3.4	Comment reconnaître qu'un point appartient à une droite ? .	411
3.5	Droites coplanaires, intersection de deux droites	412

- 4. Équations paramétriques d'un plan 415
 - 4.1 Historique 415
 - 4.2 Détermination de l'équation paramétrique d'un plan 416
 - 4.3 Comment reconnaître qu'un point appartient à un plan ? . . . 417
 - 4.4 Comment reconnaître que quatre points sont coplanaires ? . 418
 - 4.5 Intersection d'un plan et d'une droite 419
- 5. Équation cartésienne d'un plan 423
 - 5.1 Produit scalaire de 2 vecteurs 423
 - 5.2 Équation d'un plan défini par un de ses points
et par un vecteur normal 423
 - 5.3 Équation d'un plan défini par trois points non alignés 424
 - 5.4 Intersection d'une droite et d'un plan 427
 - 5.5 Distance d'un point à un plan 428
 - 5.6 Intersection de deux plans 430

Annexes

- 1. Bibliographie 433
- 2. Comment utiliser les scripts du livre ? 437

Notes 439

Index 441

Les éléments à télécharger sont disponibles à l'adresse suivante :
<http://www.editions-eni.fr>
Saisissez la référence ENI de l'ouvrage **RI33PYT** dans la zone de recherche et validez. Cliquez sur le titre du livre puis sur le bouton de téléchargement.

Avant-propos

1. Introduction	23
2. Contenu de l'ouvrage	23
3. Progressivité de l'ouvrage	24
4. À destination des enseignants et élèves	25
5. À destination des chercheurs ou doctorants	27
6. À destination de ceux qui viennent d'un autre langage	27

Partie 1 : Les atouts de Python

Chapitre 1.1

Clés théoriques

1. Petite histoire des langages informatiques	29
1.1 Informatique théorique	29
1.2 Chronologie de l'informatique	30
1.2.1 Évolutions des problématiques liées à l'informatique	30
1.2.2 Chronologie des langages informatiques	31
2. Typologie des langages de programmation	35
2.1 Paradigmes	35
2.1.1 Définition	35
2.1.2 Paradigme impératif et dérivés	36
2.1.3 Paradigme objet et dérivés	37
2.1.4 Programmation orientée aspect	37
2.1.5 Paradigme fonctionnel	37
2.1.6 Paradigme logique	38
2.1.7 Programmation concurrente	38
2.1.8 Synthèse	38
2.2 Interopérabilité	39

2.3	Niveau de programmation	41
2.3.1	Machine	41
2.3.2	Bas niveau	41
2.3.3	Haut niveau	42
2.4	Typage	43
2.4.1	Faible vs fort	43
2.4.2	Statique vs dynamique	43
2.5	Grammaire	43
2.5.1	Langages formels	43
2.5.2	Syntaxe	44
3.	Python et le reste du monde	45
3.1	Positionnement stratégique du langage Python	45
3.1.1	Segments de marchés	45
3.1.2	Niveau de complexité	45
3.1.3	Forces du langage	45
3.1.4	Points faibles	46
3.2	Intégration avec d'autres langages	46
3.2.1	Extensions C	46
3.2.2	Intégration de programmes écrits en C	47
3.2.3	Intégration de programmes Python dans du C	47
3.2.4	Intégration de programmes écrits en Java	47
3.2.5	Intégration de programmes Python dans Java	47
3.2.6	Autres intégrations	47

Chapitre 1.2

Présentation de Python

1.	Philosophie	49
1.1	Python en quelques lignes	49
1.1.1	D'où vient le nom « Python » ?	49
1.1.2	Présentation technique	49
1.1.3	Présentation conceptuelle	50
1.2	Comparaison avec d'autres langages	50
1.2.1	Shell	50
1.2.2	Perl	51
1.2.3	C, C++	51
1.2.4	Java	52
1.2.5	PHP	54

1.3	Grands principes	55
1.3.1	Le zen de Python	55
1.3.2	Le développeur n'est pas stupide	55
1.3.3	Documentation	56
1.3.4	Python est livré piles incluses	56
1.3.5	Duck Typing	56
1.3.6	Notion de code pythonique	57
2.	Histoire de Python.	57
2.1	La genèse	57
2.2	Extension du périmètre fonctionnel	58
2.3	Évolution de la licence	62
2.4	Avenir	62
3.	Gouvernance	63
3.1	Développement	63
3.1.1	Branches	63
3.1.2	Communauté.	64
3.2	Mode de gouvernance.	65
3.2.1	Créateur du langage.	65
3.2.2	PEP	65
3.2.3	Prise de décisions	65
3.2.4	Contribuer à Python	66
4.	Que contient Python ?	66
4.1	Une grammaire et une syntaxe	66
4.2	Plusieurs implémentations.	66
4.3	Une bibliothèque standard.	67
4.4	Des bibliothèques tierces	67
4.5	Des frameworks	67
5.	Phases d'exécution d'un programme Python.	67
5.1	Chargement de la machine virtuelle	67
5.2	Compilation	68
5.3	Interprétation	69

Chapitre 1.3

Pourquoi choisir Python

1. Qualités du langage	71
1.1 Ticket d'entrée	71
1.2 Qualités intrinsèques	73
1.3 Couverture fonctionnelle	73
1.4 Domaines d'excellence	74
1.5 Garanties	75
2. Diffusion	76
2.1 Entreprises	76
2.2 Le monde de la recherche	77
2.3 Le monde de l'éducation	78
2.4 Communauté	79
3. Références	80
3.1 Poids lourds de l'industrie informatique	80
3.1.1 Google	80
3.1.2 Mozilla	81
3.1.3 Microsoft	81
3.1.4 Canonical	81
3.1.5 Cisco	82
3.2 Entreprises innovantes	82
3.2.1 Services de stockage en ligne	82
3.2.2 Informatique dématérialisée	82
3.2.3 Forge	82
3.2.4 Réseaux sociaux	83
3.3 Éditeurs de contenus	83
3.3.1 Disney Animation Studio	83
3.3.2 YouTube	83
3.3.3 Box ADSL	83
3.3.4 Spotify	83
3.4 Éditeurs de logiciels	83
4. Retours d'expérience	84
4.1 Internet des objets	84
4.2 Système et développement web	85
4.3 Enseignement	85
4.4 Embarqué	86

4.5 Développement web..... 87
4.6 ERP..... 87

Chapitre 1.4
Installer son environnement de travail

1. Introduction..... 89
2. Installer Python..... 89
 2.1 Pour Windows..... 89
 2.2 Pour Mac..... 92
 2.3 Pour GNU/Linux et BSD..... 92
 2.4 Par la compilation..... 93
 2.5 Pour un smartphone..... 94
3. Installer une bibliothèque tierce..... 94
 3.1 À partir de Python 3.4..... 94
 3.2 Pour une version inférieure à Python 3.4..... 96
 3.3 Pour Linux..... 96
4. Créer un environnement virtuel..... 97
 4.1 À quoi sert un environnement virtuel ?..... 97
 4.2 Pour Python 3.3 ou version supérieure..... 97
 4.3 Pour toute version de Python..... 98
 4.4 Pour Linux..... 99
5. Installer Anaconda..... 100
 5.1 Pour Windows..... 100
 5.2 Pour Linux..... 103
 5.3 Pour Mac..... 103
 5.4 Mettre à jour Anaconda..... 104
 5.5 Installer une bibliothèque externe..... 104
 5.6 Environnements virtuels..... 104
6. Docker..... 104
7. La console Python..... 105
 7.1 Démarrer la console Python..... 105
 7.2 BPython..... 105
 7.3 IPython..... 106
 7.4 IPython Notebook..... 106

8.	Installer un IDE	107
8.1	Liste d'IDE	107
8.2	Présentation de PyCharm	107
8.3	Configuration de PyCharm	108

Partie 2 : Guide Python

Chapitre 2.1

Les premiers pas

1.	Avant de commencer	113
1.1	Quelques notions importantes	113
1.1.1	Comment fonctionne un ordinateur ?	113
1.1.2	Qu'est-ce qu'un programme informatique ?	114
1.1.3	Qu'est-ce qu'un code source ?	114
1.2	Quelques conventions utilisées dans ce livre	114
1.2.1	Code Python	114
1.2.2	Terminal	115
1.2.3	Mise en forme	115
1.3	Quelle est la meilleure méthode pour apprendre ?	116
2.	Premier programme	116
2.1	Hello world !	116
2.2	Affectation	118
2.3	Valeur booléenne	119
2.4	Type	120
2.5	Exceptions	121
2.6	Bloc conditionnel	124
2.7	Conditions avancées	125
2.8	Bloc itératif	126
3.	Premier jeu : Devine le nombre	128
3.1	Description du jeu	128
3.2	Aides	128
3.2.1	Gestion du hasard	128
3.2.2	Étapes de développement	128
3.3	Pour aller plus loin	129

Chapitre 2.2
Fonctions et modules

- 1. Les fonctions 131
 - 1.1 Pourquoi utiliser des fonctions ? 131
 - 1.2 Introduction aux fonctions 133
 - 1.2.1 Comment déclarer une fonction. 133
 - 1.2.2 Gestion d'un paramètre. 134
 - 1.2.3 Comment rendre une fonction plus générique 136
 - 1.2.4 Paramètres par défaut 138
 - 1.3 Problématiques de couplage et duplication de code 139
 - 1.3.1 Niveau de ses fonctions. 139
 - 1.3.2 Notion de complexité 141
 - 1.3.3 Bonnes pratiques 143
- 2. Les modules 144
 - 2.1 Introduction 144
 - 2.1.1 Qu'est-ce qu'un module ? 144
 - 2.1.2 Comment crée-t-on un module Python ? 145
 - 2.1.3 Organiser son code 145
 - 2.2 Gérer le code de ses modules 145
 - 2.2.1 Exécuter un module, importer un module. 145
 - 2.2.2 Gérer une arborescence de modules 146
- 3. Terminer le jeu. 147
 - 3.1 Créer des niveaux 148
 - 3.2 Déterminer un nombre de coups maximal 148
 - 3.3 Enregistrer les meilleurs scores. 148
 - 3.4 Intelligence artificielle 148

Chapitre 2.3
Les principaux types

- 1. Chaînes de caractères 149
 - 1.1 Syntaxe 149
 - 1.2 Formatage d'une chaîne 150
 - 1.3 Notion de casse. 151
 - 1.4 Notion de longueur. 152
 - 1.5 Appartenance 152
 - 1.6 Notion d'occurrence. 153

1.7	Remplacement	154
1.8	Notion de caractère	154
1.9	Typologie des caractères	155
1.10	Séquencer une chaîne de caractères	156
2.	Listes	156
2.1	Syntaxe	156
2.2	Indices	157
2.3	Valeurs	158
2.4	Hasard	159
2.5	Techniques d'itération	160
2.6	Tri	162
3.	Dictionnaires	164
3.1	Présentation des dictionnaires	164
3.2	Parcourir un dictionnaire	165
3.3	Exemple	165

Chapitre 2.4

Les classes

1.	Syntaxe	167
2.	Notion d'instance courante	168
3.	Opérateurs	170
4.	Héritage	172
4.1	Spécialisation	172
4.2	Programmation par composants	173

Partie 3 : Les fondamentaux du langage

Chapitre 3.1

Algorithmique de base

1.	Délimiteurs	175
1.1	Instruction	175
1.2	Une ligne de code = une instruction	175
1.3	Commentaire	176
1.4	Une instruction sur plusieurs lignes	176
1.5	Mots-clés	176

1.6	Mots réservés	177
1.7	Indentation	178
1.8	Symboles	179
1.9	Opérateurs	182
1.9.1	Opérateur expression d'affectation :=	185
1.10	Utilisation du caractère souligné	186
1.11	PEP-8	187
1.12	PEP-7	187
1.13	PEP-257	187
2.	Instructions	187
2.1	Définitions	187
2.1.1	Variable	187
2.1.2	Fonction	189
2.1.3	Fonctions lambda	190
2.1.4	Classe	191
2.1.5	Instruction vide	192
2.1.6	Suppression	192
2.1.7	Renvoyer le résultat de la fonction	193
2.2	Instructions conditionnelles	194
2.2.1	Définition	194
2.2.2	Condition	194
2.2.3	Instruction if	194
2.2.4	Instruction elif	195
2.2.5	Instruction else	195
2.3	Utilisation d'une expression d'affectation	197
2.3.1	Instruction switch	197
2.3.2	Interruptions	197
2.3.3	Approfondissement des conditions	198
2.3.4	Performances	199
2.4	Itérations	200
2.4.1	Instruction for	200
2.4.2	Instruction while	200
2.4.3	Quelle différence entre for et while ?	201
2.4.4	Instruction break	201
2.4.5	Instruction return	203
2.4.6	Instruction continue	203
2.4.7	Instruction else	203

2.4.8	Générateurs	204
2.5	Constructions fonctionnelles	207
2.5.1	Construction conditionnelle	207
2.5.2	Générateurs	207
2.5.3	Compréhensions de listes	207
2.5.4	Compréhensions d'ensembles	208
2.5.5	Compréhensions de dictionnaires	208
2.6	Compréhensions et expressions d'affectation	208
2.7	Gestion des exceptions	208
2.7.1	Présentation rapide des exceptions	208
2.7.2	Lever une exception	209
2.7.3	Pourquoi lever une exception ?	209
2.7.4	Assertions	210
2.7.5	Capturer une exception	211
2.7.6	Effectuer un traitement de l'exception	212
2.7.7	Gérer la sortie du bloc de capture	214
2.7.8	Gérer le non-déclenchement d'exceptions	214
2.7.9	Prise et libération de ressources	216
2.7.10	Programmation asynchrone	217
2.8	Divers	218
2.8.1	Gérer des imports	218
2.8.2	Traverser les espaces de nommage	219
2.8.3	Fonctions print, help, eval et exec	221

Chapitre 3.2

Déclarations

1.	Variable	223
1.1	Qu'est-ce qu'une variable ?	223
1.1.1	Contenu	223
1.1.2	Contenant	223
1.1.3	Modes de modification d'une variable	225
1.2	Typage dynamique	228
1.2.1	Affectation : rappels	228
1.2.2	Primitive type et nature du type	228
1.2.3	Caractéristiques du typage Python	229

1.3	Visibilité	231
1.3.1	Espace global	231
1.3.2	Notion de bloc	232
2.	Fonction	235
2.1	Déclaration	235
2.2	Paramètres	236
2.2.1	Signature d'une fonction	236
2.2.2	Notion d'argument ou de paramètre	237
2.2.3	Valeur par défaut	237
2.2.4	Valeur par défaut mutable	239
2.2.5	Paramètres nommés	240
2.2.6	Déclaration de paramètres extensibles	240
2.2.7	Passage de paramètres étoilés	242
2.2.8	Signature universelle	242
2.2.9	Obliger un paramètre à être nommé (keyword-only)	243
2.3	Obliger un paramètre à être positionnel (Positional-only)	245
2.3.1	Annotations	245
2.3.2	Types hint	249
3.	Classe	251
3.1	Déclaration	251
3.1.1	Signature	251
3.1.2	Attribut	251
3.1.3	Méthode	252
3.1.4	Bloc local	252
3.2	Instanciation	253
3.2.1	Syntaxe	253
3.2.2	Relation entre l'instance et la classe	253
4.	Module	254
4.1	À quoi sert un module ?	254
4.2	Déclaration	254
4.3	Instructions spécifiques	254
4.4	Comment appréhender le contenu d'un module ?	255
4.5	Compilation des modules	256

Chapitre 3.3

Modèle objet

1.	Tout est objet	259
1.1	Principes	259
1.1.1	Quel sens donner à « objet » ?	259
1.1.2	Adaptation de la théorie objet dans Python	260
1.1.3	Généralités	261
1.2	Classes	261
1.2.1	Introduction	261
1.2.2	Déclaration impérative d'une classe	262
1.2.3	Instance	262
1.2.4	Objet courant	264
1.2.5	Déclaration par prototype d'une classe	264
1.2.6	Tuples nommés	267
1.3	Méthodes	267
1.3.1	Déclaration	267
1.3.2	Appel de méthode	269
1.3.3	Méthodes et attributs spéciaux	271
1.3.4	Constructeur et initialisateur	275
1.3.5	Gestion automatisée des attributs	276
1.3.6	Intérêt du paradigme objet	276
1.3.7	Relation entre objets	277
1.4	Héritage	277
1.4.1	Polymorphisme par sous-typage	277
1.4.2	Surcharge de méthode	278
1.4.3	Surcharge des opérateurs	280
1.4.4	Polymorphisme paramétrique	281
1.4.5	Héritage multiple	283
2.	Autres outils de la programmation objet	285
2.1	Principes	285
2.2	Interfaces	285
2.3	Attributs	288
2.4	Propriétés	290
2.5	Emplacements	292
2.6	Métaclasses	294
2.7	Classes abstraites	296

- 2.8 La Zope Component Architecture. 299
 - 2.8.1 Présentation 299
 - 2.8.2 Installation 299
 - 2.8.3 Définir une interface et un composant 300
 - 2.8.4 Autres fonctionnalités. 301
 - 2.8.5 Avantages de la ZCA 301
- 3. Fonctions spéciales et primitives associées 301
 - 3.1 Personnalisation 301
 - 3.1.1 Classes 301
 - 3.1.2 Instances. 303
 - 3.1.3 Comparaison 304
 - 3.1.4 Évaluation booléenne 304
 - 3.1.5 Relations d’héritage ou de classe à instance. 305
 - 3.2 Classes particulières 305
 - 3.2.1 Itérateurs 305
 - 3.2.2 Conteneurs. 308
 - 3.2.3 Instances assimilables à des fonctions 308
 - 3.2.4 Ressources à protéger 309
 - 3.2.5 Types 310
 - 3.2.6 Classes de données. 310

Chapitre 3.4

Types de données et algorithmes appliqués

- 1. Nombres. 311
 - 1.1 Types 311
 - 1.1.1 Entiers. 311
 - 1.1.2 Réels 312
 - 1.1.3 Socle commun aux nombres entiers et réels 313
 - 1.1.4 Méthodes dédiées aux nombres entiers 314
 - 1.1.5 Méthodes dédiées aux nombres réels 315
 - 1.1.6 Complexes 315
 - 1.2 La console Python, la calculatrice par excellence 316
 - 1.2.1 Opérateurs mathématiques binaires 316
 - 1.2.2 Opérateurs binaires particuliers 317
 - 1.2.3 Opérateurs mathématiques unaires 318
 - 1.2.4 Arrondis 319
 - 1.2.5 Opérateurs de comparaison 321

1.2.6	Opérations mathématiques n-aires	322
1.2.7	Fonctions mathématiques usuelles	323
1.3	Représentations d'un nombre	329
1.3.1	Représentation décimale	329
1.3.2	Représentation par un exposant	329
1.3.3	Représentation par une fraction	329
1.3.4	Représentation hexadécimale	330
1.3.5	Représentation octale	331
1.3.6	Représentation binaire	332
1.3.7	Opérations binaires	332
1.3.8	Longueur de la représentation mémoire d'un entier	334
1.4	Conversions	335
1.4.1	Conversion entre entiers et réels	335
1.4.2	Conversion entre réels et complexes	336
1.4.3	Conversion vers un booléen	336
1.5	Travailler avec des variables	337
1.5.1	Un nombre est non mutable	337
1.5.2	Modifier la valeur d'une variable	338
1.5.3	Opérateurs d'incrément	338
1.6	Statistiques	339
2.	Séquences	340
2.1	Présentation des différents types de séquences	340
2.1.1	Généralités	340
2.1.2	Les listes	341
2.1.3	Les n-uplets	342
2.1.4	Conversion entre listes et n-uplets	344
2.1.5	Socle commun entre liste et n-uplet	344
2.1.6	Notion d'itérateur	345
2.2	Utilisation des indices et des tranches	347
2.2.1	Définition de l'indice d'un objet et des occurrences	347
2.2.2	Utiliser l'indice pour adresser la séquence	349
2.2.3	Retrouver les occurrences d'un objet et leurs indices	350
2.2.4	Taille d'une liste, comptage d'occurrences	351
2.2.5	Utiliser l'indice pour modifier ou supprimer	352
2.2.6	Itération simple	354
2.2.7	Présentation de la notion de tranches (slices)	357
2.2.8	Cas particulier de la branche 2.x de Python	366

2.2.9	Utilisation basique des tranches	367
2.2.10	Utilisation avancée des tranches	368
2.3	Utilisation des opérateurs	370
2.3.1	Opérateur +	370
2.3.2	Opérateur *	371
2.3.3	Opérateur +=	373
2.3.4	Opérateur *=	375
2.3.5	Opérateur in	376
2.3.6	Opérateurs de comparaison	377
2.4	Méthodes de modifications	378
2.4.1	Ajouter des éléments dans une liste et un n-uplet	378
2.4.2	Supprimer un objet d'une liste et d'un n-uplet	380
2.4.3	Solutions de contournement pour la modification de n-uplets	384
2.4.4	Renverser une liste ou un tuple	385
2.4.5	Trier une liste	386
2.5	Utilisation avancée des listes	388
2.5.1	Opérations d'ensemble	388
2.5.2	Pivoter une séquence	389
2.5.3	Itérer correctement	390
2.5.4	Programmation fonctionnelle	391
2.5.5	Compréhensions de listes	394
2.5.6	Itérations avancées	395
2.5.7	Combinatoire	400
2.6	Adapter les listes à des besoins spécifiques	402
2.6.1	Liste d'entiers	402
2.6.2	Présentation du type array	403
2.6.3	Utiliser une liste comme pile	405
2.6.4	Utiliser une liste comme file d'attente	406
2.6.5	Conteneur plus performant	406
2.6.6	Utiliser des listes pour représenter des matrices	407
2.6.7	Liste sans doublons	408
2.7	Autres types de données	411
3.	Ensembles	413
3.1	Présentation	413
3.1.1	Définition d'un ensemble	413
3.1.2	Différences entre set et frozenset	414

3.1.3	Utilisation pour dédoublonner des listes	415
3.1.4	Rajouter une relation d'ordre	415
3.2	Opérations ensemblistes	416
3.2.1	Opérateurs pour un ensemble à partir de deux autres	416
3.2.2	Opérateurs pour modifier un ensemble à partir d'un autre	417
3.2.3	Méthodes équivalentes à la création ou modification ensembliste	418
3.2.4	Méthodes de comparaison des ensembles	418
3.2.5	Exemples non classiques d'utilisation	419
3.3	Méthodes de modification d'un ensemble	423
3.3.1	Ajouter un élément	423
3.3.2	Supprimer un élément	423
3.3.3	Vider un ensemble	424
3.3.4	Dupliquer un élément	424
3.3.5	Sortir une valeur d'un ensemble	425
3.3.6	Utiliser un ensemble comme un recycleur d'objets	426
3.3.7	Algorithmique avancée : résolution du problème des n-dames	428
4.	Chaînes de caractères	430
4.1	Présentation	430
4.1.1	Définition	430
4.1.2	Vocabulaire	431
4.1.3	Spécificités de la branche 2.x	432
4.1.4	Changements apportés par la branche 3.x	433
4.1.5	Chaîne de caractères en tant que séquence de caractères	435
4.1.6	Caractères	437
4.1.7	Opérateurs de comparaison	438
4.2	Formatage de chaînes de caractères	441
4.2.1	Opérateur modulo	441
4.2.2	Méthodes de formatage sur l'ensemble de la chaîne	446
4.2.3	Nouvelle méthode de formatage des variables dans une chaîne	448
4.2.4	Littéraux formatés	451
4.3	Opérations d'ensemble	452
4.3.1	Séquençage de chaînes	452
4.3.2	Opérations sur la casse	454
4.3.3	Recherche sur une chaîne de caractères	455

4.3.4	Informations sur les caractères	456
4.4	Problématiques relatives à l'encodage	457
4.4.1	Encodage par défaut	457
4.4.2	Encodage du système.	458
4.4.3	L'unicode, référence absolue	458
4.4.4	Autres encodages	459
4.4.5	Ponts entre l'unicode et le reste du monde	460
4.4.6	Revenir vers l'Unicode.	461
4.5	Manipulations de bas niveau avancées	461
4.5.1	Opérations de comptage	461
4.5.2	Une chaîne de caractères vue comme une liste	462
4.5.3	Une chaîne de caractères vue comme un ensemble de caractères	463
4.6	Représentation mémoire	463
4.6.1	Présentation du type bytes	463
4.6.2	Lien avec les chaînes de caractères	464
4.6.3	Présentation du type bytearray	466
4.6.4	Gestion d'un jeu de caractères	468
5.	Dictionnaires	473
5.1	Présentation	473
5.1.1	Définition.	473
5.1.2	Évolutions et différences entre les branches 2.x et 3.x	474
5.1.3	Vues de dictionnaires.	475
5.1.4	Instanciation	477
5.1.5	Compréhension de dictionnaire	477
5.2	Manipuler un dictionnaire	478
5.2.1	Récupérer une valeur d'un dictionnaire	478
5.2.2	Modifier les valeurs d'un dictionnaire	479
5.2.3	Supprimer une entrée d'un dictionnaire	480
5.2.4	Dupliquer un dictionnaire.	480
5.2.5	Utiliser le dictionnaire comme agrégateur de données	481
5.2.6	Méthodes d'itération.	482
5.3	Utilisation avancée des dictionnaires	482
5.3.1	Rajouter une relation d'ordre	482
5.3.2	Algorithmiques classiques.	486
5.3.3	Adapter les dictionnaires à des besoins spécifiques	488
5.3.4	Représentation universelle de données	490

6.	Booléens	491
6.1	Le type booléen	491
6.1.1	Classe bool	491
6.1.2	Les deux objets True et False	492
6.1.3	Différence entre l'opérateur d'égalité et d'identité	492
6.2	Évaluation booléenne	492
6.2.1	Méthode générique	492
6.2.2	Objets classiques	492
7.	Données temporelles	493
7.1	Gérer une date calendaire	493
7.1.1	Notion de date calendaire	493
7.1.2	Travailler sur une date	494
7.1.3	Considérations astronomiques	495
7.1.4	Considérations historiques	495
7.1.5	Considérations techniques	495
7.1.6	Représentation textuelle	496
7.2	Gérer un horaire ou un moment d'une journée	498
7.2.1	Notion d'instant	498
7.2.2	Notion de fuseau horaire	499
7.2.3	Représentation textuelle	499
7.3	Gérer un instant absolu	500
7.3.1	Notion d'instant absolu	500
7.3.2	Rapport avec les notions précédentes	501
7.3.3	Représentation textuelle	502
7.3.4	Gestion des fuseaux horaires	503
7.3.5	Créer une date à partir d'une représentation textuelle	503
7.4	Gérer une différence entre deux dates ou instants	503
7.4.1	Notion de différence et de résolution	503
7.4.2	Considérations techniques	505
7.4.3	Utilisation avec des dates calendaires	506
7.4.4	Utilisation avec des horaires	506
7.4.5	Utilisation avec des dates absolues	506
7.4.6	La seconde comme unité de base	506
7.4.7	Précision à la nanoseconde	507
7.5	Spécificités des fuseaux horaires	507

- 7.6 Problématiques de bas niveau 508
 - 7.6.1 Timestamp et struct_time 508
 - 7.6.2 Mesures de performances 509
- 7.7 Utilisation du calendrier. 511
 - 7.7.1 Présentation du module calendar 511
 - 7.7.2 Fonctions essentielles du calendrier 516

Partie 4 : Les fonctionnalités

Chapitre 4.1
Manipulation de données

- 1. Manipuler des fichiers 519
 - 1.1 Ouvrir un fichier 519
 - 1.2 Lire un fichier 520
 - 1.3 Écrire un fichier. 521
 - 1.4 Comparer deux fichiers 522
- 2. Utilitaire de sauvegarde. 524
- 3. Lire un fichier de configuration 524
- 4. Format d'export/Import 525
 - 4.1 CSV 525
 - 4.1.1 Exploiter un fichier CSV 526
 - 4.1.2 Génération d'un fichier CSV 529
 - 4.2 JSON 531
 - 4.3 Base64 534
 - 4.4 Pickle 534
- 5. Compresser et décompresser un fichier 537
 - 5.1 Tarfile 537
 - 5.2 Gzip 539
 - 5.3 Bz2 539
 - 5.4 Zipfile 540
 - 5.5 Interface de haut niveau. 542
- 6. Outils de manipulation de données 543
 - 6.1 Générer des nombres aléatoires 543
 - 6.2 Expressions régulières. 544

7.	Cryptographie légère	548
7.1	Nombre aléatoire sécurisé	548
7.2	Fonctions de chiffrement	548
7.3	Code d'authentification de message	550
7.4	Empreinte de fichier	551
7.5	Stéganographie	552
7.6	Communication inter-applicative sécurisée	555

Chapitre 4.2

Bases de données

1.	Introduction	559
2.	Accès à une base de données relationnelle	559
2.1	Point d'entrée	559
2.2	MySQL	560
2.3	PostgreSQL	565
2.4	SQLite	567
2.5	Oracle	567
3.	Utilisation d'un ORM	568
3.1	Qu'est-ce qu'un ORM ?	568
3.2	ORM proposés par Python	568
3.3	SQLAlchemy	569
4.	Autres bases de données	575
4.1	CSV	575
4.2	NoSQL	581
4.3	Base de données orientée objet : ZODB	581
4.4	Base de données orientée graphe : Neo4j	586
4.5	Base de données de type clé-valeur : Redis	587
4.6	Bases de données orientées documents : CouchDB et MongoDB	589
4.7	Bases de données natives XML : BaseX, eXist	590
4.8	Cassandra	591
4.9	Bases de données orientées colonnes : HBase	591
4.10	Big Data : l'écosystème Hadoop	593
5.	LDAP	595
5.1	Protocole	595
5.2	Serveurs	595
5.3	Terminologie	596

- 5.4 Installation 596
- 5.5 Ouvrir une connexion à un serveur 596
- 5.6 Effectuer une recherche 598
- 5.7 Synchrone vs asynchrone 599
- 5.8 Connexions sécurisées 599

Partie 5 : Mise en pratique

Chapitre 5.1

Créer une application web en 30 minutes

- 1. Description de l'application à construire 601
- 2. Mise en place 602
 - 2.1 Isolation de l'environnement 602
 - 2.2 Création du projet 603
 - 2.3 Paramétrage 603
 - 2.4 Premiers essais 604
- 3. Réalisation de l'application 605
 - 3.1 Modèles 605
 - 3.2 Vues 608
 - 3.3 Contrôleurs 609
- 4. Pour aller plus loin 613

Chapitre 5.2

Créer une application console en 10 minutes

- 1. Objectif 615
- 2. Enregistrer le script 616
- 3. Création des données 616
- 4. Parseur d'arguments 617

Chapitre 5.3

Créer une application graphique en 20 minutes

- 1. Objectif 619
 - 1.1 Fonctionnel 619
 - 1.2 Technique 619

2. Présentation rapide de Gtk et d'astuces	620
2.1 Présentation	620
2.2 Astuces	620
3. Démarrer le programme	621
4. Interface graphique avec Glade	623
5. Créer le composant graphique	626
6. Contrôleur	628
7. Autres bibliothèques graphiques	629
7.1 TkInter	629
7.2 wxPython	629
7.3 PyQt	629
7.4 PySide	630
7.5 Autres	630

Chapitre 5.4

Créer un jeu en 30 minutes avec PyGame

1. Présentation de PyGame	631
2. Réalisation d'un jeu Tetris	632
2.1 Présentation du jeu	632
2.2 Présentation des problématiques	633
2.3 Création des constantes	633

Annexes

1. Objets mutables et non mutables	645
2. Table Unicode	647
2.1 Script	647
3. Bytes	647
3.1 Script	647
3.2 Résultat	647
4. Guide de portage vers Python 3	650

Index	653
-----------------	-----