

# Entrées/sorties

Pratique de la programmation orientée-objet  
Michel Schinz – 2019-03-25

# Flots en Java (`java.io`)

Le flot (*stream*) est l'abstraction centrale des entrées/sorties de `java.io`. Il y en a quatre types :

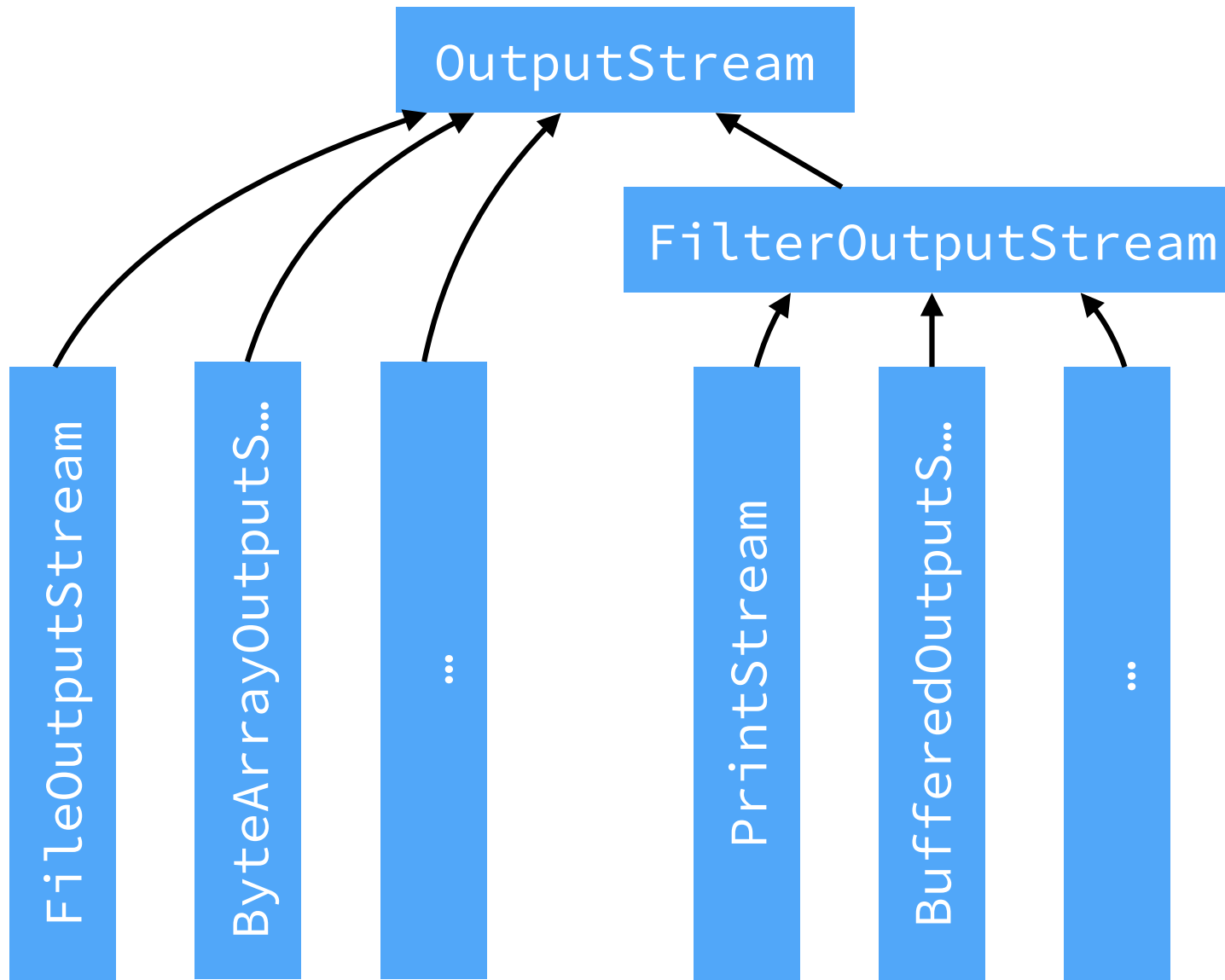
	Entrée	Sortie
Octets	<code>InputStream</code>	<code>OutputStream</code>
Caractères	<code>Reader</code>	<code>Writer</code>

A chaque type de flot, représenté par une classe abstraite, correspond une hiérarchie.

# OutputStream

```
abstract public class OutputStream {  
    // Ecrit les 8 bits de poids faible de b  
    // dans le flot.  
    abstract void write(int b);  
  
    // Ecrit la totalité des octets de b dans  
    // le flot.  
    void write(byte[] b);  
  
    // Ferme le flot.  
    void close();  
}
```

# Flots de sortie d'octets



# InputStream

```
abstract public class InputStream {  
    // Lit et retourne le prochain octet du  
    // flot (entier entre 0 et 255) ou -1.  
    abstract int read();  
  
    // Lit et place dans le tableau b les  
    // prochains octets du flot, retourne le  
    // nombre lu ou -1.  
    int read(byte[] b);  
  
    // Ferme le flot.  
    void close();  
}
```

# Représentation des caractères

# ASCII (1975)

1 caractère = 7 bits (128 caractères)

PDF : fr en v · d · m	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>E</b>	<b>F</b>
<b>000</b>	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
<b>001</b>	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
<b>002</b>	SP	!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
<b>003</b>	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
<b>004</b>	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
<b>005</b>	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
<b>006</b>	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
<b>007</b>	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL



caractères de contrôle



caractères imprimables

# ISO 8859-1 (1985)

1 caractère = 8 bits (256 caractères)

ISO-8859-1																
	-0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-A	-B	-C	-D	-E	-F
0-	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BS	HT	LF	VT	FF	CR	SO	SI	
	0000	0001	0002	0003	0004	0005	0006	0007	0008	0009	000A	000B	000C	000D	000E	000F
1-	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
	0010	0011	0012	0013	0014	0015	0016	0017	0018	0019	001A	001B	001C	001D	001E	001F
2-	SP	!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
	0020	0021	0022	0023	0024	0025	0026	0027	0028	0029	002A	002B	002C	002D	002E	002F
3-	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
	0030	0031	0032	0033	0034	0035	0036	0037	0038	0039	003A	003B	003C	003D	003E	003F
4-	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
	0040	0041	0042	0043	0044	0045	0046	0047	0048	0049	004A	004B	004C	004D	004E	004F
5-	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
	0050	0051	0052	0053	0054	0055	0056	0057	0058	0059	005A	005B	005C	005D	005E	005F
6-	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
	0060	0061	0062	0063	0064	0065	0066	0067	0068	0069	006A	006B	006C	006D	006E	006F
7-	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL
	0070	0071	0072	0073	0074	0075	0076	0077	0078	0079	007A	007B	007C	007D	007E	007F
8-	PAD	HOP	BPH	NBH	IND	NEL	SSA	ESA	HTS	HTJ	VTS	PLD	PLU	RI	SSI	SSJ
	0080	0081	0082	0083	0084	0085	0086	0087	0088	0089	008A	008B	008C	008D	008E	008F
9-	DCS	FUL	FU2	STS	CCH	MW	SPA	EPA	SOS	SGCI	SCI	CSI	ST	OSC	SW	APC
	0090	0091	0092	0093	0094	0095	0096	0097	0098	0099	009A	009B	009C	009D	009E	009F
A-	NSBP	j	ç	£	¤	¥	¦	§	¨	©	ª	«	¬	®	¯	°
	00A0	00A1	00A2	00A3	00A4	00A5	00A6	00A7	00A8	00A9	00AA	00AB	00AC	00AD	00AE	00AF
B-	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
	00B0	00B1	00B2	00B3	00B4	00B5	00B6	00B7	00B8	00B9	00BA	00BB	00BC	00BD	00BE	00BF
C-	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
	00C0	00C1	00C2	00C3	00C4	00C5	00C6	00C7	00C8	00C9	00CA	00CB	00CC	00CD	00CE	00CF
D-	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
	00D0	00D1	00D2	00D3	00D4	00D5	00D6	00D7	00D8	00D9	00DA	00DB	00DC	00DD	00DE	00DF
E-	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
	00E0	00E1	00E2	00E3	00E4	00E5	00E6	00E7	00E8	00E9	00EA	00EB	00EC	00ED	00EE	00EF
F-	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ
	00F0	00F1	00F2	00F3	00F4	00F5	00F6	00F7	00F8	00F9	00FA	00FB	00FC	00FD	00FE	00FF
	-0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-A	-B	-C	-D	-E	-F

ASCII



# Unicode (1987-...)

1 caractère = 20 bits (~1 million de caractères)

	000	001	002	003	004	005	006	007
0	<span>NUL</span>	<span>DLE</span>	<span>SP</span>	0	@	P	`	p
1	<span>SOH</span>	<span>DC1</span>	!	1	A	Q	a	q
2	<span>STX</span>	<span>DC2</span>	"	2	B	R	b	r
3	<span>ETX</span>	<span>DC3</span>	#	3	C	S	c	s
4	<span>EOT</span>	<span>DC4</span>	\$	4	D	T	d	t
5	<span>ENQ</span>	<span>NAK</span>	%	5	E	U	e	u
6	<span>ACK</span>	<span>SYN</span>	&	6	F	V	f	v
7	<span>BEL</span>	<span>ETB</span>	'	7	G	W	g	w
8	<span>BS</span>	<span>CAN</span>	(	8	H	X	h	x
9	<span>HT</span>	<span>EM</span>	)	9	I	Y	i	y
A	<span>LF</span>	<span>SUB</span>	*	:	J	Z	j	z
B	<span>VT</span>	<span>ESC</span>	+	;	K	[	k	{
C	<span>FF</span>	<span>FS</span>	,	<	L	\	l	
D	<span>CR</span>	<span>GS</span>	-	=	M	]	m	}
E	<span>SO</span>	<span>RS</span>	.	>	N	^	n	~
F	<span>SI</span>	<span>US</span>	/	?	O	_	o	<span>DEL</span>

	008	009	00A	00B	00C	00D	00E	00F
0	<span>XXX</span>	<span>NB SP</span>	°	À	Ð	à	ð	
1	<span>XXX</span>	<span>PU1</span>	¡	±	Á	Ñ	á	ñ
2	<span>BPH</span>	<span>PU2</span>	¢	²	Â	Ô	â	ô
3	<span>NBH</span>	<span>STS</span>	£	³	Ã	Ó	ã	ó
4	<span>IND</span>	<span>CCH</span>	¤	´	Ä	Ö	ä	ö
5	<span>NEL</span>	<span>NW</span>	¥	µ	Å	Õ	å	õ
6	<span>SSA</span>	<span>SPA</span>	¦	¶	Æ	Ö	æ	ö
7	<span>ESA</span>	<span>EPA</span>	§	·	Ç	×	ç	÷
8	<span>WTS</span>	<span>SOS</span>	¨	¸	È	Ø	è	ø
9	<span>HTJ</span>	<span>XXX</span>	©	¹	É	Ù	é	ù
A	<span>VTS</span>	<span>SCI</span>	ª	º	Ê	Ú	ê	ú
B	<span>PLD</span>	<span>CSI</span>	«	»	Ë	Û	ë	û
C	<span>PLU</span>	<span>ST</span>	¬	¼	Ì	Ü	ì	ü
D	<span>RI</span>	<span>OSC</span>	½	½	Í	Ý	í	ý
E	<span>SS2</span>	<span>PM</span>	¾	¾	Î	Þ	î	þ
F	<span>SS3</span>	<span>APC</span>	¿	¿	Ï	ß	ï	ÿ

...















ASCII

ISO 8859-1

















































































# Unicode (1987-...)

	2A0	2A1	2A2	2A3	2A4	2A5	2A6	2A7	2A8	2A9	2AA	2AB	2AC	2AD	2AE	2AF
0	2A00	2A10	2A20	2A30	2A40	2A50	2A60	2A70	2A80	2A90	2AA0	2AB0	2AC0	2AD0	2AE0	2AF0
1	2A01	2A11	2A21	2A31	2A41	2A51	2A61	2A71	2A81	2A91	2AA1	2AB1	2AC1	2AD1	2AE1	2AF1
2	2A02	2A12	2A22	2A32	2A42	2A52	2A62	2A72	2A82	2A92	2AA2	2AB2	2AC2	2AD2	2AE2	2AF2
3	2A03	2A13	2A23	2A33	2A43	2A53	2A63	2A73	2A83	2A93	2AA3	2AB3	2AC3	2AD3	2AE3	2AF3
4	2A04	2A14	2A24	2A34	2A44	2A54	2A64	2A74	2A84	2A94	2AA4	2AB4	2AC4	2AD4	2AE4	2AF4
5	2A05	2A15	2A25	2A35	2A45	2A55	2A65	2A75	2A85	2A95	2AA5	2AB5	2AC5	2AD5	2AE5	2AF5
6	2A06	2A16	2A26	2A36	2A46	2A56	2A66	2A76	2A86	2A96	2AA6	2AB6	2AC6	2AD6	2AE6	2AF6
7	2A07	2A17	2A27	2A37	2A47	2A57	2A67	2A77	2A87	2A97	2AA7	2AB7	2AC7	2AD7	2AE7	2AF7
8	2A08	2A18	2A28	2A38	2A48	2A58	2A68	2A78	2A88	2A98	2AA8	2AB8	2AC8	2AD8	2AE8	2AF8
9	2A09	2A19	2A29	2A39	2A49	2A59	2A69	2A79	2A89	2A99	2AA9	2AB9	2AC9	2AD9	2AE9	2AF9
A	2A0A	2A1A	2A2A	2A3A	2A4A	2A5A	2A6A	2A7A	2A8A	2A9A	2AA A	2ABA	2ACA	2ADA	2AEA	2AFA
B	2A0B	2A1B	2A2B	2A3B	2A4B	2A5B	2A6B	2A7B	2A8B	2A9B	2AAB	2ABB	2ACB	2ADB	2AEB	2AFB
C	2A0C	2A1C	2A2C	2A3C	2A4C	2A5C	2A6C	2A7C	2A8C	2A9C	2AAC	2ABC	2ACC	2ADC	2AEC	2AFC
D	2A0D	2A1D	2A2D	2A3D	2A4D	2A5D	2A6D	2A7D	2A8D	2A9D	2AAD	2ABD	2ACD	2ADD	2AED	2AFD
E	2A0E	2A1E	2A2E	2A3E	2A4E	2A5E	2A6E	2A7E	2A8E	2A9E	2AAE	2ABE	2ACE	2ADE	2AEE	2AFE
F	2A0F	2A1F	2A2F	2A3F	2A4F	2A5F	2A6F	2A7F	2A8F	2A9F	2AAF	2ABF	2ACF	2ADF	2AEF	2AFF

# Unicode (1987-...)

	1300	1301	1302	1303	1304	1305	1306	1307	1308	1309	130A	130B	130C	130D
0	 13000	 13010	 13020	 13030	 13040	 13050	 13060	 13070	 13080	 13090	 130A0	 130B0	 130C0	 130D0
1	 13001	 13011	 13021	 13031	 13041	 13051	 13061	 13071	 13081	 13091	 130A1	 130B1	 130C1	 130D1
2	 13002	 13012	 13022	 13032	 13042	 13052	 13062	 13072	 13082	 13092	 130A2	 130B2	 130C2	 130D2
3	 13003	 13013	 13023	 13033	 13043	 13053	 13063	 13073	 13083	 13093	 130A3	 130B3	 130C3	 130D3
4	 13004	 13014	 13024	 13034	 13044	 13054	 13064	 13074	 13084	 13094	 130A4	 130B4	 130C4	 130D4
5	 13005	 13015	 13025	 13035	 13045	 13055	 13065	 13075	 13085	 13095	 130A5	 130B5	 130C5	 130D5
6	 13006	 13016	 13026	 13036	 13046	 13056	 13066	 13076	 13086	 13096	 130A6	 130B6	 130C6	 130D6
7	 13007	 13017	 13027	 13037	 13047	 13057	 13067	 13077	 13087	 13097	 130A7	 130B7	 130C7	 130D7
8	 13008	 13018	 13028	 13038	 13048	 13058	 13068	 13078	 13088	 13098	 130A8	 130B8	 130C8	 130D8
9	 13009	 13019	 13029	 13039	 13049	 13059	 13069	 13079	 13089	 13099	 130A9	 130B9	 130C9	 130D9
A	 1300A	 1301A	 1302A	 1303A	 1304A	 1305A	 1306A	 1307A	 1308A	 1309A	 130AA	 130BA	 130CA	 130DA
B	 1300B	 1301B	 1302B	 1303B	 1304B	 1305B	 1306B	 1307B	 1308B	 1309B	 130AB	 130BB	 130CB	 130DB
C	 1300C	 1301C	 1302C	 1303C	 1304C	 1305C	 1306C	 1307C	 1308C	 1309C	 130AC	 130BC	 130CC	 130DC
D	 1300D	 1301D	 1302D	 1303D	 1304D	 1305D	 1306D	 1307D	 1308D	 1309D	 130AD	 130BD	 130CD	 130DD
E	 1300E	 1301E	 1302E	 1303E	 1304E	 1305E	 1306E	 1307E	 1308E	 1309E	 130AE	 130BE	 130CE	 130DE
F	 1300F	 1301F	 1302F	 1303F	 1304F	 1305F	 1306F	 1307F	 1308F	 1309F	 130AF	 130BF	 130CF	 130DF

# Unicode (1987-...)

	1F60	1F61	1F62	1F63	1F64
0	 1F600	 1F610	 1F620	 1F630	 1F640
1	 1F601	 1F611	 1F621	 1F631	 1F641
2	 1F602	 1F612	 1F622	 1F632	 1F642
3	 1F603	 1F613	 1F623	 1F633	 1F643
4	 1F604	 1F614	 1F624	 1F634	 1F644
5	 1F605	 1F615	 1F625	 1F635	 1F645
6	 1F606	 1F616	 1F626	 1F636	 1F646
7	 1F607	 1F617	 1F627	 1F637	 1F647
8	 1F608	 1F618	 1F628	 1F638	 1F648
9	 1F609	 1F619	 1F629	 1F639	 1F649
A	 1F60A	 1F61A	 1F62A	 1F63A	 1F64A
B	 1F60B	 1F61B	 1F62B	 1F63B	 1F64B
C	 1F60C	 1F61C	 1F62C	 1F63C	 1F64C
D	 1F60D	 1F61D	 1F62D	 1F63D	 1F64D
E	 1F60E	 1F61E	 1F62E	 1F63E	 1F64E
F	 1F60F	 1F61F	 1F62F	 1F63F	 1F64F

# UTF-8

Encodage d'Unicode à taille variable (1 à 4 octets par caractère).

Plage	octet 1	octet 2	octet 3	octet 4
0 à 7F	0xxxxxxx			
... à 7FF	110xxxxx	10xxxxxx		
... à FFFF	1110xxxx	10xxxxxx	10xxxxxx	
... à 1FFFFFFF	11110xxx	10xxxxxx	10xxxxxx	10xxxxxx

# UTF-16

Encodage d'Unicode à taille variable (2 ou 4 octets par caractère) :

- de 0 à D7FF et de E000 à FFFF : représentés tels quels,
- de 10000 à 10FFFF : représentés par 4 octets, au moyen de *surrogate pairs*.

# Reader

```
abstract public class Reader {  
    // Lit et retourne le prochain caractère du  
    // flot (entier entre 0 et  $2^{16}-1$ ) ou -1.  
    int read();  
    // Lit et place dans le tableau c les  
    // prochains caractères du flot, retourne  
    // le nombre lu ou -1.  
    int read(char[] c);  
    // Ferme le flot.  
    void close();  
}
```

# Writer

```
abstract public class Writer {  
    // Ecrit les 16 bits de poids faible de c  
    // dans le flot.  
    void write(int c);  
    // Ecrit la totalité des caractères de a  
    // dans le flot.  
    void write(char[] a);  
    // Ferme le flot.  
    void close();  
}
```