

CSE 201

JAVA PROGRAMMING I

Primitive Data Type

byte

short

int

long

float

double

boolean

char

Primitive Data Type

byte

8-bit signed

Two's complement

Integer

-128 ~ 127

Primitive Data Type

short

16-bit signed
Two's complement
Integer
 $-32768 \sim 32767$

Primitive Data Type

int

32-bit signed
Two's complement
Integer
 $-(2^{31}) \sim (2^{31})-1$

Primitive Data Type

long

64-bit signed
Two's complement
Integer
 $-(2^{63}) \sim (2^{63})-1$

Primitive Data Type

float

32-bit

IEEE 754 floating point

i.e. 12.42581

Primitive Data Type

double

64-bit

IEEE 754 floating point

i.e. 12.42581214314

Primitive Data Type

float

double

Never use float or double to represent precise values

Such as currency

Solution: [java.math.BigDecimal](#) class

Primitive Data Type

Convention between data types

Small size data type → Big size data type **OK**
i.e. int → long

Small size data type ↯ Big size data type **NO**
i.e. double → int
(data loss)

Primitive Data Type

On June 4, 1996 Ariane 5 rocket launched by the European Space Agency exploded just 40 seconds after its lift-off from Kourou, French Guiana.



It costs \$7 billion Development
+ \$500 million cargo loss

Reason: Convention failed
between floating number and 16-bit integer

Primitive Data Type

boolean

true, false

This data type represents
1 bit of information

Primitive Data Type

char

16-bit Unicode character

Dec	Hx	Char	Dec	Hx	HTML	Char	Dec	Hx	HTML	Char	Dec	Hx	HTML	Char
0	0	NUL (null)	32	20	 	Space	64	40	@	@	96	60	`	`
1	1	SOH (Start of heading)	33	21	!	!	65	41	A	A	97	61	a	a
2	2	STX (Start of text)	34	22	"	"	66	42	B	B	98	62	b	b
3	3	ETX (End of text)	35	23	#	#	67	43	C	C	99	63	c	c
4	4	EOT (End of transmission)	36	24	$	\$	68	44	D	D	100	64	d	d
5	5	ENQ (Enquiry)	37	25	%	%	69	45	E	E	101	65	e	e
6	6	ACK (Acknowledge)	38	26	&	&	70	46	F	F	102	66	f	f
7	7	BEL (Bell)	39	27	'	'	71	47	G	G	103	67	g	g
8	8	BS (Backspace)	40	28	((72	48	H	H	104	68	h	h
9	9	TAB (Horizontal tab)	41	29))	73	49	I	I	105	69	i	i
10	A	LF (NL line fd, new line)	42	2A	*	*	74	4A	J	J	106	6A	j	j
11	B	VT (Vertical tab)	43	2B	+	+	75	4B	K	K	107	6B	k	k
12	C	FF (NP form fd, new page)	44	2C	,	,	76	4C	L	L	108	6C	l	l
13	D	CR (Carriage return)	45	2D	-	-	77	4D	M	M	109	6D	m	m
14	E	SO (Shift out)	46	2E	.	.	78	4E	N	N	110	6E	n	n
15	F	SI (Shift in)	47	2F	/	/	79	4F	O	O	111	6F	o	o
16	10	DLE (Data link escape)	48	30	0	0	80	50	P	P	112	70	p	p
17	11	DC1 (Device control 1)	49	31	1	1	81	51	Q	Q	113	71	q	q
18	12	DC2 (Device control 2)	50	32	2	2	82	52	R	R	114	72	r	r
19	13	DC3 (Device control 3)	51	33	3	3	83	53	S	S	115	73	s	s
20	14	DC4 (Device control 4)	52	34	4	4	84	54	T	T	116	74	t	t
21	15	NAK (Negative acknowledge)	53	35	5	5	85	55	U	U	117	75	u	u
22	16	SYN (Synchronous idle)	54	36	6	6	86	56	V	V	118	76	v	v
23	17	ETB (End of trans. block)	55	37	7	7	87	57	W	W	119	77	w	w
24	18	CAN (Cancel)	56	38	8	8	88	58	X	X	120	78	x	x
25	19	EM (End of medium)	57	39	9	9	89	59	Y	Y	121	79	y	y
26	1A	SUB (Substitute)	58	3A	:	:	90	5A	Z	Z	122	7A	z	z
27	1B	ESC (Escape)	59	3B	;	;	91	5B	[[123	7B	{	{
28	1C	FS (File separator)	60	3C	<	<	92	5C	\	\	124	7C	|	
29	1D	GS (Group separator)	61	3D	=	=	93	5D]]	125	7D	}	}
30	1E	RS (Record separator)	62	3E	>	>	94	5E	^	^	126	7E	~	~
31	1F	US (Unit separator)	63	3F	?	?	95	5F	_	_	127	7F		DEL

Binary Number

1010111000011101010000111110001010101

0's or 1's are allowed only

Binary Number

Decimal to Binary Number

$$14 = 1110$$

Binary Number

Binary Number to Decimal

$$10101 = 21$$

Binary Number

Decimal Number: 5428

10^4 = $10 * 10 * 10 * 10$ = 10000	10^3 = $10 * 10 * 10$ = 1000	10^2 = $10 * 10$ = 100	10^1 = 10	10^0 = 1
0	5	4	2	8

Binary Number

Decimal Number: 5428

10^4 = $10 * 10 * 10 * 10$ = 10000	10^3 = $10 * 10 * 10$ = 1000	10^2 = $10 * 10$ = 100	10^1 = 10	10^0 = 1
0	5	4	2	8

$$\begin{aligned} & 0 * 10000 + 5 * 1000 + 4 * 100 + 2 * 10 + 8 \\ & = 0 + 5000 + 400 + 20 + 8 \\ & = 5428 \end{aligned}$$

Binary Number

Binary Number: 1101 à Decimal Number: ??

2^4 $= 2 * 2 * 2 * 2$ $= 16$	2^3 $= 2 * 2 * 2$ $= 8$	2^2 $= 2 * 2$ $= 4$	2^1 $= 2$	2^0 $= 1$

Binary Number

Binary Number: 1101 à Decimal Number: 13

2^4 $= 2 * 2 * 2 * 2$ $= 16$	2^3 $= 2 * 2 * 2$ $= 8$	2^2 $= 2 * 2$ $= 4$	2^1 $= 2$	2^0 $= 1$
0	1	1	0	1

$$\begin{aligned} & 0 * 16 + 1 * 8 + 1 * 4 + 0 * 2 + 1 * 1 \\ & = 0 + 8 + 4 + 0 + 1 \\ & = 13 \end{aligned}$$

Binary Number

Decimal Number: 12 à Binary Number: ??

2^4 $= 2 * 2 * 2 * 2$ $= 16$	2^3 $= 2 * 2 * 2$ $= 8$	2^2 $= 2 * 2$ $= 4$	2^1 $= 2$	2^0 $= 1$



12 >= 16 ?

Binary Number

Decimal Number: 12 à Binary Number: ??

2^4 $= 2 * 2 * 2 * 2$ $= 16$	2^3 $= 2 * 2 * 2$ $= 8$	2^2 $= 2 * 2$ $= 4$	2^1 $= 2$	2^0 $= 1$
0				



12 >= 16 ?

Binary Number

Decimal Number: 12 à Binary Number: ??

2^4 = $2 * 2 * 2 * 2$ = 16	2^3 = $2 * 2 * 2$ = 8	2^2 = $2 * 2$ = 4	2^1 = 2	2^0 = 1
0				



12 \geq 8?

Binary Number

Decimal Number: 12 à Binary Number: ??

2^4 $= 2 * 2 * 2 * 2$ $= 16$	2^3 $= 2 * 2 * 2$ $= 8$	2^2 $= 2 * 2$ $= 4$	2^1 $= 2$	2^0 $= 1$
0	1			



12 \geq 8?

Binary Number

Decimal Number: 12 à Binary Number: ??

2^4 $= 2 * 2 * 2 * 2$ $= 16$	2^3 $= 2 * 2 * 2$ $= 8$	2^2 $= 2 * 2$ $= 4$	2^1 $= 2$	2^0 $= 1$
0	1			

$$12 - 8 = 4$$

Binary Number

Decimal Number: 12 à Binary Number: ??

2^4 $= 2 * 2 * 2 * 2$ $= 16$	2^3 $= 2 * 2 * 2$ $= 8$	2^2 $= 2 * 2$ $= 4$	2^1 $= 2$	2^0 $= 1$
0	1			



4 >= 4?

Binary Number

Decimal Number: 12 à Binary Number: ??

2^4 $= 2 * 2 * 2 * 2$ $= 16$	2^3 $= 2 * 2 * 2$ $= 8$	2^2 $= 2 * 2$ $= 4$	2^1 $= 2$	2^0 $= 1$
0	1	1		



4 >= 4?

Binary Number

Decimal Number: 12 à Binary Number: ??

2^4 $= 2 * 2 * 2 * 2$ $= 16$	2^3 $= 2 * 2 * 2$ $= 8$	2^2 $= 2 * 2$ $= 4$	2^1 $= 2$	2^0 $= 1$
0	1	1		

$$4 - 4 = 0$$

Binary Number

Decimal Number: 12 à Binary Number: ??

2^4 $= 2 * 2 * 2 * 2$ $= 16$	2^3 $= 2 * 2 * 2$ $= 8$	2^2 $= 2 * 2$ $= 4$	2^1 $= 2$	2^0 $= 1$
0	1	1		



0 >= 2?

Binary Number

Decimal Number: 12 à Binary Number: ??

2^4 $= 2 * 2 * 2 * 2$ $= 16$	2^3 $= 2 * 2 * 2$ $= 8$	2^2 $= 2 * 2$ $= 4$	2^1 $= 2$	2^0 $= 1$
0	1	1	0	



0 >= 2?

Binary Number

Decimal Number: 12 à Binary Number: ??

2^4 $= 2 * 2 * 2 * 2$ $= 16$	2^3 $= 2 * 2 * 2$ $= 8$	2^2 $= 2 * 2$ $= 4$	2^1 $= 2$	2^0 $= 1$
0	1	1	0	



0 >= 1?

Binary Number

Decimal Number: 12 à Binary Number: ??

2^4 $= 2 * 2 * 2 * 2$ $= 16$	2^3 $= 2 * 2 * 2$ $= 8$	2^2 $= 2 * 2$ $= 4$	2^1 $= 2$	2^0 $= 1$
0	1	1	0	0



0 >= 1?

Binary Number

Decimal Number: 12 à Binary Number: ??

2^4 $= 2 * 2 * 2 * 2$ $= 16$	2^3 $= 2 * 2 * 2$ $= 8$	2^2 $= 2 * 2$ $= 4$	2^1 $= 2$	2^0 $= 1$
0	1	1	0	0

DONE

Binary Number

Decimal Number: 12 à Binary Number: ??

2^4 = $2 * 2 * 2 * 2$ = 16	2^3 = $2 * 2 * 2$ = 8	2^2 = $2 * 2$ = 4	2^1 = 2	2^0 = 1
0	1	1	0	0

12 => 1100

Two's Complement

4-bit

$-1 * 2^3$ $= -1 * 2 * 2 * 2$ $= -8$	2^2 $= 2 * 2$ $= 4$	2^1 $= 2$	2^0 $= 1$
0	1	1	1

7

Two's Complement

4-bit

$-1 * 2^3$ $= -1 * 2 * 2 * 2$ $= -8$	2^2 $= 2 * 2$ $= 4$	2^1 $= 2$	2^0 $= 1$
1	0	0	0

-8

Two's Complement

4-bit

$-1 * 2^3$ $= -1 * 2 * 2 * 2$ $= -8$	2^2 $= 2 * 2$ $= 4$	2^1 $= 2$	2^0 $= 1$
1	1	0	1

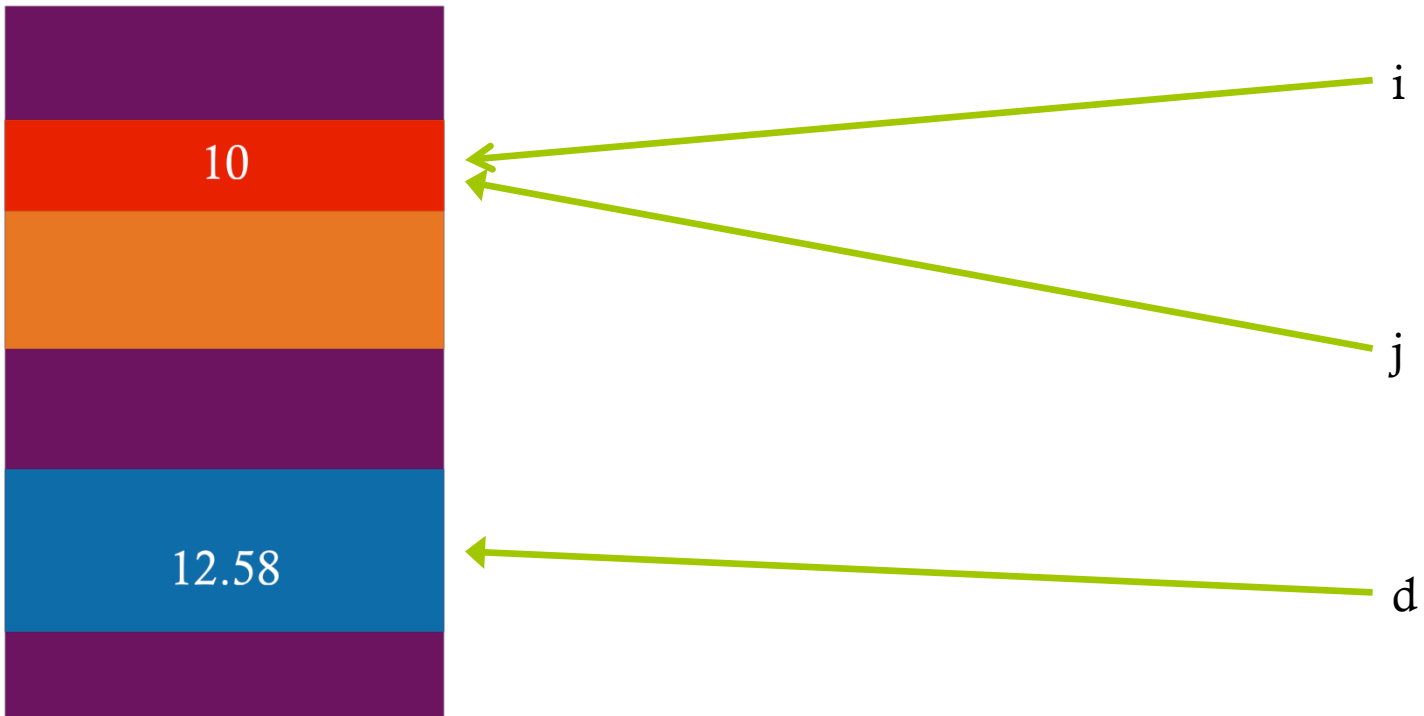
-3

Variable

Variable in computer science is the name you give to computer memory locations which are used to store values in a computer program.

```
int i = 10;  
double d = 12.58;  
boolean isEmpty = false;  
char c = 'q';
```

Variable



Variable

Declaration:

```
type variableName;
```

For example:

```
int number;  
boolean isEmpty;  
String firstName;
```


Variable

Assignment:

```
variableName = value;
```

For example:

```
number = 10;
```

```
isEmpty = true;
```

```
firstName = "Tom";
```

Variable

```
int number = 10;  
boolean isEmpty =  
    true;  
String firstName = "Tom";
```

Variable

```
public static void main(String[] args) {  
    int i = 10;  
    int j;  
    j = 7;  
    int k = i + j;  
    System.out.println(k);  
}
```

Variable

```
public static void main(String[] args) {  
    int i = 10;  
    int j;  
    j = 7;  
    int k = i + j;  
    System.out.println(k);  
}
```

17

Variable

```
public static void main(String[] args){  
    int i = 10;  
    int j;  
    j = 7;  
    int k = i + j;  
    System.out.println(i + " + " + j + " = " + k);  
}
```

Variable

```
public static void main(String[] args) {  
    int i = 10;  
    int j;  
    j = 7;  
    int k = i + j;  
    System.out.println(i + " + " + j + " = " + k);  
}
```

$$10 + 7 = 17$$

Variable

```
public static void main(String[] args)
{
    String s = "horse";
    System.out.println(s);
    System.out.println(s + s);
}
```

Variable

```
public static void main(String[] args) {  
    String s = "horse";  
    System.out.println(s);  
    System.out.println(s + s);  
}
```

horse
horsehorse

Convention

Rule:

1. int with int=> int
2. int with double => double
3. Anything with String => String
4. Expression operation from left to right

Convention

Example:

int with int => int

$$5 / 2 = 2$$

$$10 * 3 = 30$$

Convention

Example:

int with double => double

$$5 / 2.0 = 2.5$$

$$5.0 / 2 = 2.5$$

$$10 * 3.0 = 30.0$$

Convention

Example:

Anything with String => String

$4 + \text{"1"} \Rightarrow \text{"41"}$

$\text{"1"} + 4 \Rightarrow \text{"41"}$

$\text{"1"} + \text{"2"} \Rightarrow \text{"12"}$

Convention

Example:
Left to right

$1/1.0 + 4 + "1" \Rightarrow "5.01"$

$1/1.0 + (4 + "1") \Rightarrow "1.041"$