



@ Lesson 2 –



_____ .

_____ .

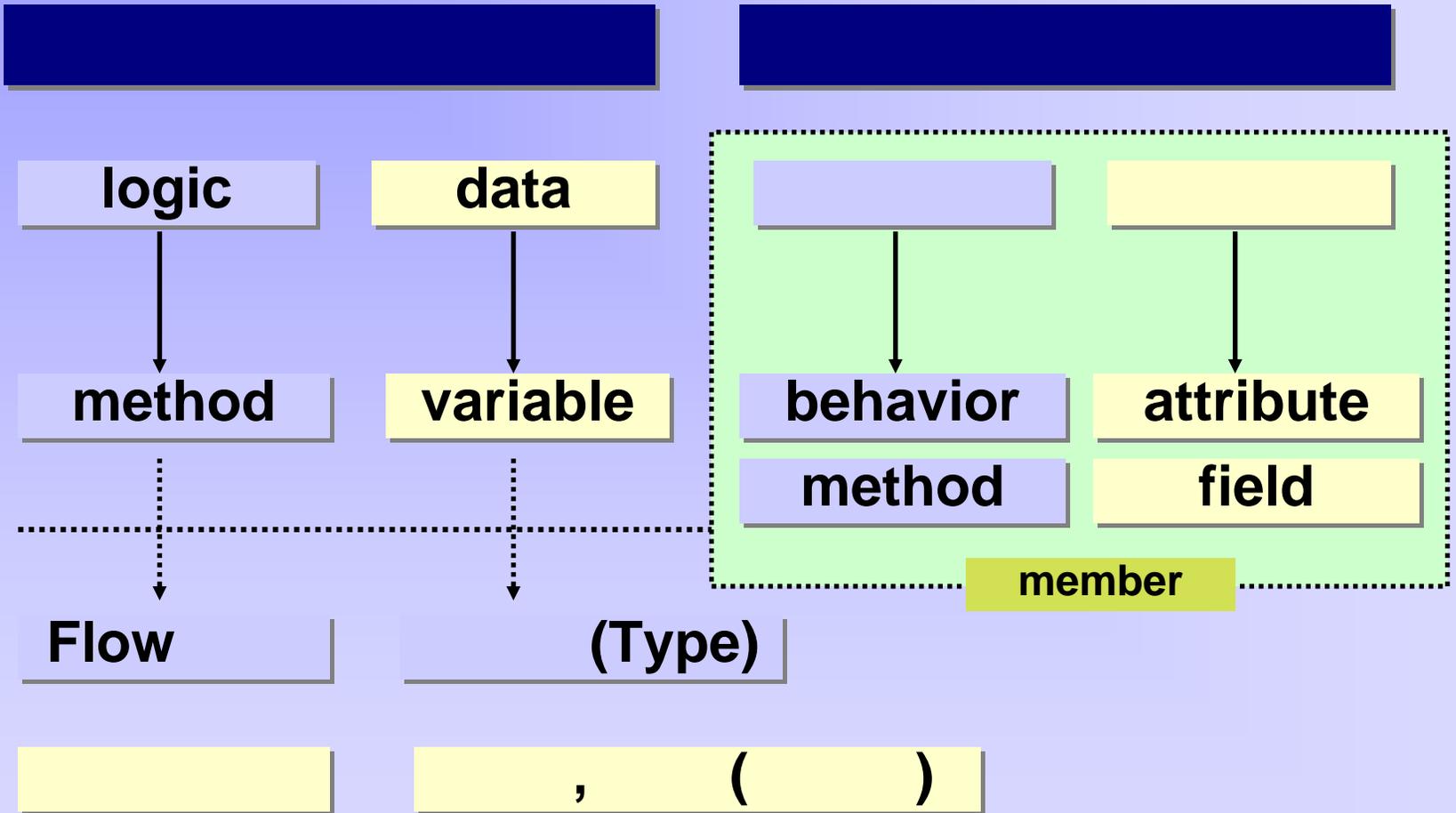
_____ .

() _____ .

() _____ .

@

vs.

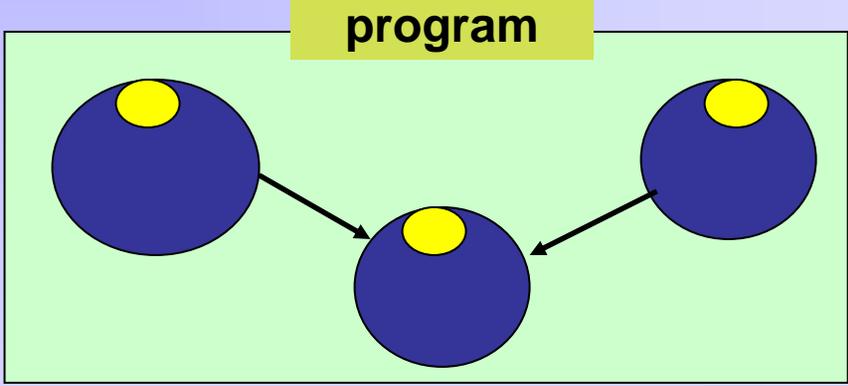
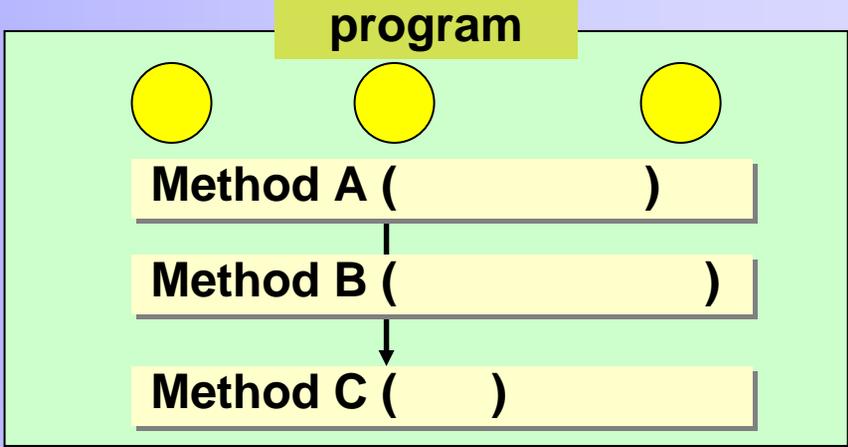


@ ()

: C



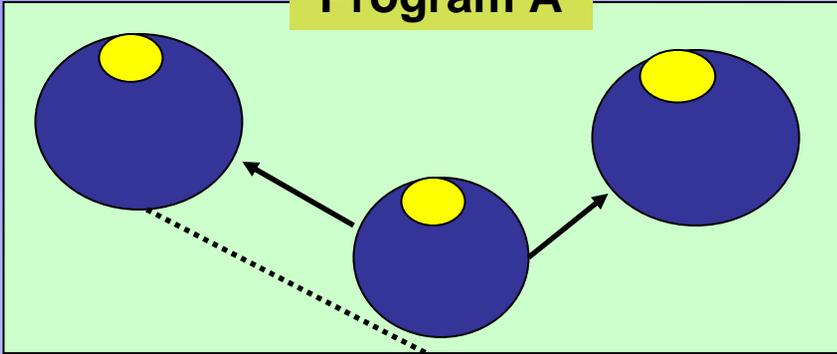
**: Java,
C++, C#**



 **data**

@

Program A

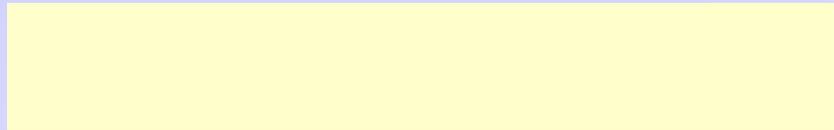
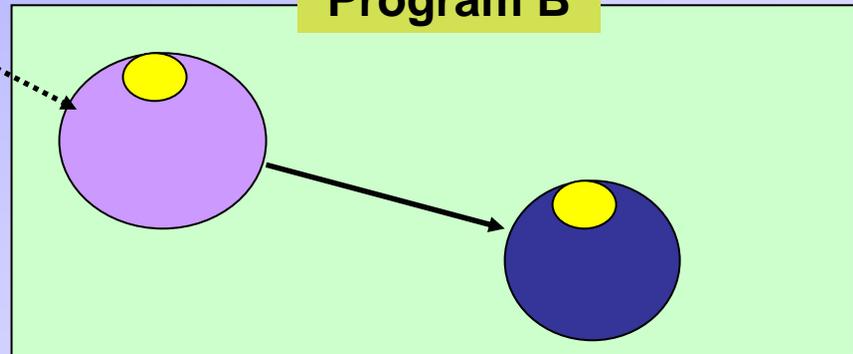


**Loose coupling
(Robust)**

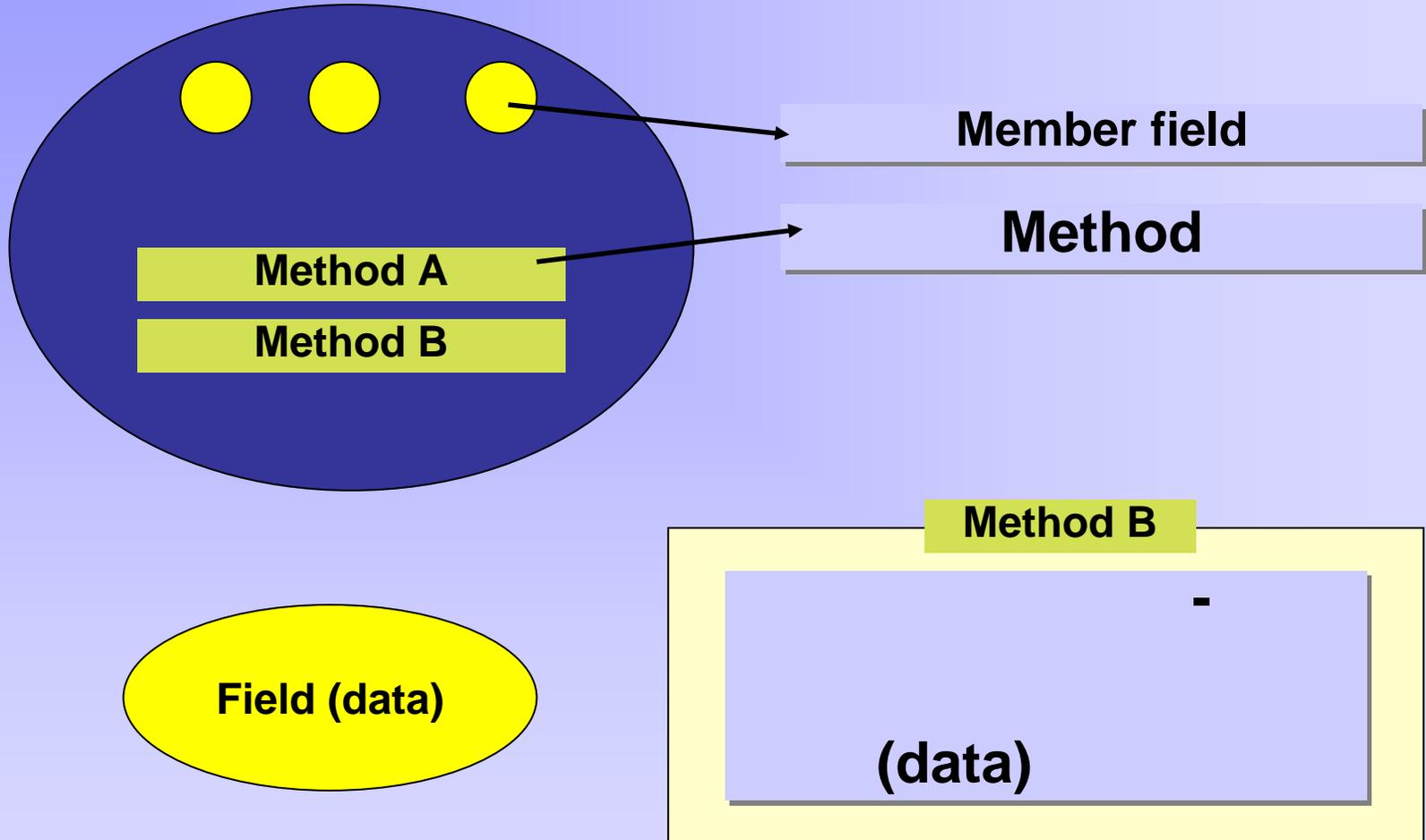
Tight cohesive

(Reusable)

Program B



@



@ Identifiers

[Redacted]

, - Pascal ex) **PersonInSchool**

, - Camel ex) **moneyInMyPocket, goSchool()**

가

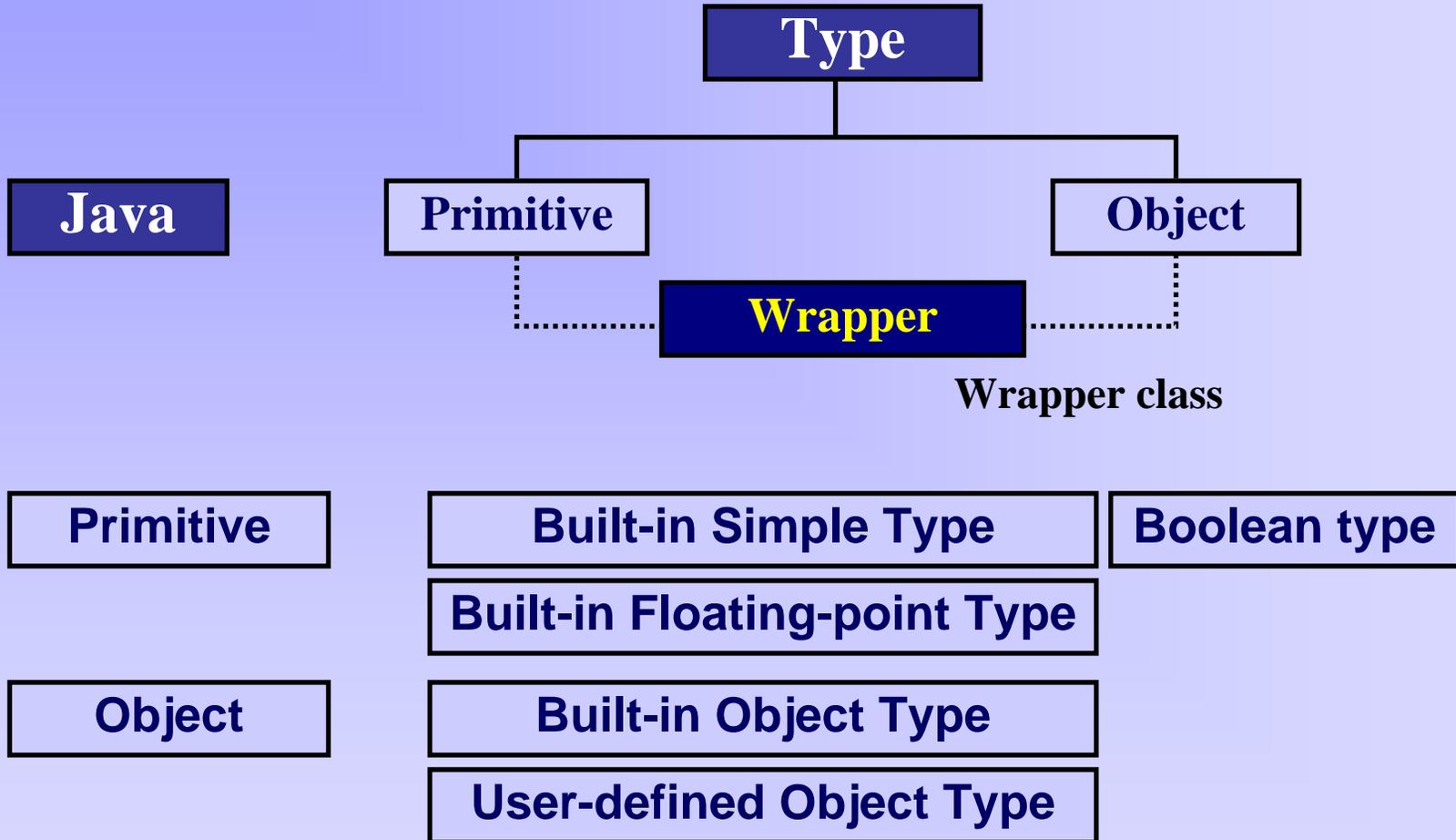
(,keyword 가)

\$_, _ (\$ nested class , _ c)

.(2 가 ,s2)

\$_, _ .

@ Type



@ Keywords

| Keywords | | |
|----------|------------|--------------|
| abstract | float | protected |
| boolean | for | public |
| break | future | strictfp |
| byte | generic | return |
| byvalue | goto | short |
| assert | if | static |
| case | implements | super |
| catch | import | switch |
| char | inner | synchronized |
| class | instanceof | this |
| const | int | throw |
| continue | interface | throws |
| default | long | transient |
| do | native | true |
| double | new | try |
| else | null | var |
| extends | operator | void |
| false | outer | volatile |
| final | package | while |
| finally | private | |

New keywords assert ,Grayed words are reserved but not currently in use.

Note: Grayed words are reserved but not currently in use. New keywords assert

@ keywords in types,

byte

short

int

long

char

float

double

boolean

true

false

if

else

switch

case

default

for

do while

while

continue

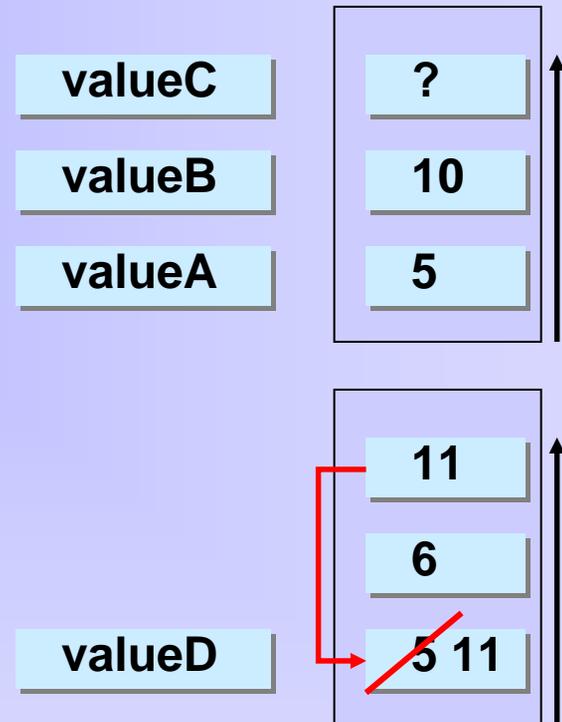
break

@ variable

data (Storage of data)

```
valueA = 5;  
valueB = 10;  
valueC = valueB + valueA;
```

```
valueD = 5;  
valueD = valueD + 6;
```



@ data

가

가

?



(down) cast

double

float

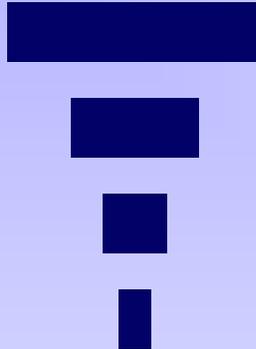
long

int

short, char

byte

promotion



@ data

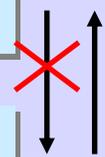
| | | |
|-------------|--------|--|
| | | 320 → 0.32E3, 0.00032 → 0.32E-3 |
| double | 8 byte | + -4.9E-324 ~ 1.8E308 |
| float | 4 byte | + - 1.4E-05 ~ 3.4E038 |
| long | 8 byte | -2 ⁶³ ~ 2 ⁶³ -1 |
| int | 4 byte | -2 ³¹ ~ 2 ³¹ -1 |
| short, char | 2 byte | -2 ¹⁵ ~ 2 ¹⁵ -1, 0 ~ 2 ¹⁶ (65536) |
| byte | 1 byte | -2 ⁷ ~ 2 ⁷ -1 |

2048 1024 512 256 128 64 32 16 8 4 2 1

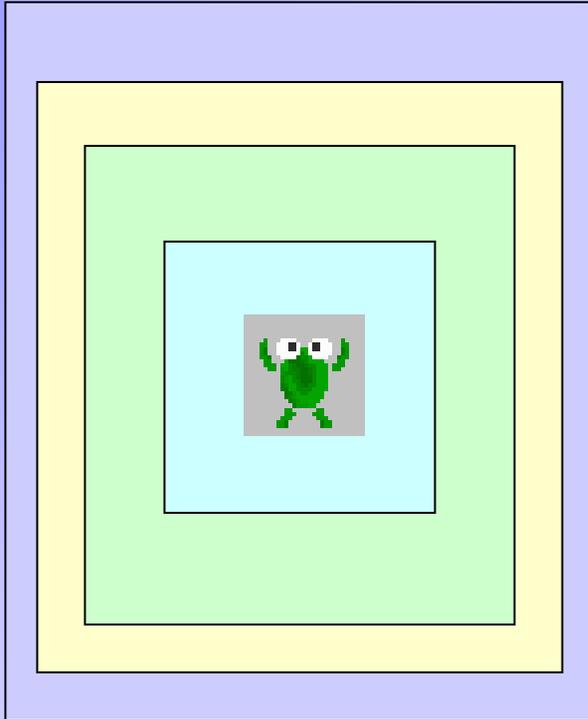
int 123 1 1 1 1 0 1 1

int 123 00000000 00000000 00000000 01111011

short 123 00000000 01111011



@ data ()



```

{ int d = 98;
  { int c = 9;
    { int b = 0;
      { int a = 10;
        }
      }
    }
  }
}

```

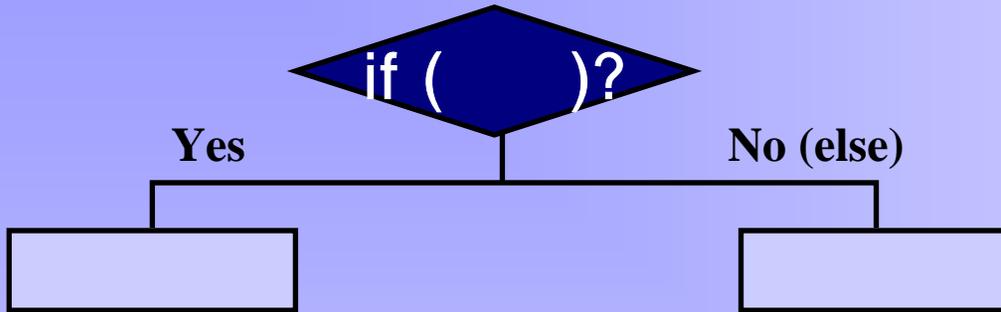
{ }

, { }

, //

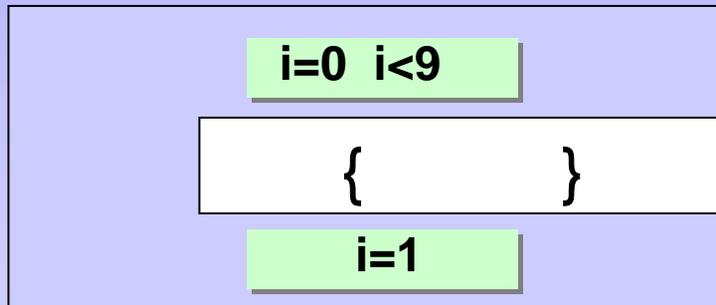
@

if, for



```

if( a%2==0){
    System.out.println(" ");
}else {
    System.out.println(" ");
}
  
```



```

for( int i=0 ; i< 9 ; i++){
}
  
```

```

{ } , .
  
```

`i=0 0<9 → { } → i=1`

`1<9 → { } → i=2 , ...`

`9<9 → end`

@ - int to char

casting

```
public class IntToChar{
    public static void main(String[] args) {
        for(int i=21;i<=130;i++){
            System.out.print( (char)i+" ");
            if(i%10==0){
                System.out.println();
            }
        }
    }
}
```

```
l T | ↑ | → ← L ↔ ▲
▼ ! " # $ % & ' <
> * + , - . / 0 1 2
3 4 5 6 7 8 9 : ; <
= > ? @ A B C D E F
G H I J K L M N O P
Q R S T U U W X Y Z
[ \ ] ^ _ ` a b c d
e f g h i j k l m n
o p q r s t u v w x
y z < | > ~ Δ ? ? ?
```

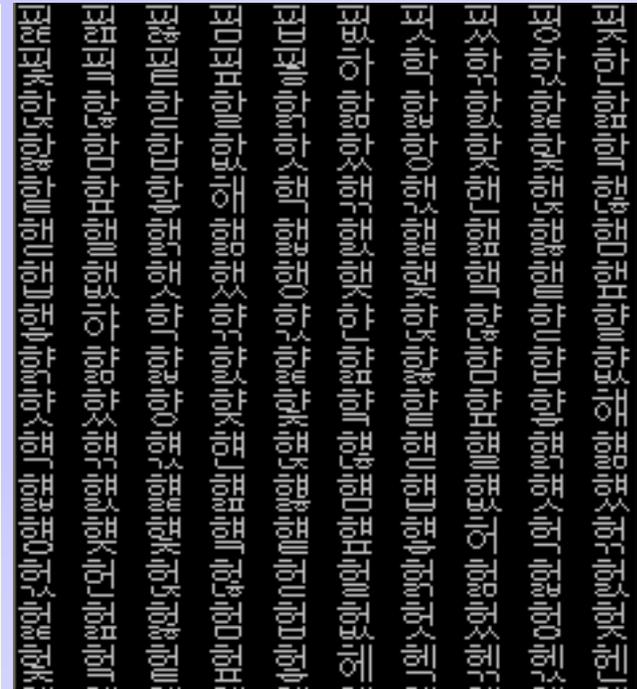
@ Unicode

0 ~ 65536 //

50000

```
char cc=' ';
System.out.println((int)cc); //54620

for(int i=54101;i<=55400;i++){
    System.out.print( (char)i+" ");
        if(i%10==0){
            System.out.println();
        }
}
```



@

→ int , → double

```
byte ba=10;  
byte bb=20;  
byte bc=10+20;  
byte bd=ba+bb;  
  
System.out.println(ba);  
System.out.println(bb);  
System.out.println(bc);  
System.out.println(bd); //error
```

```
int ia=20;  
int ib=30;  
long la=40L;  
long lb=50L;  
float fa=45.0f;  
float fb=46.67F;  
double da=4.5;  
int ic=(int)(la+lb);  
float fc=ia+ib;  
double db=fa+ia;
```

@ Literal

- - - 8 : 057, 030
 - 10 : 57, 30
 - 16 : 0x57, 0x30
 - - : 3.141592, -0.0023
 - : 0.3141592e01, -0.23E-2
 - : true, false

@ Literal

- `String` : `String` .
- `String` (escape) .
 - `'a'`, `':'`
 - `'n'` (= new line)
 - `'r'` (= return)
 - `'t'` (= tab)
 - `""` (= `' u0022'`)
 - `003a'` (= `':'`)
- `String` :
- `"`, `'`
- `" This is Java's string literal", " t tab "`
- `String` : null

@ Literal

```
System.out.println("Hello\nMy name is Kildong Hong.");
//Hello
//My name is Kildong Hong.
//
System.out.println("My Uncles' book");
//My Uncles' book
System.out.println("Hucks\b high.");
//Huck high.
System.out.println("  (\)      4      .");
//      (\)      4      .
System.out.println("      \"      .");
//      "      .
System.out.println(0xFA); //250 16
System.out.println(0222); //146 8
```

@ Comment ()

- ?
,
,
- - one-line comment : //
 - multi-line comment : /* */
 - API comment : API /** */

@ ()

+

-

*

/

%

$45+34 = 79$

$45-34 = 11$

$45*34 = 1530$

$45/34 \rightarrow 1$

$45\%34 \rightarrow 11$

$x = x + 25;$

$x += 25;$

$x = x - 25;$

$x -= 25;$

$x = x * 25;$

$x *= 25;$

$x = x / 25;$

$x /= 25;$

$x = x \% 25;$

$x \% = 25;$

$x = x + 1;$

$x += 1;$

$x ++;$

$++x;$

@ (,)

==

!=

<

>

<=

>=

&

|

^

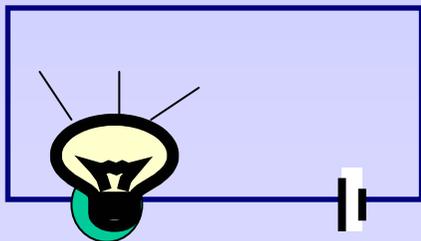
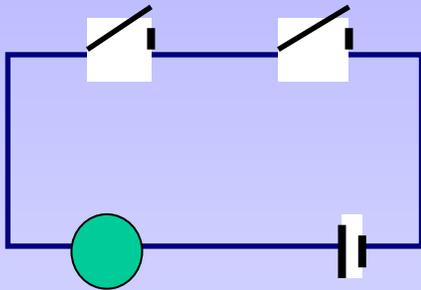
||

&&

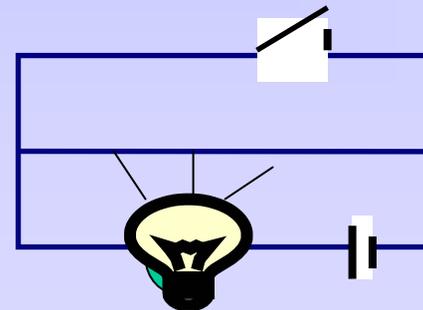
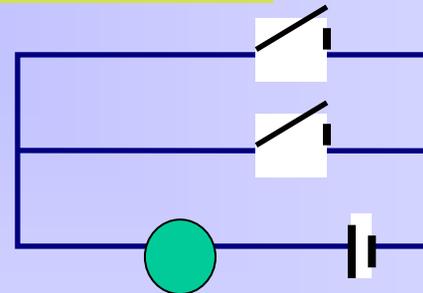
!

||, && → short circuit

& vs. &&



| vs. ||



@ Short-Circuit Test

```
int ca=10, cb=20, cc=-30, cd=15, temp=23;

if((temp>ca)|(temp=='c')){
    System.out.println("    1!!");
}
if((temp>ca)||((temp=='c'))){
    System.out.println("    2!!");
}
if((temp<ca)&(temp=='c')){
    System.out.println("    3!!");
}
if((temp<ca)&&(temp=='c')){
    System.out.println("    4!!");
}
```

@ Logical Operate Test

```
boolean boa1=true, boa2=false, bob1=true, bob2=false;
```

```
System.out.println("P|Q  P&Q  P^Q  !P");
```

```
System.out.print((boa1|boa1)+" "+(boa1&boa1)+" "+(boa1^boa1)+" "+(!boa1)+"\n");
```

```
System.out.print((boa1|boa2)+" "+(boa1&boa2)+" "+(boa1^boa2)+" "+(!boa1)+"\n");
```

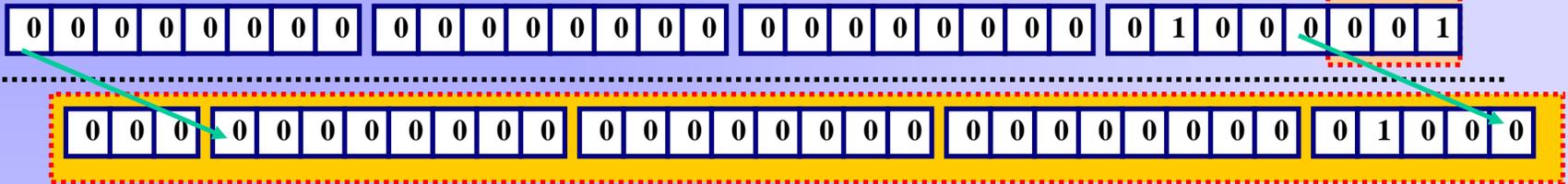
```
System.out.print((boa2|boa1)+" "+(boa2&boa1)+" "+(boa2^boa1)+" "+(!boa2)+"\n");
```

```
System.out.print((boa2|boa2)+" "+(boa2&boa2)+" "+(boa2^boa2)+" "+(!boa2)+"\n");
```


@ bit

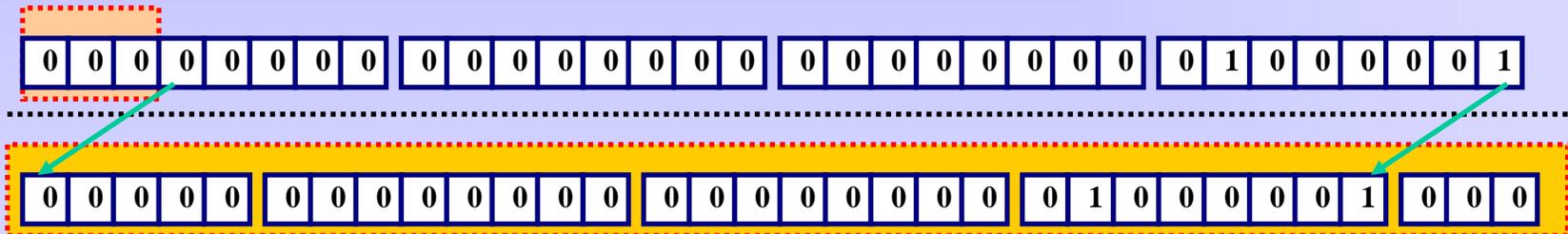
shift >>

$65 \gg 3 = 8$



shift <<

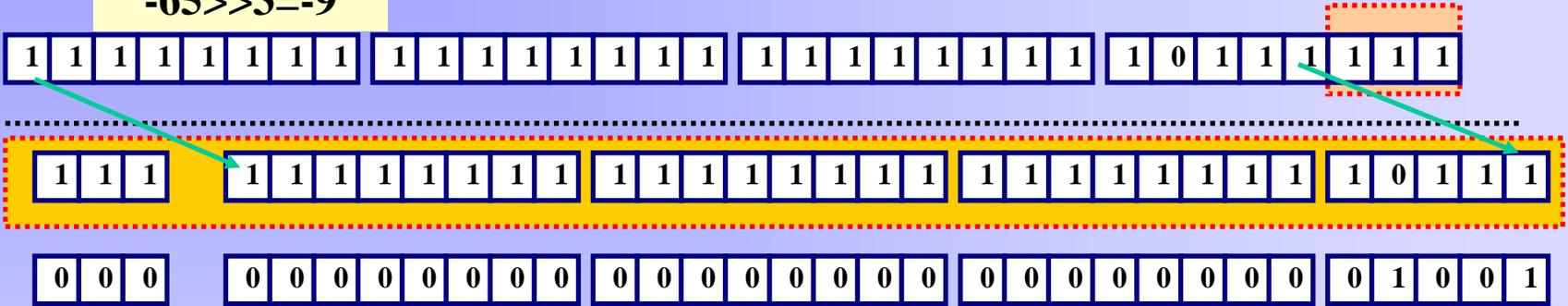
$65 \ll 3 = 520$



@ bit

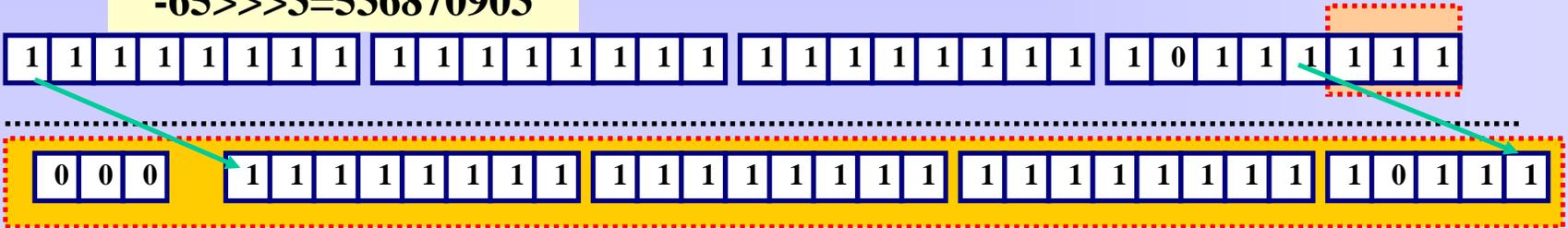
shift >>

$-65 \gg 3 = -9$



shift >>>

$-65 \ggg 3 = 536870903$



@

({}, tab)

Short circuit

@

-

가

(Tab)**space**

.

,

.

.

.

.

.

.

main**(override)**

.

30

.

continue**break**

.

final**private**

.



```
public class DataTypePractice {
    public static void main (String args[]) {
        int intnum;
        float floatnum;
        double doublenum;
        char characterval;
        boolean bool = true; //

        intnum = 120;
        floatnum = 12.23f;
        doublenum = 12.23;
        characterval = 'a';

        System.out.println("Data Practice Program");
        System.out.println("Integer Data Type");
        System.out.println(intnum);
        System.out.println("Float Data Type");
        System.out.println(floatnum);
        System.out.println("Double Data Type");
        System.out.println(doublenum);
        System.out.println("Character Data Type");
        System.out.println(characterval);
        System.out.println("Boolean Data Type");
        System.out.println(bool);
    }
}
```



```
public class OperatorPractice {
    public static void main (String args[]) {
        double dnum;
        int num, assign;

        System.out.println("Operator Practice");
        /*          가          */
        dnum = 12.3 + 34.5 * 12.3;
        /* print   println   가          가          */
        System.out.print("12.3 + 34.5 * 12.3 = ");
        Syetem.out.println(dnum);
        /*          */
        num = 10;
        Syetem.out.println("num = 10");
        assign = num++;
        System.out.print("After assign = num++ : assign = ");
        System.out.println(assign);
        assign = ++num;
        System.out.print("After assign = ++num : assign = ");
        System.out.println(assign);
        /*          */
        System.out.print("10 > 5 : ");
        System.out.println(10 > 5);
        /*          */
        assign = (num > 5) ? 10 : 20; // 3
        System.out.print("(num > 5) ? 10 : 20 = ");
        System.out.println(assign);
    }
}
```

@

-

:: If else

```
if(condition) {
    statement; // condition    (true)        statement
}
```

```
if(condition) {
    statement1; // condition
} else {
    statement2; // condition
}
```

```
if(condition 1) {
    statement1; // condition 1
} else if(condition 2) {
    statement2; // condition 1    , condition 2
} else {
    statement3; // condition 1, 2
}
```

```
public class IfPractice {
    public static void main (String args[]) {
        int num;

        num = 13 / 2;
        if(num == 6) {
            System.out.println("13 / 2 is 6");
        } else {
            System.out.println("13 / 2 = " + num);
        } // + "13 / 2 = " num
    }
}
```

@

-

:: switch case

```
switch( ) {  
    case 1 : 1; break;  
    case 2 : 2; break;  
    ...  
    case n : n; break;  
    default : ;  
}
```

break : 가

```
public class SwitchPractice {  
    public static void main (String args[]) {  
        int num;  
        num = 13 / 2;  
        switch(num) {  
            case 6 : System.out.println("13 / 2 is 6"); break;  
            default : System.out.println("13 / 2 = " + num); break;  
        }  
    }  
}
```

@

-

:: while

```
while( ) {  
    ; // , while  
}
```

```
public class WhilePractice {  
    public static void main (String args[]) {  
        int i;  
        i = 10;  
        System.out.println("Countdown start!");  
        /* 가 10 1 i */  
        while(i > 0) {  
            System.out.println(i);  
            i--;  
        }  
    }  
}
```

@

-

:: do while

```
do {  
    ;  
}while( );
```

```
public class DoWhilePractice {  
    public static void main (String args[]) {  
        int i;  
        i = 10;  
        System.out.println("Countdown start!");  
        do {  
            System.out.println(i);  
            i--;  
        }while(i > 0);  
    }  
}
```

@

-

:: for

```
for( ; ; ) {  
    ;  
}
```

```
public class ForPractice {  
    public static void main (String args[]) {  
        int i;  
        System.out.println("Countdown start");  
        for(i = 10; i > 0; i--) {  
            System.out.println(i);  
        }  
    }  
}
```

@ - :: break, continue, return

for , while , do while Loop switch case .

```
public class BreakPractice {
    public static void main (String args[]) {
        int i;
        i = 10;
        System.out.println("Countdown start!");
        while(true) {
            if(i == 0) break;
            System.out.println(i);
            i--;
        }
    }
}
```

@ - :: break, continue, return

for , while , do while .

```

public class ContinuePractice {
    public static void main (String args[]) {
        int i;
        i = 20;
        System.out.println("Countdown start!");
        while(true) {
            i - -;
            if(i > 10) continue; // i 가 10 while
            if(i == 0) break; // i 가 0 while
            System.out.println(i);
        }
    }
}

```

@ - :: break, continue, return

, return int , void . int
return

```
public class ReturnPractice {
    public static void main (String args[]) {
        int i;
        i = 10;
        System.out.println("Countdown start!");
        while(true) {
            if(i == 0) return;
            System.out.println(i);
            i--;
        }
    }
}
```

@

-

가

) 1

100

,

.