

CSE2016

프로그램설계방법론

식, 변수, 타입

Expression, Variable, Type

도경구

한양대학교 ERICA 소프트웨어학부



정수

Integer

byte

8 bit

short

16 bit

int

32 bit

long

64 bit

..., -3, -2, -1, 0, 1, 2, 3, ...

연산자

+	더하기
-	빼기
*	곱하기
/	나누기
%	나머지

사례

미화 동전의 총액 계산

동전의 종류



penny
1 cent



nickel
5 cent



dime
10 cent



quarter
25 cent

100 cent = 1 dollar (\$)

사례

미화 동전의 총액 계산

동전의 종류

penny
1 cent



nickel
5 cent



dime
10 cent



quarter
25 cent

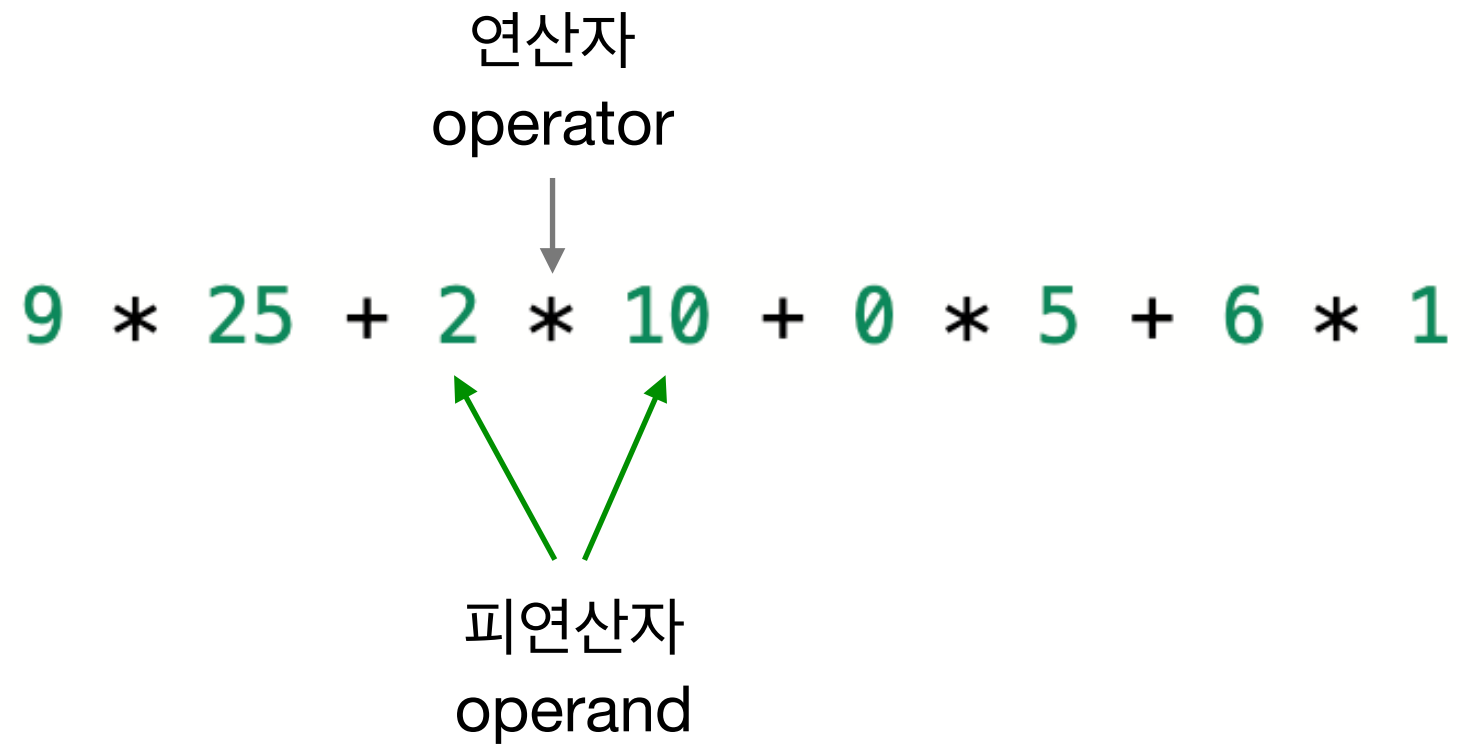
100 cent = 1 dollar (\$)



$$\begin{array}{r}
 9 \times 25 = 225 \text{ cent} \\
 2 \times 10 = 20 \text{ cent} \\
 0 \times 5 = 0 \text{ cent} \\
 6 \times 1 = 6 \text{ cent} \\
 \hline
 251 \text{ cent}
 \end{array}
 +$$

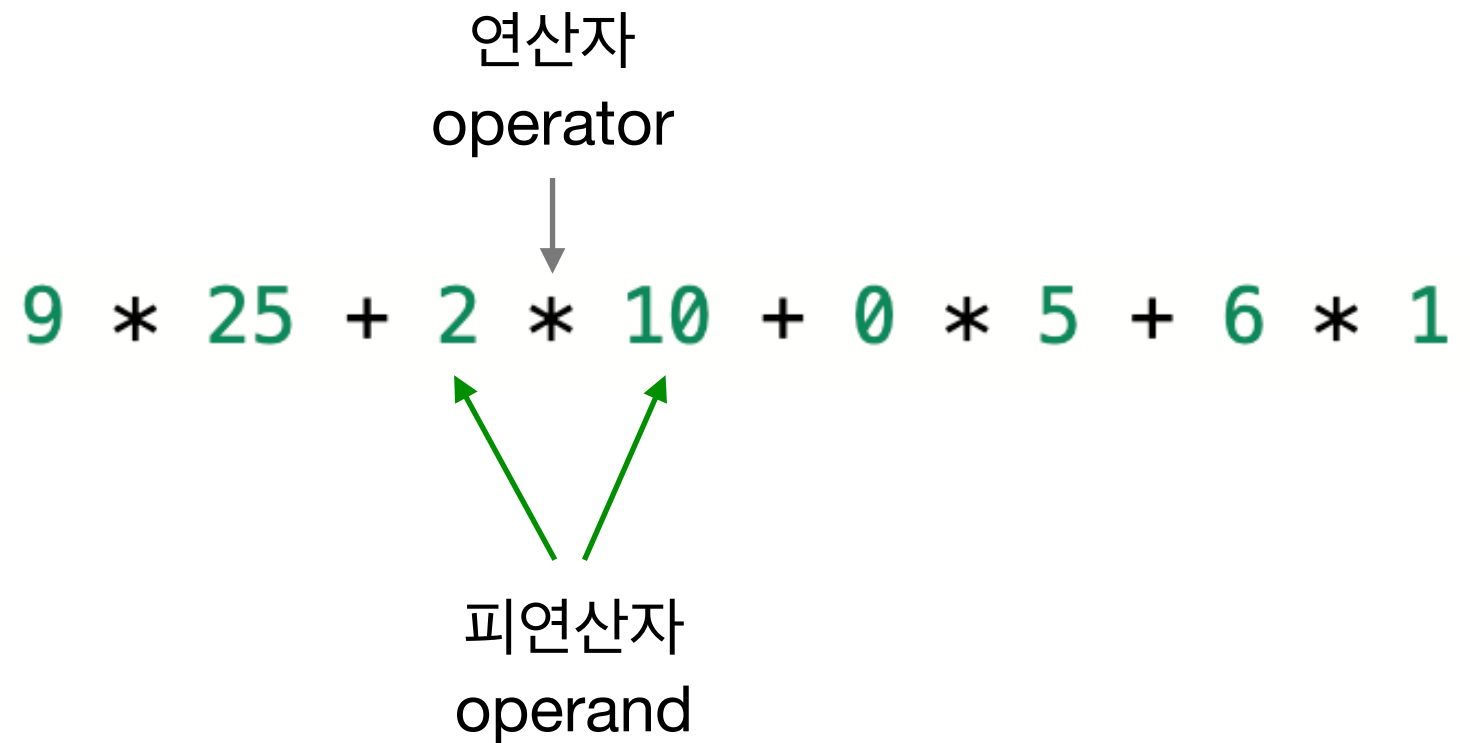
Expression

$$\begin{array}{r} 9 \times 25 = 225 \text{ cent} \\ 2 \times 10 = 20 \text{ cent} \\ 0 \times 5 = 0 \text{ cent} \\ 6 \times 1 = 6 \text{ cent} \\ \hline 251 \text{ cent} \end{array} +$$



Java Expression

$$\begin{array}{r}
 9 \times 25 = 225 \text{ cent} \\
 2 \times 10 = 20 \text{ cent} \\
 0 \times 5 = 0 \text{ cent} \\
 6 \times 1 = 6 \text{ cent} \\
 \hline
 251 \text{ cent}
 \end{array}
 +$$



```

1 public class CoinCalculator {
2     public static void main(String[] args) {
3         System.out.println(9 * 25 + 2 * 10 + 0 * 5 + 6 * 1);
4     }
5 }

```

```
1 public class CoinCalculator {  
2     public static void main(String[] args) {  
3         System.out.println(9 * 25 + 2 * 10 + 0 * 5 + 6 * 1);  
4     }  
5 }
```



```
1 public class CoinCalculator {  
2   public static void main(String[] args) {  
3       System.out.println(9 * 25 + 2 * 10 + 0 * 5 + 6 * 1);  
4   }  
5 }
```



Nickel 하나 추가


```
1 public class CoinCalculator {  
2     public static void main(String[] args) {  
3         System.out.println(9 * 25 + 2 * 10 + 0 * 5 + 6 * 1);  
4     }  
5 }
```

1로 변경



Neckel 하나 추가

```
1 public class CoinCalculator {  
2   public static void main(String[] args) {  
3     System.out.println(9 * 25 + 2 * 10 + 0 * 5 + 6 * 1);  
4   }  
5 }
```

변하는가? vs. 변하지 않는가?

```
1 public class CoinCalculator {  
2   public static void main(String[] args) {  
3     System.out.println(9 * 25 + 2 * 10 + 0 * 5 + 6 * 1);  
4   }  
5 }
```

변하는가? vs. 변하지 않는가?

↓
변수

↓
상수

변수 Variables

변수

Variable

변수 선언
variable declaration

```
int quarters;
```

quarters $\xrightarrow{\text{int}}$

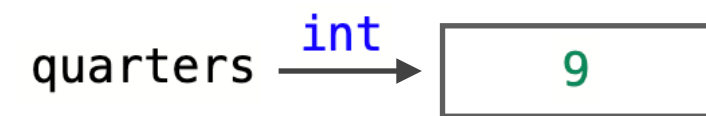
변수

Variable

변수 선언

variable declaration

```
int quarters;
```



변수 초기값 저장

variable initialization

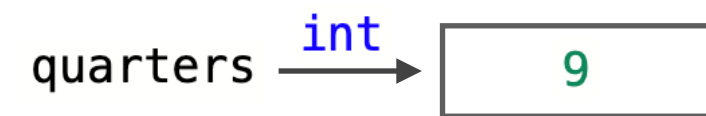
```
quarters = 9;
```

변수

Variable

변수 선언
variable declaration

```
int quarters;
```

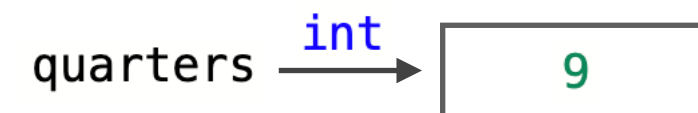


변수 초기값 저장
variable assignment

```
quarters = 9;
```

변수 선언과 값 저장

```
int quarters = 9;
```

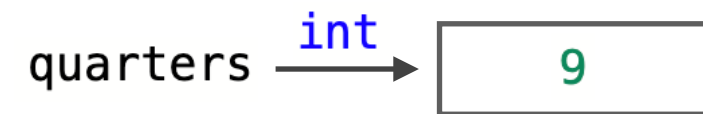


변수

Variable

변수 선언
variable declaration

```
int quarters;
```

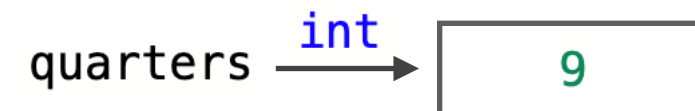


변수 초기값 저장
variable initialization

```
quarters = 9;
```

변수 선언과 값 저장

```
int quarters = 9;
```



변수 사용

```
System.out.print(quarters * 25);
```


변수 이름 작명 규칙

영문 알파벳 대소문자	a-z A-Z	
숫자	0-9	맨앞에 올 수 없음
밑줄	_	
달러 기호	\$	(당분간) 비추천

quarters

myQuarters ← 일반 변수 이름으로 대문자 사용은 추천하지 않음

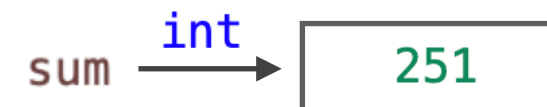
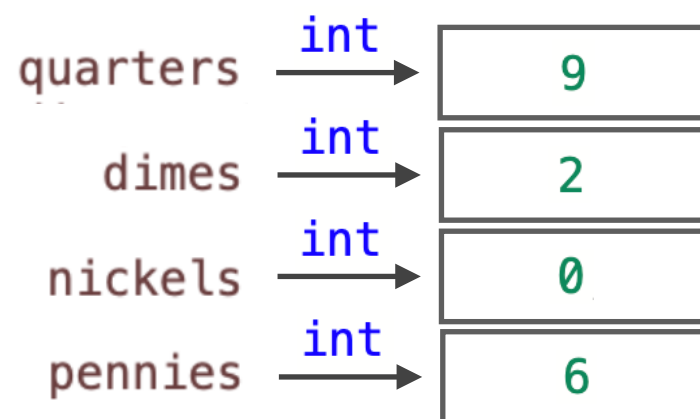
my_quarters

erica55

```
1 public class CoinCalculator {  
2     public static void main(String[] args) {  
3         System.out.println(9 * 25 + 2 * 10 + 0 * 5 + 6 * 1);  
4     }  
5 }
```



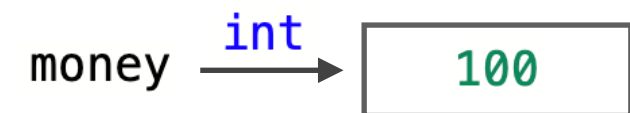
```
1 public class CoinCalculator {  
2     public static void main(String[] args) {  
3         int quarters = 9;  
4         int dimes = 2;  
5         int nickels = 0;  
6         int pennies = 6;  
7         int sum = quarters * 25 + dimes * 10 + nickels * 5 + pennies * 1;  
8         System.out.println(sum);  
9     }  
10 }
```



지정

Assignment

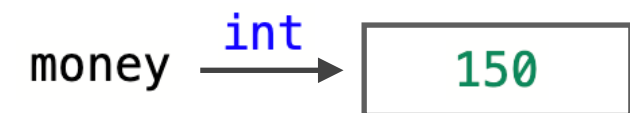
```
1 public class CoinCalculator {  
2     public static void main(String[] args) {  
3         int money = 100;  
4         System.out.println(money);  
5         money = money + 50;  
6         System.out.println(money);  
7     }  
8 }
```



지정

Assignment

```
1 public class CoinCalculator {  
2     public static void main(String[] args) {  
3         int money = 100;  
4         System.out.println(money);  
5         money = money + 50;  
6         System.out.println(money);  
7     }  
8 }
```



실수

Real Number

float

32 bit

double

64 bit

3.141592

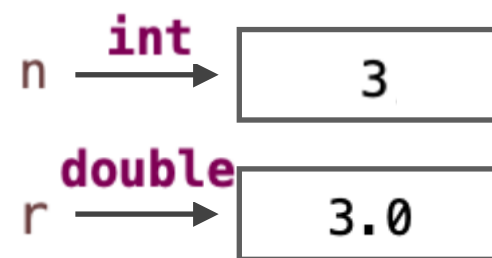
연산자

+	더하기
-	빼기
*	곱하기
/	나누기
%	나머지

```
1 public class MathDemo {  
2     public static void main(String[] args) {  
3         double r = 3.14;  
4         System.out.println(r);  
5     }  
6 }  
7 }
```

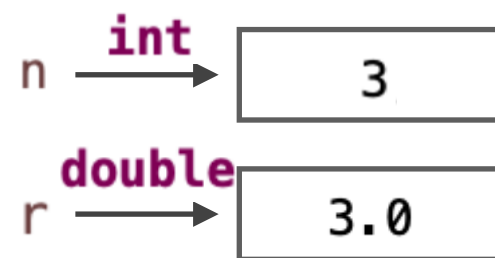
double
r → 3.14

```
1 public class MathDemo {  
2     public static void main(String[] args) {  
3         int n = 3;  
4         double r = n;  
5         System.out.println(r);  
6     }  
7 }  
8 }
```



묵시적 타입 변환
implicit type casting

```
1 public class MathDemo {  
2     public static void main(String[] args) {  
3         int n = 3;  
4         double r = n;  
5         n = r + 1;  
6         System.out.println(n);  
7     }  
8 }  
9 }
```



컴파일 오류
타입 불일치


```
1 public class MathDemo {  
2     public static void main(String[] args) {  
3         int n = 3;  
4         double r = n;  
5         n = (int)(r + 1);  
6         System.out.println(n);  
7     }  
8 }  
9 }
```

명시적 타입 변환
explicit type casting

```
1 public class MathDemo {  
2     public static void main(String[] args) {  
3         double r = 3.14;  
4         System.out.println(Math.sqrt(r));  
5         System.out.println(Math.pow(r,2));  
6         System.out.println(Math.abs(-r));  
7     }  
8 }  
9 }
```

Math 라이브러리



논리값

boolean

1 bit

true

false

연산자

! 논리역 not

&& 논리곱 and

|| 논리합 or

```
1 public class Demo {  
2     public static void main(String[] args) {  
3         boolean b = false;  
4         b = (b && true) || false;  
5         System.out.println(!b);  
6     }  
7 }  
8 }
```



```
1 public class Demo {  
2     public static void main(String[] args) {  
3         boolean b = false;  
4         b = (b && true) || false;  
5         System.out.println(!b);  
6     }  
7 }  
8 }
```



```
1 public class Demo {  
2     public static void main(String[] args) {  
3         boolean b = false;  
4         b = (b && true) || false;  
5         System.out.println(!b);  
6     }  
7 }  
8 }
```



비교 연산자

연산	연산자	사례
크다	<code>></code>	<code>3.14 > 3 + 0.1</code>
작다	<code><</code>	<code>3.14 < 3 + 0.1</code>
작거나 같다	<code><=</code>	<code>3.14 * (-1) <= 0</code>
크거나 같다	<code>>=</code>	<code>1 >= 2</code>
같다	<code>==</code>	<code>1 + 2 == 2 + 1</code>
다르다	<code>!=</code>	<code>3.14 != 3.141592</code>

연산자의 우선순위, 결합순서

Precedence Associativity

우선 순위	연산자	비고	결합 순서
가장 높음	-	음수 기호, -3	우결합
높음	* / %		좌결합
가운데	+ -		
낮음	< <= > >=		
가장 낮음	== !=		

---1
=
-(-(-1))

1 - 2 - 3
=
(1 - 2) - 3

문자열

String

"What a wonderful world!"

객체!!!

`java.lang.String`

```
1 public class Demo {  
2     public static void main(String[] args) {  
3         String slogan = "자유는 공짜가 아니다";  
4         System.out.println(slogan);  
5         System.out.println(slogan.length());  
6         String slogan2 = "    " + slogan + "    ";  
7         System.out.println(slogan2 + "!");  
8         System.out.println(slogan2.length());  
9         String slogan3 = slogan2.trim();  
10        System.out.println(slogan3 + "!");  
11        System.out.println(slogan);  
12        System.out.println(slogan3);  
13        System.out.println(slogan == slogan3);  
14        System.out.println(slogan.equals(slogan3));  
15        System.out.println(slogan.charAt(5));  
16        System.out.println(slogan.substring(4,6));  
17        String slogan_E = "Freedom is not free";  
18        System.out.println(slogan_E.toUpperCase());  
19        System.out.println(slogan_E.toLowerCase());  
20        System.out.println(slogan_E.indexOf("o"));  
21        System.out.println(slogan_E.indexOf("o",6));  
22        System.out.println("a".compareTo("c"));  
23        System.out.println("C".compareTo("A"));  
24    }  
25 }
```

문자

Characters

char

16 bit

```
1 public class Demo {  
2     public static void main(String[] args) {  
3         char c = 'A';  
4         char n = '0';  
5         char s = '#';  
6         char backspace = '\b';  
7         char tab = '\t';  
8         char newline = '\n';  
9         char return_character = '\r';  
10        char double_quote = '\"';  
11        char single_quote = '\'';  
12        char backslash = '\\';  
13        char hangul = '한';  
14        System.out.println(hangul);  
15        System.out.println((int)hangul);  
16        System.out.println((char)(hangul+1));  
17    }  
18 }
```

← Unicode

사례

미화 동전의 총액 계산

총액 표시를 다음과 같은 형식으로 달러로 해주세요!

\$2.51

힌트

$$251 / 100 = 2$$

$$251 \% 100 = 51$$

사례

미화 동전의 총액 계산

```
1 public class CoinCalculator {  
2     public static void main(String[] args) {  
3         int quarters = 9;  
4         int dimes = 2;  
5         int nickels = 0;  
6         int pennies = 6;  
7         int sum = quarters * 25 + dimes * 10 + nickels * 5 + pennies * 1;  
8         int dollar = sum / 100;  
9         int cent = sum % 100;  
10        System.out.println("$" + dollar + "." + cent);  
11    }  
12 }
```

사례

미화 동전의 총액 계산

```
1 public class CoinCalculator {  
2     public static void main(String[] args) {  
3         int quarters = 9;  
4         int dimes = 2;  
5         int nickels = 0;  
6         int pennies = 6;  
7         int sum = quarters * 25 + dimes * 10 + nickels * 5 + pennies * 1;  
8         int dollar = sum / 100;  
9         int cent = sum % 100;  
10        System.out.println("$" + dollar + "." + cent);  
11    }  
12 }
```

정수 더하기

문자열 이어 붙이기

+ 연산자의 중복 사용

overloading

사례

미화 동전의 총액 계산

```
1 public class CoinCalculator {
2   public static void main(String[] args) {
3       int quarters = 9;
4       int dimes = 2;
5       int nickels = 0;
6       int pennies = 6;
7       int sum = quarters * 25 + dimes * 10 + nickels * 5 + pennies * 1;
8       int dollar = sum / 100;
9       int cent = sum % 100;
10      System.out.println("$" + dollar + "." + cent);
11  }
12 }
```

```
1 public class CoinCalculator {
2   public static void main(String[] args) {
3       int quarters = 9;
4       int dimes = 2;
5       int nickels = 0;
6       int pennies = 6;
7       int sum = quarters * 25 + dimes * 10 + nickels * 5 + pennies * 1;
8       System.out.println("$" + (sum / 100) + "." + (sum % 100));
9   }
10 }
```

타입 검사

Type Checking

int

double

boolean

char

기본 타입

primitive type

String

...

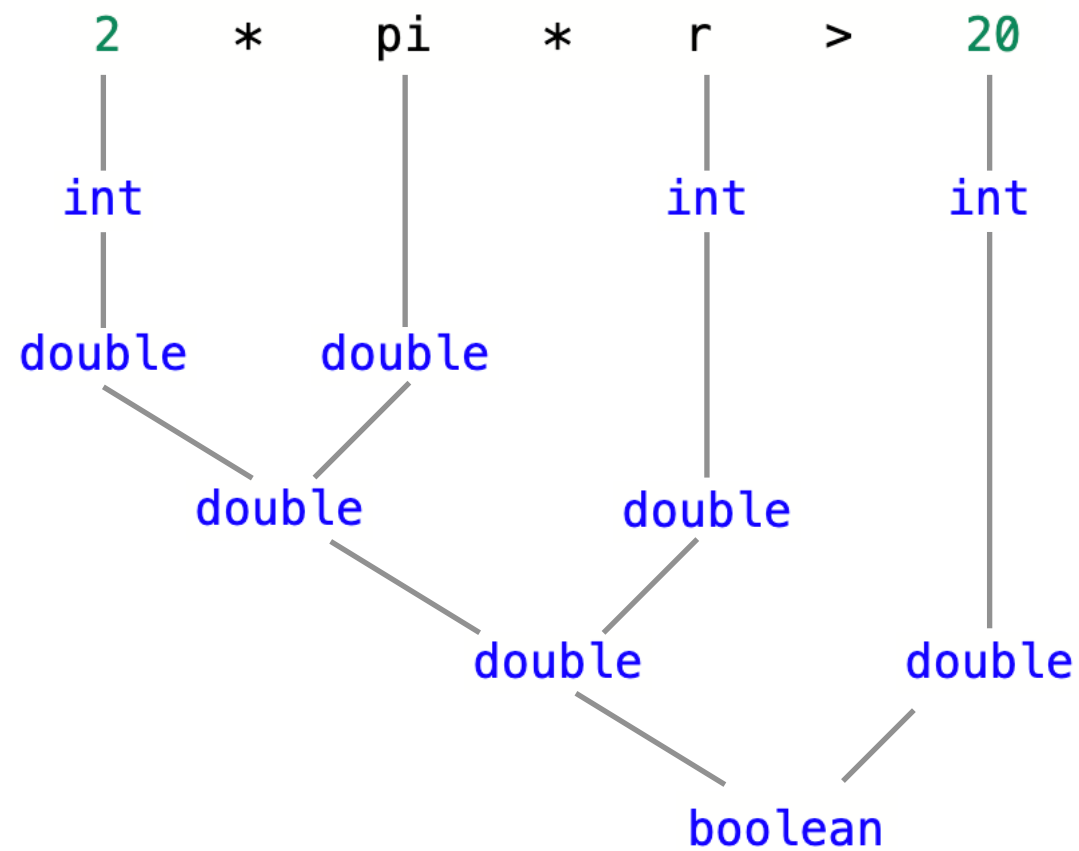
객체 타입

reference type

object type


```
1 public class Demo {  
2     public static void main(String[] args) {  
3         int r = 3;  
4         double pi = 3.14;  
5         System.out.println(2 * pi * r > 20);  
6     }  
7 }
```

타입 검사



```
1 public class Demo {  
2     public static void main(String[] args) {  
3         int i = 1 + 2.3;  
4         int x = "Hello!";  
5         boolean b = true;  
6         System.out.println(b + 5);  
7         System.out.println(Math.println(";-(");  
8     }  
9 }
```

컴파일러의 오류 메시지

Exception in thread "main" java.lang.Error: Unresolved compilation problems:
Type mismatch: cannot convert from double to int
Type mismatch: cannot convert from String to int
The operator + is undefined for the argument type(s) boolean, int
The method println(String) is undefined for the type Math

at Demo.main([Demo.java:3](#))

오류 Errors

- 컴파일 오류 compile-time error
 - 파싱 오류 parsing error
 - 타입 오류 type error
 - 선언 오류 declaration error
- 실행 오류 run-time error
 - 예외발생 오류 exception error
 - 논리 오류 logic error

```
1  class Main {
2      public static void main(String[] args) {
3          // compile-time errors
4          // parsing errors
5          System.out.println((1 + 2( * 3);
6          // type errors
7          System.out.println(3 + true);
8          int i = true;
9          System.out.println(i * 2);
10         // declaration errors
11         int n = 3;
12         double n = 3.14;
13         System.out.println(n);
14
15         // run-time errors
16         // exception errors
17         int d = Integer.parseInt(args[0]);
18         System.out.println(1 / d);
19         // logic errors
20         int x = 3;
21         int y = 7;
22         System.out.println(x = y);
23     }
24 }
```

사례 #2 - 거스름돈 문제

거스름 돈이 \$3.46인데, 동전의 개수가 최소가 되려면 동전 별로 몇 개씩?



13



2



1



몫

나머지

$$346 - (13 \times 25) = 21$$

$$21 - (2 \times 10) = 1$$

$$1 - (0 \times 5) = 1$$

$$1 - (1 \times 1) = 0$$

알고리즘

1. 거슬러 줄 액수를 센트 단위로 환산하여 money 라고 한다.
2. 거슬러 줄 수 있는 quarter 동전을 최대한 많이 꺼내고, 해당 금액을 money에서 빼고 꺼낸 quarter 동전의 개수를 프린트 한다.
3. 거슬러 줄 수 있는 dime 동전을 최대한 많이 꺼내고, 해당 금액을 money에서 빼고 꺼낸 dime 동전의 개수를 프린트 한다.
4. 거슬러 줄 수 있는 nickel 동전을 최대한 많이 꺼내고, 해당 금액을 money에서 빼고 꺼낸 nickel 동전의 개수를 프린트 한다.
5. money 값이 penny 동전의 개수이므로, money 값을 프린트 한다.

구현

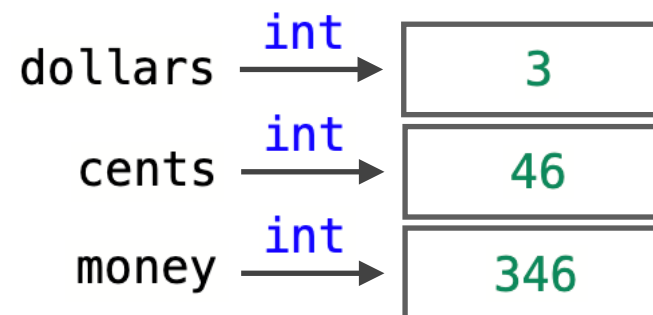
거스름돈 문제

동전의 개수가 최소가 되어야 함

```

1 public class Change {
2     public static void main(String[] args) {
3         int dollars = 3;
4         int cents = 46;
5         int money = dollars * 100 + cents;
6         System.out.println("quarters = " + (money / 25));
7         money = money % 25;
8         System.out.println("dimes = " + (money / 10));
9         money = money % 10;
10        System.out.println("nickels = " + (money / 5));
11        money = money % 5;
12        System.out.println("pennies = " + money);
13    }
14 }

```

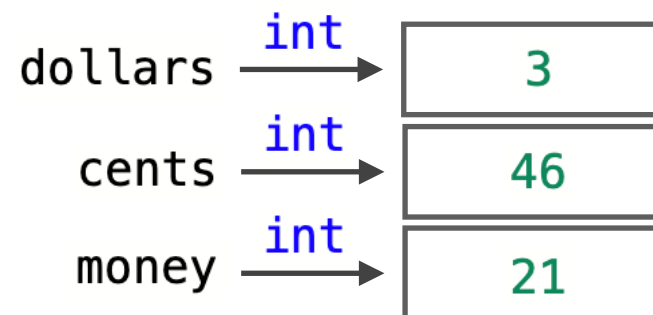


구현

거스름돈 문제

동전의 개수가 최소가 되어야 함

```
1 public class Change {  
2     public static void main(String[] args) {  
3         int dollars = 3;  
4         int cents = 46;  
5         int money = dollars * 100 + cents;  
6         System.out.println("quarters = " + (money / 25));  
7         money = money % 25;  
8         System.out.println("dimes = " + (money / 10));  
9         money = money % 10;  
10        System.out.println("nickels = " + (money / 5));  
11        money = money % 5;  
12        System.out.println("pennies = " + money);  
13    }  
14 }
```



구현

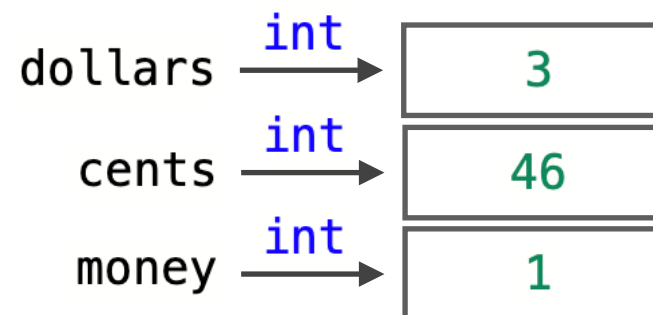
거스름돈 문제

동전의 개수가 최소가 되어야 함

```

1 public class Change {
2     public static void main(String[] args) {
3         int dollars = 3;
4         int cents = 46;
5         int money = dollars * 100 + cents;
6         System.out.println("quarters = " + (money / 25));
7         money = money % 25;
8         System.out.println("dimes = " + (money / 10));
9         money = money % 10;
10        System.out.println("nickels = " + (money / 5));
11        money = money % 5;
12        System.out.println("pennies = " + money);
13    }
14 }

```

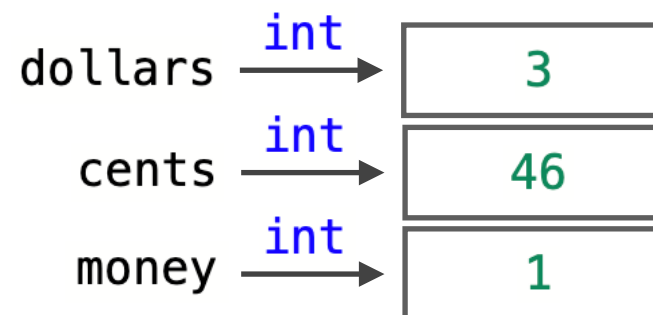


구현

거스름돈 문제

동전의 개수가 최소가 되어야 함

```
1 public class Change {  
2     public static void main(String[] args) {  
3         int dollars = 3;  
4         int cents = 46;  
5         int money = dollars * 100 + cents;  
6         System.out.println("quarters = " + (money / 25));  
7         money = money % 25;  
8         System.out.println("dimes = " + (money / 10));  
9         money = money % 10;  
10        System.out.println("nickels = " + (money / 5));  
11        money = money % 5;  
12        System.out.println("pennies = " + money);  
13    }  
14 }
```



사용자 입력

Input

프로그램 인수로 전달하는 입력

Input via Program Arguments

```
1 public class Demo {  
2     public static void main(String[] args) {  
3         String who = args[0];  
4         String whom = args[1];  
5         System.out.println(who + "은(는) " + whom + "을(를) 좋아합니다.");  
6     }  
7 }
```

```
1 public class Demo {  
2   public static void main(String[] args) {  
3       String who = args[0];  
4       String whom = args[1];  
5       System.out.println(who + "은(는) " + whom + "을(를) 좋아합니다.");  
6   }  
7 }
```

```
1 public class Change {  
2   public static void main(String[] args) {  
3       int dollars = Integer.parseInt(args[0]);  
4       int cents = Integer.parseInt(args[1]);  
5       int money = dollars * 100 + cents;  
6       System.out.println("quarters = " + (money / 25));  
7       money = money % 25;  
8       System.out.println("dimes = " + (money / 10));  
9       money = money % 10;  
10      System.out.println("nickels = " + (money / 5));  
11      money = money % 5;  
12      System.out.println("pennies = " + money);  
13  }  
14 }
```

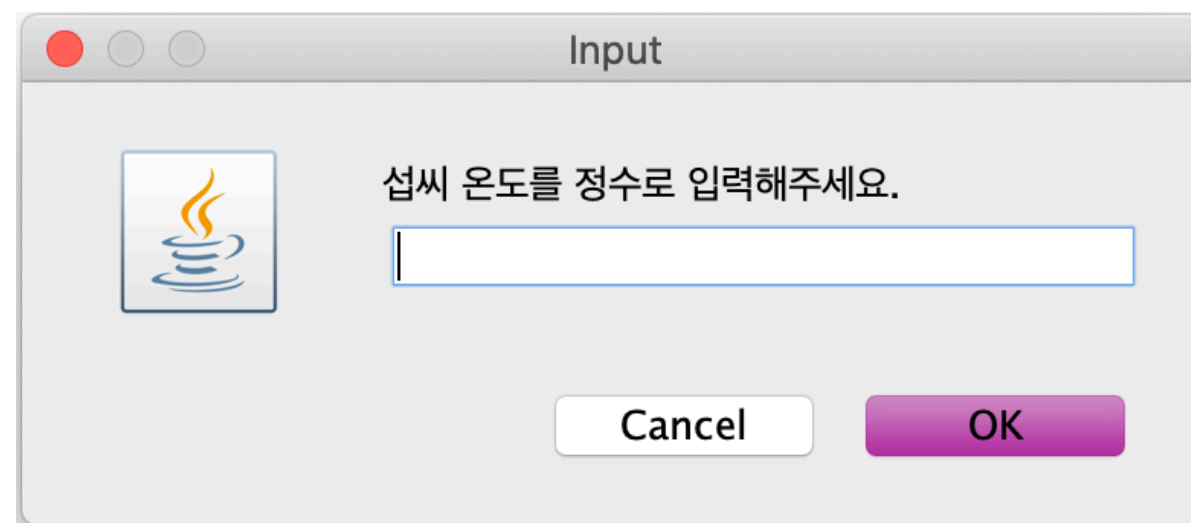
```
1 public class TempConverter {  
2     public static void main(String[] args) {  
3         double c = Double.parseDouble(args[0]);  
4         double f = (9.0 / 5.0) * c + 32;  
5         System.out.println("섭씨로 " + c + "도는 화씨로 " + f + "도 입니다.");  
6     }  
7 }
```

```
1 import java.text.*;  
2  
3 public class TempConverter {  
4     public static void main(String[] args) {  
5         double c = Double.parseDouble(args[0]);  
6         double f = (9.0 / 5.0) * c + 32;  
7         DecimalFormat formatter = new DecimalFormat("0.0");  
8         System.out.println("섭씨 " + c + "도 = 화씨 " + formatter.format(f) + "도");  
9     }  
10 }
```

사용자 입력 Input

입력창에서 입력

Input via Input Message Window



CelsiusToFahrenheit.java

```
1 ⊖ import java.text.*;
2 import javax.swing.*;
3
4 public class CelsiusToFahrenheit {
5
6 ⊖     public static void main(String[] args) {
7         String message = "섭씨 온도를 정수로 입력해주세요.";
8         String input = JOptionPane.showInputDialog(message);
9         int c = Integer.parseInt(input);
10        double f = (9.0 / 5.0) * c + 32;
11        DecimalFormat formatter = new DecimalFormat("0.0");
12        System.out.print("섭씨 " + c + "도는 ");
13        System.out.println("화씨로 " + formatter.format(f) + "도 입니다.");
14    }
15 }
```