

[디지털 컨버전스] 스마트 콘텐츠와 웹 융합 응용SW 개발자 양성과정

강사 : 이상훈

학생 : 임초롱

Constructor (생성자), Method (매서드), Instance (인스턴스), 객체 (Object)

링크 <https://github.com/limcholong/LectureContents/blob/main/java/CholongLim/Day8/src/ConstructorTest.java>

< 기본 개념 >

Constructor(생성자) :

- 자바에서 클래스는 '생성자'라고 하는 특수한 매서드를 구현할 수 있는 기능을 제공한다.
- 생성자는 객체를 처음 생성할 때 초기값을 설정해주는 역할을 수행한다.
- 초기 생성에 한정하여 Setter의 역할을 대신해줄 수 있다.

Constructor(생성자)의 특징 :

- 리턴 타입이 없다.
: 생성자는 인스턴스를 생성해주는 역할을 하는 특수한 매서드.
그러나 리턴 값이 있다면 엉뚱한 객체가 생성될 것이다.
따라서 리턴 값을 필요로 하는 작업에서는 생성자를 사용하지 않는다.
- 클래스 이름과 매서드 이름이 같다.
- New를 할 때 호출된다.
- 생성자에 붙어있는 괄호는 호출하는 용도이다.

Method(매서드)의 출력 :

- 매서드의 리턴값 뒤에 있는 값이 실행결과가 된다.
- 리턴값은 그 매서드를 종료시키는 일을 한다.
- 매서드는 리턴값이 어떤 데이터타입인지 정해주어야 한다.
(String, int)
- void는 리턴값이 없는 매서드를 만들 때 사용한다.

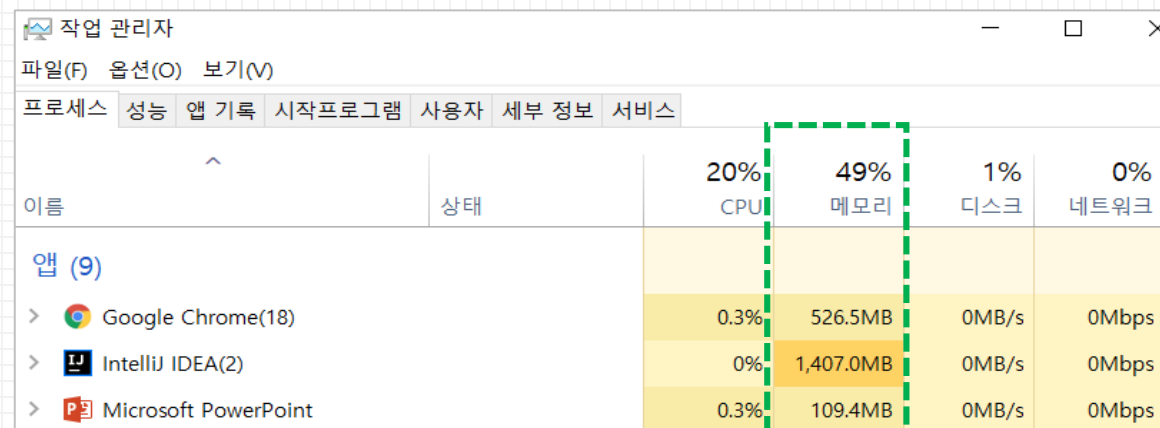
< 기본 개념 >

Instance(인스턴스) :

- 인스턴스는 클래스를 이용하여 서로 다른 데이터로 같은 매서드를 사용할 수 있도록 클래스를 복제하는 것 .
- 코드의 중복을 피하고 다용도로 사용할 수 있도록 하는 것이다.
- 특정 클래스를 인스턴스로 바꿀 경우 클래스 코드에서 static을 제거해 주어야한다.

객체(Object) :

- 우리가 사용하는 모든 데이터는 메모리에 올라가야지 사용할 수 있고 눈으로 볼 수 있다.
- 객체란 메모리에 데이터를 올렸습니다의 추상화 된 표현이다.
- 프로그램이 꺼지지 않고 켜진 상태,



작업 관리자					
파일(F) 옵션(O) 보기(V)					
프로세스 성능 앱 기록 시작프로그램 사용자 세부 정보 서비스					
이름		상태	20% CPU	49% 메모리	1% 디스크
					0% 네트워크
앱 (9)					
>	Google Chrome(18)		0.3%	526.5MB	0MB/s
>	IntelliJ IDEA(2)		0%	1,407.0MB	0MB/s
>	Microsoft PowerPoint		0.3%	109.4MB	0Mbps

Constructor (생성자)

링크 <https://github.com/limcholong/LectureContents/blob/main/java/CholongLim/Day8/src/WhyConstructorTest.java>

```
1 class Constest {
2     /* 데이터 저장 영역 시작 */
3     int age;
4     String name;
5     /* 데이터 저장 영역 끝 */
```

생성자의 형식 :

여러 형식으로 구성할 수 있다.

함수 오버로딩 :

이름이 같고 입력이 다른 케이스의 매서드를 사용하는 방식.
입력의 개수로 판단하는 것이 아닌 사용되는 입력의 데이터 타입을 보고 판단한다.

- ① Constest() {} :
- ② Constest(int a) {} : 정수 a값
- ③ Constest(float f) {} : 실수 f값
- ④ Constest(int a, String n) {} : 정수 a값, 문자열 n값

```
7     /* 기능 설정 영역 시작 */
8     // 생성자는 아래와 같이 여러 형식으로 구성할 수 있다!
9     // 이와 같이 이름이 같고 입력이 다른 케이스로 매서드를 사용하는 방식에 대해 함수 오버로딩이라고 한다.
10    // 입력의 개수를 가지고 판단하지 않으며 사용되는 입력에 데이터타입을 보고 판단한다는 점에 주의해야 한다.
11    Constest() {
12        System.out.println("안녕 나는 Constest() 이라고해!");
13    }
14    Constest(int a) {
15        System.out.println("안녕 나는 Constest(int a) 이라고해!");
16
17        age = a;
18    }
19    Constest(float f) {
20        System.out.println("안녕 나는 Constest(float f) 라고해!");
21    }
22    Constest(int a, String n) {
23        System.out.println("안녕 나는 Constest(int a, String n) 이라고해!");
24
25        name = n;
26        age = a;
27    }
28
29    public int getAge() {
30        return age;
31    }
32    public String getName() {
33        return name;
34    }
35    /* 기능 설정 영역 끝 */
```

Constructor (생성자)

링크 <https://github.com/limcholong/LectureContents/blob/main/java/CholongLim/Day8/src/WhyConstructorTest.java>

```
38 public class WhyConstructorTest {
39     public static void main(String[] args) {
40         ConstTest ct1 = new ConstTest();
41         ConstTest ct2 = new ConstTest(a: 10);
42         ConstTest ct3 = new ConstTest(a: 20, n: "hi");
43         ConstTest ct4 = new ConstTest(a: 40);
44         ConstTest ct5 = new ConstTest(f: 3.3f);
45
46
47
48         System.out.printf("ct1 name = %s, age = %d\n", ct1.getName(), ct1.getAge());
49         System.out.printf("ct2 name = %s, age = %d\n", ct2.getName(), ct2.getAge());
50         System.out.printf("ct1 name = %s, age = %d\n", ct1.getName(), ct1.getAge());
51
52         System.out.printf("ct3 name = %s, age = %d\n", ct3.getName(), ct3.getAge());
53         System.out.printf("ct4 name = %s, age = %d\n", ct4.getName(), ct4.getAge());
54     }
```

```
Run: WhyConstructorTest x
"C:\Program Files\Java\jdk-15.0.2\bin\java.exe"
안녕 나는 ConstTest() 이라고해!
안녕 나는 ConstTest(int a) 이라고해!
안녕 나는 ConstTest(int a, String n) 이라고해!
안녕 나는 ConstTest(int a) 이라고해!
안녕 나는 ConstTest(float f) 라고해!
ct1 name = null, age = 0
ct2 name = null, age = 10
ct1 name = null, age = 0
ct3 name = hi, age = 20
ct4 name = null, age = 40
Process finished with exit code 0
```

인스턴스를 호출 할 때 매개 변수에 해당하는 값을 넣어준다.

```
ConstTest ct1 = new ConstTest();
ConstTest ct2 = new ConstTest(a:10);
ConstTest ct3 = new ConstTest(a:20,n:"hi" );
ConstTest ct4 = new ConstTest(a:40);
ConstTest ct5 = new ConstTest(f:3.3f);
```

ConstTest()에는 변수가 입력되지 않아 인스턴스 생성 시 자동으로 초기화 값이 들어간다.
(String 초기값 = null, int 초기값 = 0)

기능설정 영역에서 생성자에 적었던

```
ConstTest() {
    System.out.println("안녕 나는 ConstTest() 이라고해!");
}
등이 인스턴스 호출에 따라 출력이 되며,
```

```
System.out.printf("ct1 name = %s, age = %d\n", ct1.getName(), ct1.getAge());
System.out.printf("ct2 name = %s, age = %d\n", ct2.getName(), ct2.getAge());
System.out.printf("ct1 name = %s, age = %d\n", ct1.getName(), ct1.getAge());
```

등이 getAge, getName 의 리턴 타입의 값을 얻으며 출력된다.

34번 : 생성자 문제

링크 https://github.com/limcholong/LectureContents/blob/main/java/CholongLim/Day8/src/Day8_Quiz34Explain.java

```
1 class TestDice {
2     int comDice;
3     int userDice;
4
5     TestDice() {
6         comDice = getRandomDice();
7         userDice = getRandomDice();
8     }
9
10    int getRandomDice() {
11        return (int)(Math.random() * 6 + 1);
12    }
13    Boolean userWin() {
14        System.out.printf("%d(컴퓨터) vs %d(사용자)\n", comDice, userDice);
15
16        if (comDice > userDice) {
17            return false;
18        } else if (comDice < userDice) {
19            return true;
20        } else {
21            System.out.println("무승부입니다.");
22            return false;
23        }
24    }
25
26    public class Day8_Quiz34Explain {
27        public static void main(String[] args) {
28
29            TestDice td = new TestDice();
30
31            if (td.userWin()) {
32                System.out.println("사용자가 승리하였습니다.");
33            } else {
34                System.out.println("컴퓨터도 못이겼다. 폐관수련이 답이다.");
35            }
36        }
37    }
```

34번 문제 내용 :

이전에 random과 제어문을 활용한 주사위 게임을 Class 방식으로 다시 만들어보자. 컴퓨터도 주사위를 굴리고 사용자도 주사위를 굴려서 누가 더 큰 숫자를 얻었는지 확인해보자.
(필요에 따라 커스텀 매서드를 만들어 사용하는 것도 좋다.)

```
Run: Day8_Quiz34Explain
"C:\Program Files\Java\jdk-15.0.2\
4(컴퓨터) vs 1(사용자)
컴퓨터도 못이겼다. 폐관수련이 답이다.
Process finished with exit code 0
```

생성자 문제 풀이 1)

-Boolean형 생성자의 사용 :

Boolean은 참, 거짓을 표현할 수 있는 데이터 타입이다.
userWin의 true, false에 따라 값이 달라진다.
comDice < userDice일때만 참값이다.

이하, 하단 if문에서 userWin()이 참일때,
“사용자가 승리하였습니다.”를 출력하고
userWin()이 거짓일때, (comDice > userDice, comDice = userDice)
“컴퓨터도 못이겼다. 폐관수련이 답이다.”를 출력한다.

한계_ 무승부 일때도 false값으로, 컴퓨터를 못이긴 것으로 출력된다.

35번 : 생성자 문제

링크 https://github.com/limcholong/LectureContents/blob/main/java/CholongLim/Day8/src/Day8_Quiz35Explain.java

```
1 class TestDice3 {
2     int comDice;
3     int userDice;
4
5     TestDice3() {
6         comDice = getRandomDice();
7         userDice = getRandomDice();
8     }
9
10    int getRandomDice() {
11        return (int)(Math.random() * 6 + 1);
12    }
13
14    void checkWinner() {
15        switch (whoWin()) {
16            case 1:
17                System.out.printf("패관수련입니다. %d(컴퓨터) vs %d(사용자)\n", comDice, userDice);
18                break;
19            case 2:
20                System.out.printf("사용자가 이겼습니다. %d(컴퓨터) vs %d(사용자)\n", comDice, userDice);
21                break;
22            case 3:
23                System.out.printf("비겼으니 형은 면하였습니다. %d(컴퓨터) vs %d(사용자)\n", comDice, userDice);
24                break;
25        }
26    }
27 }
```

Boolean은 참, 거짓으로 두가지만 결과만을 도출,
switch ~ case를 통해 여러가지 결과를 도출할 수 있다.

35번 문제 내용 :

34번 문제를 보완하여, 무승부일 경우를 포함한
프로그램 코드를 만들어보자.

```
28 int whoWin() {
29     if (comDice > userDice) {
30         return 1;
31     } else if (comDice < userDice) {
32         return 2;
33     } else {
34         return 3;
35     }
36 }
37
38
39 public class Day8_Quiz35Explain {
40     public static void main(String[] args) {
41         TestDice3 td = new TestDice3();
42         td.checkWinner();
43     }
44 }
45
```

생성자로 인해 간단해진 메인 코드

Run: Day8_Quiz35Explain x

↑
↓
C:\Program Files\Java\jdk-15.0.2\bin\java.exe
비겼으니 형은 면하였습니다. 1(컴퓨터) vs 1(사용자)

Process finished with exit code 0

35번 - 2 : 생성자 문제

링크 https://github.com/limcholong/LectureContents/blob/main/java/CholongLim/Day8/src/Day8_Quiz35Explain.java

```
1 class Dice3 {
2     int comDice;
3     int userDice;
4
5     Dice3() {
6         comDice = getRandomDice();
7         userDice = getRandomDice();
8     }
9     int getRandomDice() {
10        return (int)(Math.random() * 6 + 1);
11    }
12 }
13
14 public class Day8_Quiz35 {
15     public static void main(String[] args) {
16
17         Dice3 dg = new Dice3();
18
19         if(dg.comDice > dg.userDice) {
20             System.out.printf("%d(컴퓨터) vs %d(사용자)\n", dg.comDice, dg.userDice);
21             System.out.println("컴퓨터의 승리입니다.");
22         } else if(dg.comDice < dg.userDice) {
23             System.out.printf("%d(컴퓨터) vs %d(사용자)\n", dg.comDice, dg.userDice);
24             System.out.println("사용자의 승리입니다.");
25         } else {
26             System.out.printf("%d(컴퓨터) vs %d(사용자)\n", dg.comDice, dg.userDice);
27             System.out.println("무승부입니다.");
28         }
29     }
30 }
```

35번 문제 내용 :
다시 풀어보기

Run: Day8_Quiz35 ×

6(컴퓨터) vs 5(사용자)
컴퓨터의 승리입니다.

Process finished with exit code 0

Run: Day8_Quiz35 ×

3(컴퓨터) vs 6(사용자)
사용자의 승리입니다.

Process finished with exit code 0

Run: Day8_Quiz35 ×

2(컴퓨터) vs 2(사용자)
무승부입니다.

Process finished with exit code 0