

ШИНЖЛЭХ УХААН ТЕХНОЛОГИЙН ИХ СУРГУУЛЬ
МЭДЭЭЛЭЛ, ХОЛБООНЫ ТЕХНОЛОГИЙН СУРГУУЛЬ

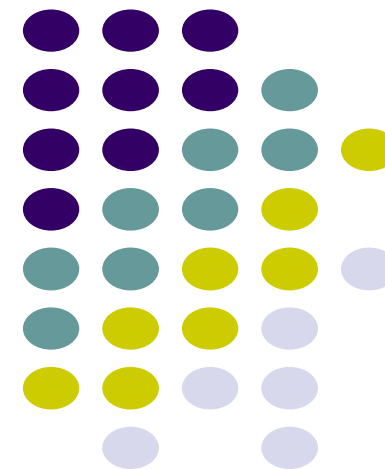
F.CS202
ОБЪЕКТ ХАНДЛАГАТ ПРОГРАМЧЛАЛ

Лекц №5

Битүүмжлэл ба классын бүрэлдэхүүн

док., дэд проф. Б.Батзолбоо
маг. Б.Мөнхбуян

2021 он



Агуулга



- Рекурс функц
- Жавагийн стандарт классын сантай ажиллах
 - Математик функцууд ашиглах
 - Тэмдэгт мөртэй ажиллах
 - Огноотой ажиллах
- BlueJ хэрэгжүүлэлт.



Өмнөх лекцээр

- Байгуулагч функц, түүний олон хэлбэржилт
- Устгагч функцийг тухай
- Функц, процедур, тэдгээрийн олон хэлбэржилт
- Жава пакетын тухай



Рекурсив функц

- Өөрөө өөрийгөө дууддаг функцийг рекурсив функц гэнэ.
- Жишээ: факториал
 - $5! = 5 * 4 * 3 * 2 * 1$
 - Дараах тэнцэтгэлүүд үнэн:
 - $5! = 5 * 4! \quad 3! = 3 * 2! \quad 1! = 1$
 - $4! = 4 * 3! \quad 2! = 2 * 1! \quad 0! = 1$
- Факториалыг рекурсээр тооцоолж болно.

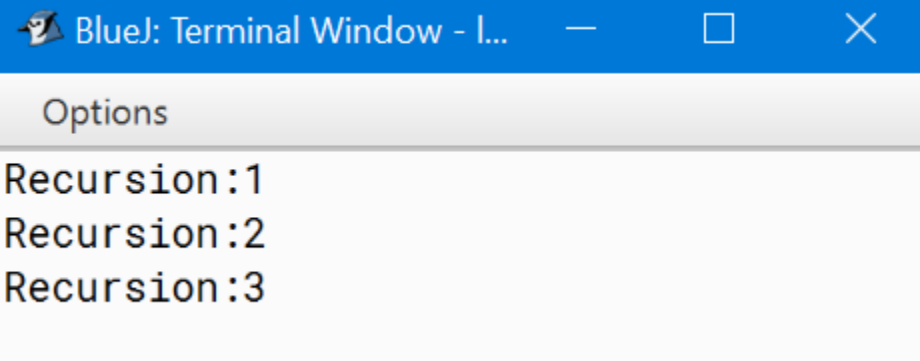
Рекурсив функц



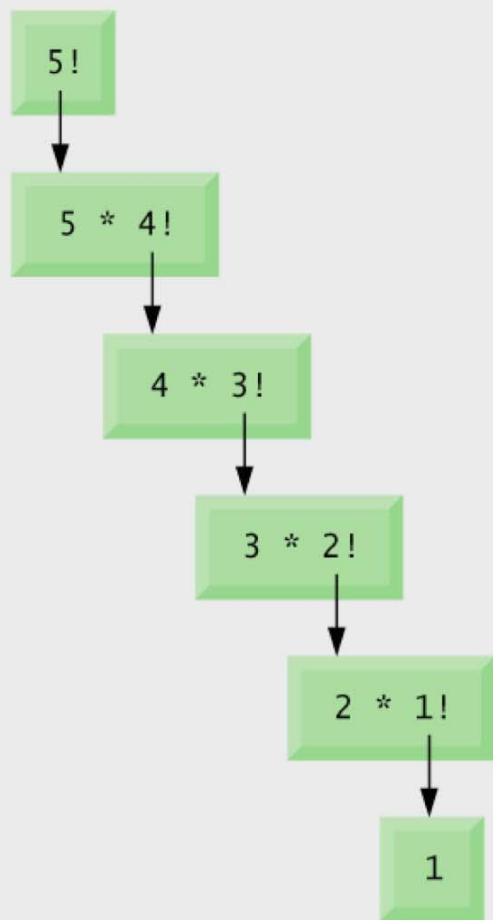
```
буцаах_төрөл_методынНэр() {  
    //функцийн код бичих хэсэг  
    ...  
    //өөрийгөө дуудах  
    методынНэр();  
}
```

Жишээ

```
static int count=0;  
static void recFunc() {  
    count++;  
    if(count<4) {  
        System.out.println("Recursion:" + count);  
        recFunc();  
    }  
}
```



5! –ын рекурсив тооцоолол



Рекурс дуудалтын дараалал

```
factorial(5)
  factorial(4)
    factorial(3)
      factorial(2)
        factorial(1)
          return 1
        return 2*1 = 2
      return 3*2 = 6
    return 4*6 = 24
  return 5*24 = 120
```

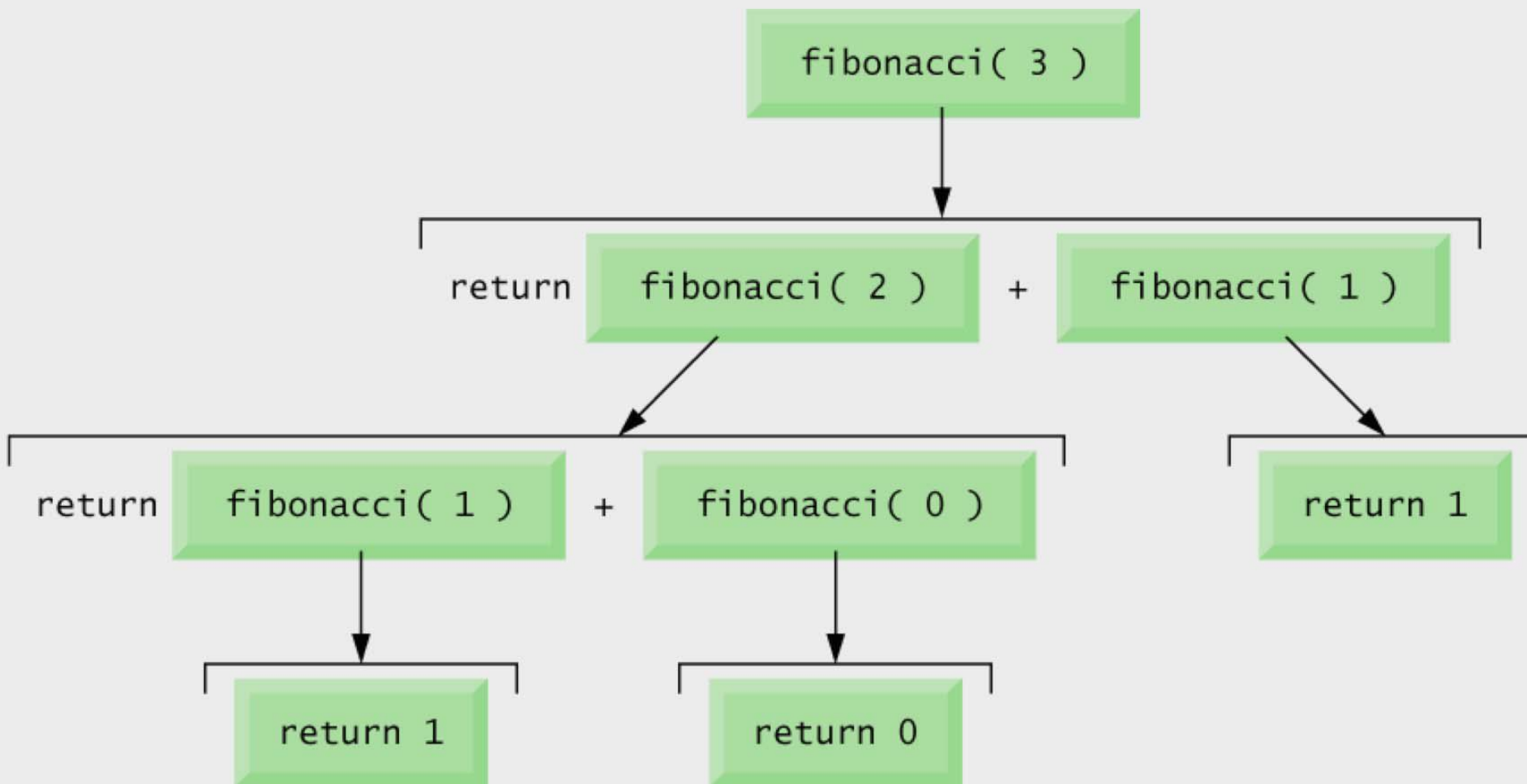
Рекурс дуудалт бүрийн буцаах утгууд

Жишээ: Факториал олох



```
public class Program {  
    public static int fac(int n) {  
        if (n <= 1)  
            return 1;  
        else  
            return (n * fac(n - 1));  
    }  
    public static void main() {  
        System.out.println(fac(5));  
    }  
}
```

Fibonacci(3) –н рекурс дуудалт



Жишээ: Фибоначийн тоо олох



```
public class Program {  
    public static long fib(long n) {  
        if (n <= 1)  
            return n;  
        else  
            return fib(n - 1) + fib(n - 2);  
    }  
    public static void main() {  
        System.out.println(fib(3));  
    }  
}
```



Тогтмол

- Хувьсагчийн утгыг өөрчилж болдог. Харин утга нь өөрчлөгддөггүй хувьсагч зарлах бол тогтмолыг ашиглана.

```
final double PI = 3.14159;  
final int MONTH_IN_YEAR = 12;  
final short FARADAY_CONSTANT = 23060;
```

final үгээр тогтмол
зарлаж байгааг
илэрхийлнэ

Тогтмолын нэр
/том үсгээр/

Тогтмолын утга

Жишээ



```
//Радиусыг оруулахад тойргийн талбай болон уртыг  
олох програмын хэсэг  
  
final double PI = 3.14159;  
  
double radius, area, circumference;  
  
//талбай болон уртыг тооцоолох  
area = PI * radius * radius;  
circumference = 2.0 * PI * radius;  
  
System.out.println("Өгөгдсөн радиус: " + radius);  
System.out.println("Талбай: " + area);  
System.out.println("Тойргийн урт: " + circumference);
```

Жава стандарт классын сан



Math класс



- **java.lang** пакетад агуулагдах **Math** класс нь өргөн ашиглагддаг математик функцүүдийг хэрэгжүүлэх методуудыг агуулна.

```
import java.lang.Math;
```

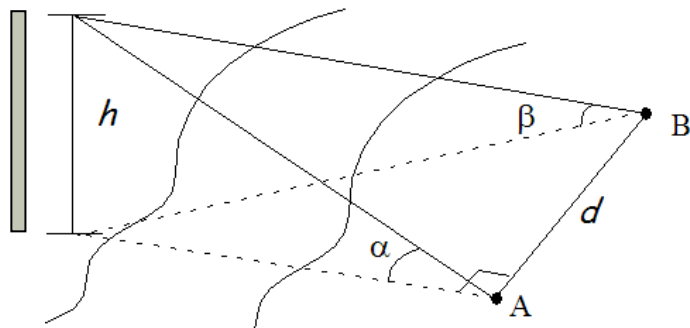
```
double    num, x, y;  
  
x = ...;  
y = ...;  
  
num = Math.sqrt(Math.max(x, y) + 12.4);
```

Math классын зарим методууд



Метод	Тайлбар	Жишээ
<code>abs(a)</code>	а тооны абсолют утга	<code>abs(5) → 5, abs(-5) → 5</code>
<code>exp(a)</code>	Натурал тоо e –г а зэрэгт дэвшүүлэх	<code>exp(2) → 7.389056099</code>
<code>log(a)</code>	а тооны натурал логарифм(e суурьтай).	<code>log(2.71183) → 1</code>
<code>floor(a)</code>	а тоог доош нь тоймлох	<code>floor(4.4) → 4</code> <code>floor(4.7) → 4</code>
<code>max(a,b)</code>	а болон b тооны ихийг олох.	<code>max(23,40) → 40</code>
<code>pow(a,b)</code>	а тоог b зэрэгт дэвшүүлэх.	<code>pow(2.0,4.0) → 16.0</code>
<code>sqrt(a)</code>	а тооны квадрат язгуурыг олох.	<code>sqrt(9.0) → 3.0</code>
<code>sin(a)</code>	(тригонометрийн бүх функцүүд радианаар тооцоологдоно.)	<code>Sin(π/2)→ 1.0</code>
<code>toDegrees</code>	Радианаар өгөгдсөн өнцгийг градус руу хөрвүүлнэ	<code>toDegrees(π/4)→ 45.0</code>
<code>toRadians</code>	Өнцгийн градусыг радианаар илэрхийлэх	<code>toRadians(90.0) → 1.5707963</code>

Шонгийн өндрийг тооцоолох



$$h = \frac{d \sin \alpha \sin \beta}{\sqrt{\sin(\alpha + \beta) \sin(\alpha - \beta)}}$$

```
alphaRad = Math.toRadians(alpha);  
betaRad  = Math.toRadians(beta);  
  
height = ( distance * Math.sin(alphaRad) * Math.sin(betaRad) )  
          /  
          Math.sqrt( Math.sin(alphaRad + betaRad) *  
                     Math.sin(alphaRad - betaRad) );
```

Random класс - Санамсаргүй тоо үүсгэх1



- Random классын nextInt(n) метод:

0 -оос n-1 хүртэл санамсаргүй тоо үүсгэх

```
import java.util.Random;
. . .
Random random = new Random();
. . .
int number = random.nextInt(11); //return x, 0 <= x <= 10
```

Харин [min, max] завсарт санамсаргүй тоо үүсгэх бол: (min <= max байх)

```
. . .
int number = random.nextInt(max - min + 1) + min;
```


DecimalFormat класс



- **DecimalFormat** - тоон утгыг форматжуулж хэвлэхэд ашиглана.

```
import java.text.*;
```

```
double num = 123.45789345;
```

```
DecimalFormat df = new DecimalFormat("0.000");  
//гурван орны нарийвчлалтай
```

```
System.out.print(num);
```

→ 123.45789345

```
System.out.print(df.format(num));
```

→ 123.458



Formatter класс

- Гаралтыг утгыг хэлбэржүүлж харуулна.
- Эхлээд классын инстансыг үүсгэнэ.

```
import java.util.*;
```

```
Formatter formatter = new Formatter(System.out);
```

- format методыг дуудна.

```
int num = 467;  
formatter.format("%6d", num);
```

- Гаралтын утгыг 6 орноор харуулна.

			4	6	7
--	--	--	---	---	---

Formatter классын format метод

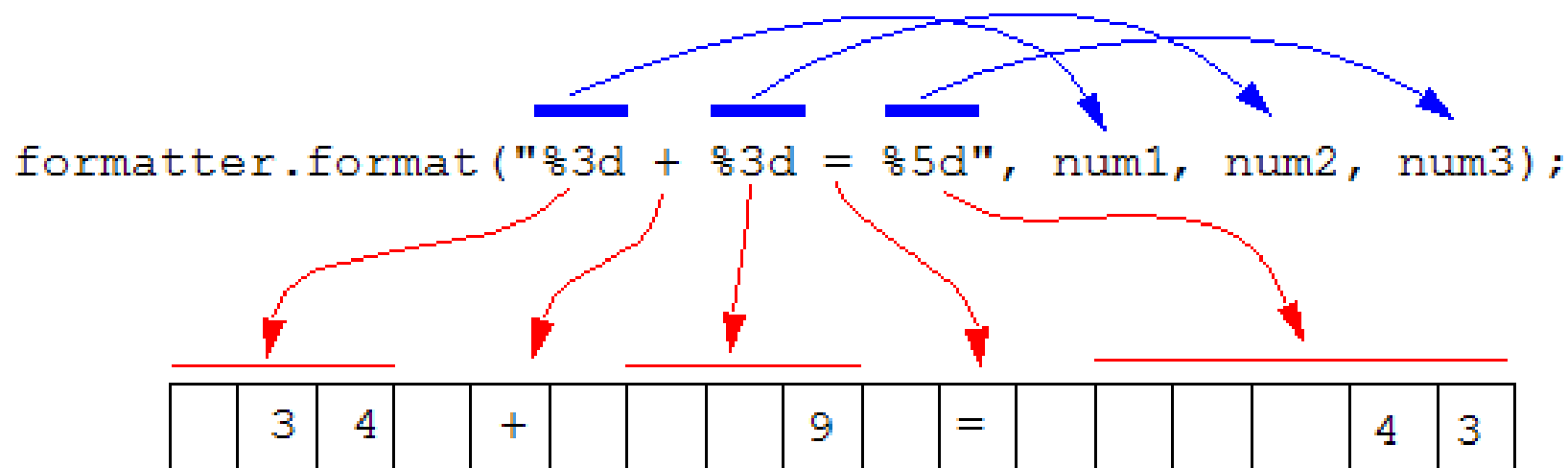


- Бичигдэх хэлбэр:

```
format(<control string>, <expr1>, <expr2>, . . . )
```

Жишээ нь:

```
int num1 = 34, num2 = 9;  
int num3 = num1 + num2;  
formatter.format("%3d + %3d = %5d", num1, num2, num3);
```



PrintStream-ын format метод



- Formatter классыг шууд ашиглахаас гадна PrintStream (System.out)-ын format методыг ашиглан өмнөхтэй ижил хэлбэрээр харуулж болно.

```
Formatter formatter = new Formatter(System.out);  
formatter.format("%6d", 498);
```

доорхтой ижил:

```
System.out.format("%6d", 498);
```

Control Strings



- Бүхэл тоо

`% <талбарын урт> d`

- Бодит тоо

`% <талбарын урт> . <орны нарийвчлал> f`

- Тэмдэгт мөр

`% s`

String класс

```
import java.lang.*;
```



- Тэмдэгт мөр - Тэмдэгтүүдийн дараалал боловч Жавад энэ нь тэмдэгтүүдийн дарааллыг дүрслэх объект байдаг. Ө/х: **String** классын инстансыг ашигладаг.
- Тэмдэгт мөр хувьсагчийг анхны утгаар үүсгэж, зарлах

```
String greeting = "Hello world!";
```

- Тэмдэгтэн массиваас тэмдэгт мөр үүсгэх

```
char[] helloArray = { 'h', 'e', 'l', 'l', 'o', '.' };  
String helloString = new String(helloArray);
```

Тэмдэгт мөртэй ажиллах



- Тэмдэгтийн уртыг олох

```
int len = s.length();
```

- Тэмдэгт мөр залгах

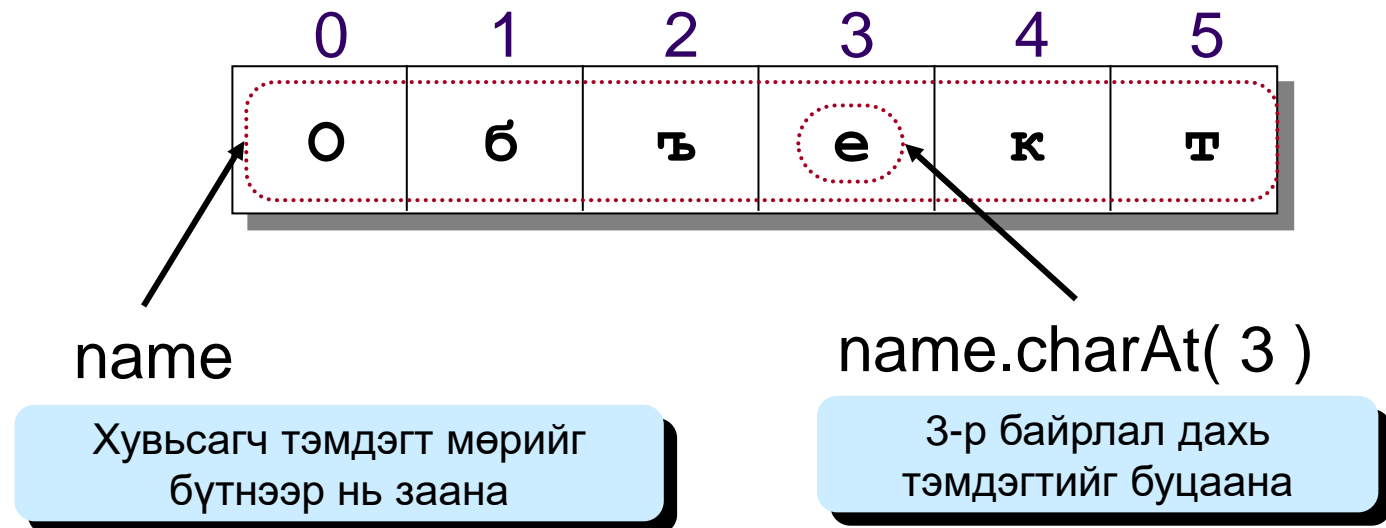
```
string1.concat(string2);  
string4 = string2 + string3;
```




Тэмдэгт мөртэй ажиллах

- Тэмдэгт мөр дэх тэмдэгтэд хандах **charAt** метод

```
String name = "Объект";
```



Тэмдэгт мөртэй ажиллах



- Жишээ: Эгшиг үсгийг тоолох

```
char    letter;
System.out.println("Таны нэр:");
String  name = scanner.next(); // 'scanner' тодорхойлогдсон гэж үзье
int     numberOfCharacters = name.length();
int     vowelCount  = 0;

for (int i = 0; i < numberOfCharacters; i++) {
    letter = name.charAt(i);

    if (    letter == 'a' || letter == 'A' ||
          letter == 'e' || letter == 'E' ||
          letter == 'i' || letter == 'I' ||
          letter == 'o' || letter == 'O' ||
          letter == 'u' || letter == 'U'
        ) {

        vowelCount++;
    }
}

System.out.print(name + ", таны нэрэнд" + vowelCount + " эгшиг байна");
```



Тэмдэгт мөртэй ажилладаг методууд

Метод	Тайлбар
compareTo	Тэмдэгт мөрүүдийг харьцуулах. <code>str1.compareTo(str2)</code>
substring	Тэмдэгт мөрөөс дэд мөр салгах. <code>str1.substring(1, 4)</code>
trim	Тэмдэгт мөрийн эхлэл болон төгсгөл дэх хоосон зайг арилгах. <code>str1.trim()</code>
valueOf	Өгөгдсөн үндсэн өгөгдлийн төрөл бүхий утгыг тэмдэгт мөр рүү хөрвүүлэх. <code>String.valueOf(123.4565)</code>
startsWith	Тэмдэгт мөр өгөгдсөн тэмдэгт мөрөөр эхэлж байвал үнэн утга буцаана. <code>str1.startsWith(str2)</code>
endsWith	Тэмдэгт мөр өгөгдсөн тэмдэгт мөрөөр төгсөж байвал үнэн утга буцаана. <code>str1.endsWith(str2)</code>

Тэмдэгт мөртэй ажиллах



- Том үсгүүд руу хувиргах, жижиг үсгүүд рүү хувиргах

```
String Str = new String("Welcome");  
System.out.println(Str.toLowerCase());  
System.out.println(Str.toUpperCase());
```

Хэлбэржүүлсэн тэмдэгт мөр үүсгэх



```
float floatVar = 123.12f;
int intVar = 12;
String stringVar = "Str";
String fs;
fs = String.format("The value of the float variable is " +
                   "%f, while the value of the integer " +
                   "variable is %d, and the string " +
                   "is %s", floatVar, intVar, stringVar);
System.out.println(fs);
```

Объект харьцуулах



- Үндсэн өгөгдлийн төрлүүдийн хувьд харьцуулах нэг л арга байдаг бол объект (reference өгөгдлийн төрөл)-ын хувьд 2 аргаар харьцуулж болно.
 1. 2 хувьсагч ижил объектыг зааж байгаа эсэхийг шалгах (== ашиглана)
 2. 2 тусдаа объект ижил агуулгатай эсэхийг шалгах

== ашиглан объект харьцуулах (Жишээ 1)



```
String str1 = new String("Java");  
String str2 = new String("Java");  
  
if (str1 == str2) {  
    System.out.println("Тэмдэгт мөрүүд тэнцүү");  
} else {  
    System.out.println("Тэмдэгт мөрүүд тэнцүү биш");  
}
```

Тэмдэгт мөрүүд тэнцүү биш

← str1 ба str2 нь өөр өөр String
объектыг зааж байгаа тул тэнцүү
биш байна.

== ашиглан объект харьцуулах (Жишээ 2)



```
String str1 = new String("Java");  
String str2 = str1;  
  
if (str1 == str2) {  
    System.out.println("Тэнцүү");  
} else {  
    System.out.println("Тэнцүү биш");  
}
```

Тэнцүү

← str1 ба str2 нь ижил
объектыг зааж байгаа
тул тэнцүү байна.

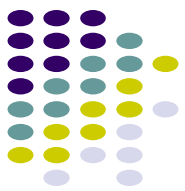
equals ашиглан ТЭМДЭГТ мөр харьцуулах



```
String str1 = new String("Java");  
String str2 = new String("Java");  
  
if (str1.equals(str2)) {  
    System.out.println("Тэнцүү");  
} else {  
    System.out.println("Тэнцүү биш");  
}
```

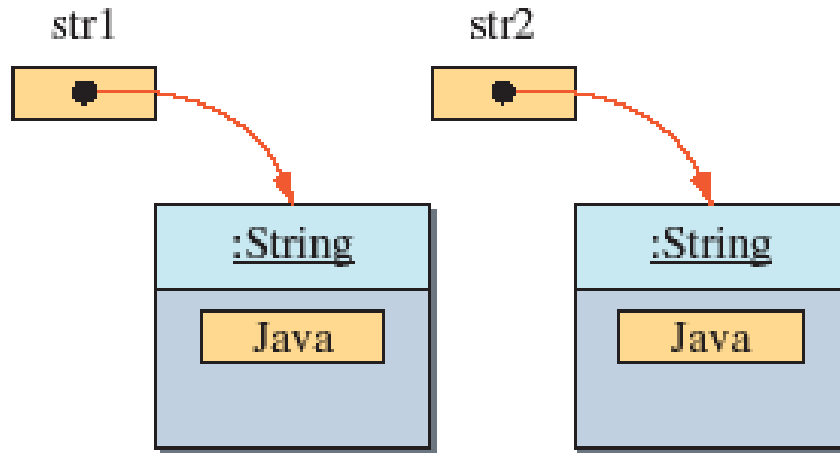
ТЭНЦҮҮ

← str1 ба str2 нь ижил
тэмдэгтийн дарааллыг
агуулж байгаа тул тэнцүү
байна.



== ажиллах зарчим

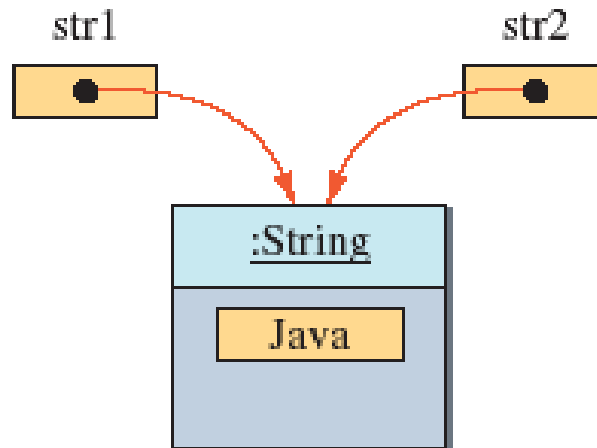
1-р тохиолдолд: 2 хувьсагч нь өөр өөр объектыг зааж байна.



```
String str1, str2;  
  
str1 = new String("Java");  
str2 = new String("Java");
```

`str1 == str2` → false

2-р тохиолдолд: 2 хувьсагч нь ижил объектыг зааж байна.



```
String str1, str2;  
  
str1 = new String("Java");  
str2 = str1;
```

`str1 == str2` → true

String объект үүсгэх

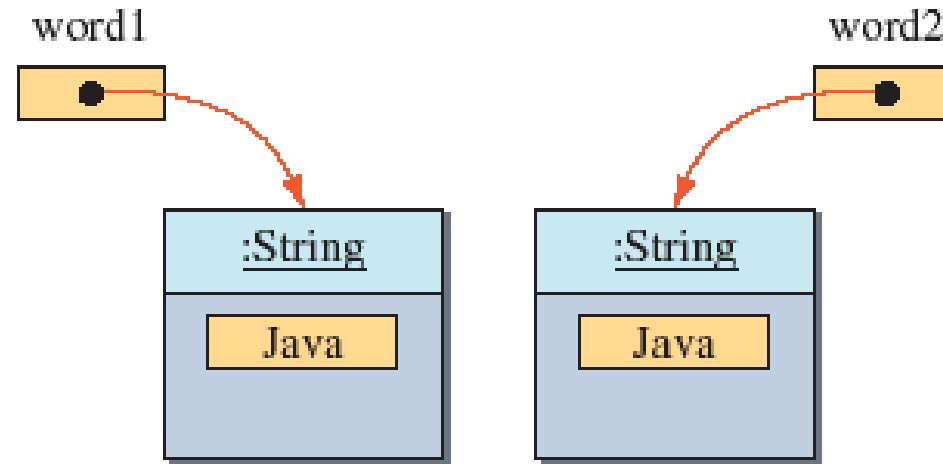


```
String word1, word2;
```

```
word1 = new String("Java");
```

```
word2 = new String("Java");
```

new оператор ашиглагдаж байгаа тул 2
шинэ объект үүснэ.



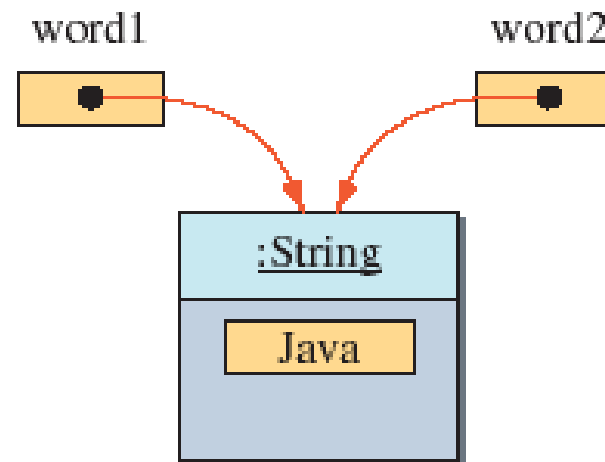
`word1 == word2` → false

```
String word1, word2;
```

```
word1 = "Java";
```

```
word2 = "Java";
```

“Java” тогтмол тэмдэгт мөр бүр нь нэг
объектыг заана.



`word1 == word2` → true



Огноотой ажиллах

- `java.time`, `java.util`, `java.sql`, `java.text` пакетууд огноо болон цагтай ажиллах классуудыг агуулж байдаг.
- Жавагийн нийтлэг хэрэглэгддэг огноотой ажилладаг классууд:
 - `java.util.Date` class
 - `java.sql.Date` class
 - `java.util.Calendar` class
 - `java.util.GregorianCalendar` class
 - `java.util.TimeZone` class
 - `java.sql.Time` class
 - `java.sql.Timestamp` class

Date класс ашиглах



- Ажиллах хугацааг тооцох

```
Date startTime = new Date();  
  
//хугацааг нь тооцохыг хүсч буй үйлдлээ энд бичих  
  
Date endTime = new Date();  
  
long elapsedTimeInMillisec =  
    endTime.getTime() - startTime.getTime();
```

Жишээ: <https://www.javatpoint.com/>



```
import java.util.Calendar;

public class CalendarExample1 {

    public static void main(String[] args) {

        Calendar calendar = Calendar.getInstance();

        System.out.println("Өнөөдрийн огноо: " + calendar.getTime());

        calendar.add(Calendar.DATE, -15);

        System.out.println("15 өдрийн өмнө: " + calendar.getTime());

        calendar.add(Calendar.MONTH, 4);

        System.out.println("4 сарын дараа: " + calendar.getTime());

        calendar.add(Calendar.YEAR, 2);

        System.out.println("2 жилийн дараа: " + calendar.getTime());

    } }
```

Хуанлийн мэдээлэл авах

```
import java.util.*;
```



```
Calendar cal = Calendar.getInstance();  
                //Өнөөдөр 2021.02.20 гэж үзье  
  
System.out.print("Өнөөдөр " +  
                (cal.get(Calendar.MONTH)+1) + "/" +  
                cal.get(Calendar.DATE) + "/" +  
                cal.get(Calendar.YEAR));
```

Гаралт

Өнөөдөр 2/20/2021



Calendar –ийн мэдээллийг авах

- Дараах хүснэгтэнд **Date**-ээс хуанлийн мэдээллийг салгаж авах тогтмолуудыг харуулав.

Constant	Description
YEAR	The year portion of the calendar date
MONTH	The month portion of the calendar date
DATE	The day of the month
DAY_OF_MONTH	Same as DATE
DAY_OF_YEAR	The day number within the year
DAY_OF_MONTH	The day number within the month
DAY_OF_WEEK	The day of the week (Sun — 1, Mon — 2, etc.)
WEEK_OF_YEAR	The week number within the year
WEEK_OF_MONTH	The week number within the month
AM_PM	The indicator for AM or PM (AM — 0 and PM — 1)
HOUR	The hour in 12-hour notation
HOUR_OF_DAY	The hour in 24-hour notation
MINUTE	The minute within the hour

ДҮГНЭЛТ



- Рекурс функц
- Тогтмолын хэрэглээ
- Жавагийн үндсэн классын сангуудтай танилцлаа
 - Тэмдэгт мөр нь объект хэлбэрээр байдаг тул харьцуулалт мөн ялгаатай хийгддэг
 - Математик функцүүдийг ашиглах
 - Огноо болон цагтай ажилладаг классуудыг агуулсан олон төрлийн пакетууд байдаг