



ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

М.ӘУЕЗОВ АТЫНДАҒЫ ОҢТҮСТІК ҚАЗАҚСТАН МЕМЛЕКЕТТІК УНИВЕРСИТЕТІ

«АҚПАРАТТЫҚ ЖҮЙЕЛЕР» КАФЕДРАСЫ

# ЗЕРТХАНАЛЫҚ ЖҰМЫС

**ТАҚЫРЫБЫ:** FOR ОПЕРАТОРЫ. FOR ОПЕРАТОРЫН ҚОЛДАНЫП  
ЦИКЛДЫҚ ҚҰРЫЛЫМДЫ ПРОГРАММАЛАРДЫ ӨНДЕУ.

ОРЫНДАҒАН: АЙДЫНБЕКОВА АРУЖАН

ҚАБЫЛДАҒАН : МАХАНОВА ЗЛИХА

# ЖОСПАР:

- for параметрімен циклдық оператор
- Циклдық құрылымды программаларды өңдеу.

# ЦИКЛ FOR ДЕГЕНІМІЗ НЕ?

FOR FOR ЦИКЛИ - БҰЛ АЛГОРИТМНІҢ СЫЗЫҚТЫҒЫН БҰЗАТЫН ЖӘНЕ КӨРСЕТІЛГЕН КОДТЫ БІРНеше РЕТ ОРЫНДАУҒА МҮМКІНДІК БЕРЕТІН БАСҚАРУ КОДЫНЫҢ ҚҰРЫЛЫМЫ. МЫСАЛЫ, СІЗГЕ 30 ТАМШЫ ДӘРІ ІШУ КЕРЕК

АЛГОРИТМ КЕЛЕСІДЕЙ БОЛАДЫ:

1. СТАҚАН ДАЙЫНДАҢЫЗ.
2. ҚАҚПАҚТЫ АШЫҢЫЗ.
3. 1 ТАМШЫ АЛЫҢЫЗ.
4. 2 ТАМШЫ АЛЫҢЫЗ.
- ...
32. 30 ТАМШЫ АЛЫҢЫЗ.
33. ДӘРІНІ ЖАБЫҢЫЗ.
34. АЛЫНҒАН БӨЛІКТІ АЛЫҢЫЗ.

БҰЛ АЛГОРИТМДІ ТЕЗІРЕК ТҮСІНДІРУГЕ БОЛАДЫ:

1. СТАҚАН ДАЙЫНДАҢЫЗ.
2. ТАМШЫ ҚАҚПАҒЫН АШЫҢЫЗ.
3. 30 ТАМШЫ АЛЫҢЫЗ.
4. ДӘРІНІ ЖАБЫҢЫЗ.
5. АЛЫНҒАН БӨЛІКТІ АЛЫҢЫЗ.

- For циклі қалай жұмыс істейді

For циклі келесідей қолданылады:

```
1  for(<начальная точка>; <условие выхода>; <оператори счетчика>) {  
2      // Тело цикла  
3  }  
4  Пример перебора цифр от 0 до 5 и вывод каждой в консоль:  
5  for(int i = 0; i < 5; i++) {  
6      System.out.println(i);  
7  }
```

Қорытынды:

0  
1  
2  
3  
4

- Циклден шығу шарты - логикалық өрнек. Егер ол жалған болса, цикл аяқталады. Жоғарыдағы мысалда,  $i$  1-ге көбейтіледі, егер оның мәні 5-тен аз болса, цикл жалғасады. Бірақ 5-тен үлкен немесе тең болғанда, цикл тоқтайды.
- Есептегіш оператор - санауыштың айнұмалысын түрлендіруді жүзеге асыратын өрнек. Жоғарыдағы мысалда  $i$  айнұмалысы 1-ге көбейтілді, яғни цикл тура 5 рет орындалады. Егер санауыш операторы  $i$  айнұмалысына 2 қосса, нәтиже басқаша болады:



```
1 for(int i = 0; i < 5; i = i + 2) {  
2     System.out.println(i);  
3 }
```

• Қорытынды:

0

2

4

Сондай-ақ, айнымалыны көбейтуге, бөлуге, дәрежеге дейін көтеруге болады, жалпы қалағаныңды істеуге болады. Ең бастысы, конверсия нәтижесінде сан шығады.



БҮКІЛ СХЕМАНЫ ҚОРЫТА ОТЫРЫП, ОСЫ ЦИКЛДІҢ ПРИНЦИПІ КЕЛЕСІДЕЙ: ЦИКЛ ДЕНЕСІНДЕ ОРНАЛАСҚАН КОД САНАУЫШ ОПЕРАТОРЫ ЦИКЛДАН ШЫҒУ ШАРТЫНА ДЕЙІН ТҮРЛЕНДІРУЛЕРДІ ҚАНША ЖАСАСА, СОНША РЕТ ОРЫНДАЛАДЫ.

```
1 for(int i = 0; true; i++) {
2     if(i % 1000000 == 0) System.out.println(i);
3 }
4 System.out.println("Loop ended");
```

СОДАН КЕЙІН ЦИКЛДАН КЕЙІНГІ КОД UNACHABLE ОПЕРАТОРЫНЫҢ ҚАТЕЛІГІМЕН БЕЛГІЛЕНЕДІ, ӨЙТКЕНІ ОЛ ЕШҚАШАН ОРЫНДАЛМАЙДЫ.

ТӨМЕНДЕГІ КОДТЫ ІСКЕ ҚОСУ КОНСОЛЬГЕ «LOOP ENDED» ШЫҒАРАДЫ НЕМЕСЕ ЦИКЛ ШЕКСІЗ ЖҰМЫС ІСТЕЙ МЕ?

```
1 for(int i = 0; i > -1; i++) {
2     if(i % 1000000 == 0) System.out.println(i);
3 }
4 System.out.println("Loop ended");
```

ЖАУАП: АЙНЫМАЛЫСЫ ЕРТЕ МЕ, КЕШ ПЕ ӨЗІНІҢ МАКСИМАЛДЫ МӘНІНЕ ЖЕТЕДІ, АЛ ОДАН ӘРІ ҰЛҒАЮ ОНЫ МАКСИМАЛДЫ ТЕРІС МӘНГЕ АЙНАЛДЫРАДЫ, НӘТИЖЕСІНДЕ ШЫҒУ ШАРТЫ ОРЫНДАЛАДЫ ( $i \leq -1$ ).

- Цикл `forEach`

- Ілмектермен жұмыс істегенде кейде массивтер мен коллекциялар бойынша қайталауға тура келеді. Сіз `for` циклын пайдаланып жиым бойынша қайтапай аласыз.

```
1 public void printAllElements(String[] stringArray) {
2     for(int i = 0; i < stringArray.length; i++) {
3         System.out.println(stringArray[i]);
4     }
5 }
```

Бұл дұрыс. Алайда, массивтің барлық элементтерін бірінен соң бірін қайтапай үшін олар әр құрыпты ойлап тапты. Оның көптанбасы

```
1 for(<Тип элемента> <Имя переменной, куда будет записан очередной элемент> : <Название массива>) {
2     // Тело цикла
3 }
```



- Сіз жолдар массивінің үстінен қайталап, әрқайсысын консольге келесі жолмен шығара

```
1 public void printAllElements(String[] stringArray) {  
2     for(String s : stringArray) {  
3         System.out.println(s);  
4     }  
5 }
```

## For циклдары қалай қолданылады?

- Енді Java-дағы for циклын әртүрлі мәселелерді шешу үшін қолданудың бірнеше мысалын қарастырайық.

Кері цикл (жоғарыдан төменге)

```
1 for(int i = 5; i > 0; i--) {  
2     System.out.println(i);  
3 }
```

Қорытынды:

5  
4  
3  
2  
1

- Бірнеше айнымалылар және цикл денесінде есептегішті көбейту
- For циклінде бірнеше айнымалыларды қолдануға болады, мысалы, оларды санауыш операторында түрлендіруге болады

```
1  int a = 0;
2  for(int i = 5; i > 0; i--, a++) {
3      System.out.print("Шаг: " + a + " Значение: ");
4      System.out.println(i);
5  }
```

- Қорытынды:

- Шаг: 0 Значение: 5
- Шаг: 1 Значение: 4
- Шаг: 2 Значение: 3
- Шаг: 3 Значение: 2
- Шаг: 4 Значение: 1

Немесе екі айнымалы және бір-біріне тең болғанға дейін цикл жариялаңыз:

```
1 for(int i = 5, j = 11; i != j; i++, j--) {  
2     System.out.println("i: " + i + " j: " + j);  
3 }
```

Қорытынды:

i: 5 j: 11

i: 6 j: 10

i: 7 j: 9

For циклінде ішкі циклдарды да жасауға болады. Бұл жағдайда циклдегі қадамдар саны көбейтіледі:

```
1  for(int i = 0; i < 5; i++) {
2      System.out.print(i + " | ");
3      for(int j = 0; j < 5; j++) {
4          System.out.print(j + " ");
5      }
6      System.out.print('\n');
7  }
```

Қорытынды:

```
0 | 0 1 2 3 4
1 | 0 1 2 3 4
2 | 0 1 2 3 4
3 | 0 1 2 3 4
4 | 0 1 2 3 4
```



- J санағышы бар циклде сыртқы циклдің санауышына қол жеткізуге болады. Бұл екіөлшемді, үшөлшемді және басқа массивтерден өтудің тамаша әдісі болып табылады:

```
1  int[][] array = { {0, 1, 2, 3, 4 },
2                      {1, 2, 3, 4, 5},
3                      {2, 3, 4, 5, 6},
4                      {3, 4, 5, 6, 7}};
5
6  for(int i = 0; i < array.length; i++) {
7      for(int j = 0; j < array[i].length; j++) {
8          System.out.print(array[i][j] + " ");
9      }
10     System.out.print('\n');
11 }
```

0 1 2 3 4

1 2 3 4 5

2 3 4 5 6

3 4 5 6 7