

While,do While,for,for- Break Kullanımı Ve Aralarındaki Farklar

Başlık altında döngülerin kullanımını ve aralarındaki farkları öğreneceksiniz.

While Döngüsü:

While : Kod bloğunun kaç kez tekrarlanacağını bilmeme durumudur.

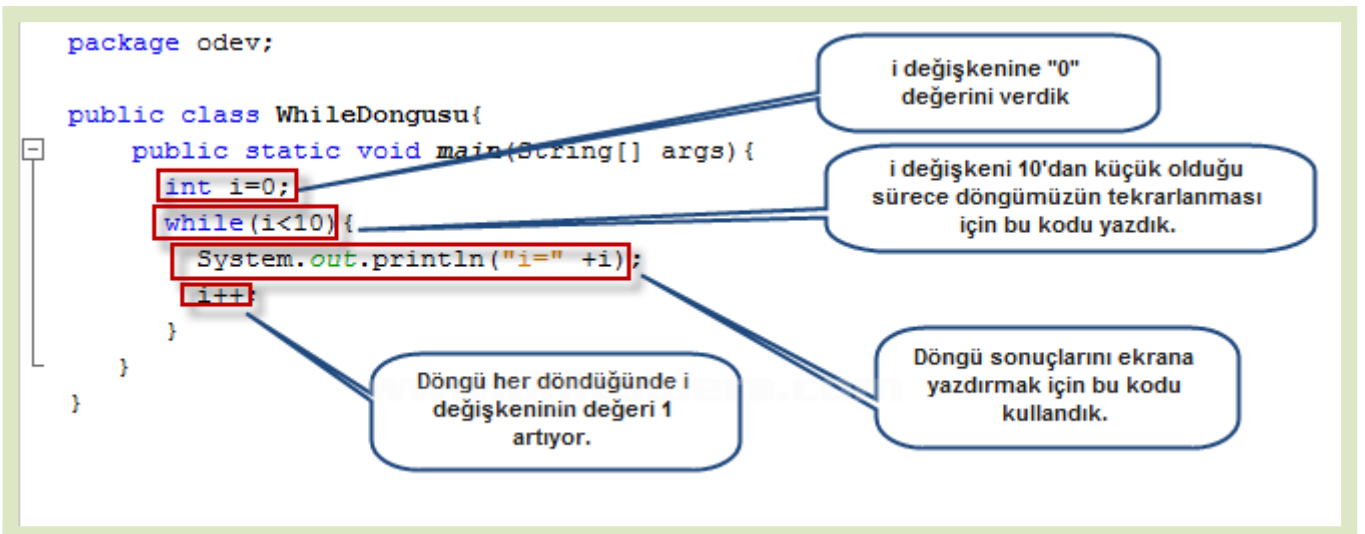
```
while(koşul){
    tekrarlanacak olan kodlar...
}
```

Bu döngü şeklinde başlangıçta koşul denetlenir.Doğruysa döngü devam eder,yanlışsa "tekrarlanacak kodlar" atlanır.Yani koşul doğru olduğu sürece işlem tekrarlanır.

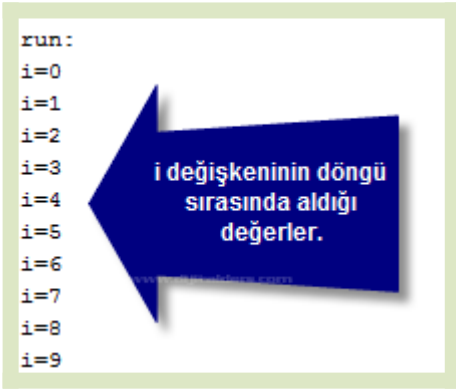
Örnek Kod:

```
public class WhileDongusu{
    public static void main(String[] args){
        int i=;
        while(i<5){
            System.out.println("i=" +i);
            i++;
        }
    }
}
```

Kod incelemesi:



Önizleme:



Do-While Döngüsü:

Do-While: Kodlar şarta bağlı olmadan en az bir kez çalıştırılır durumudur.

```
do{  
    tekrarlanacak olan kodlar...  
}while(koşul);
```

Bu döngüde koşul ne olursa olsun “en az bir kez” tekrarlanacak olan kodlar çalıştırılır.Çünkü koşul denetimi sonda olur.Koşul doğru ise döngü devam eder yoksa bir daha tekrarlanmaz.

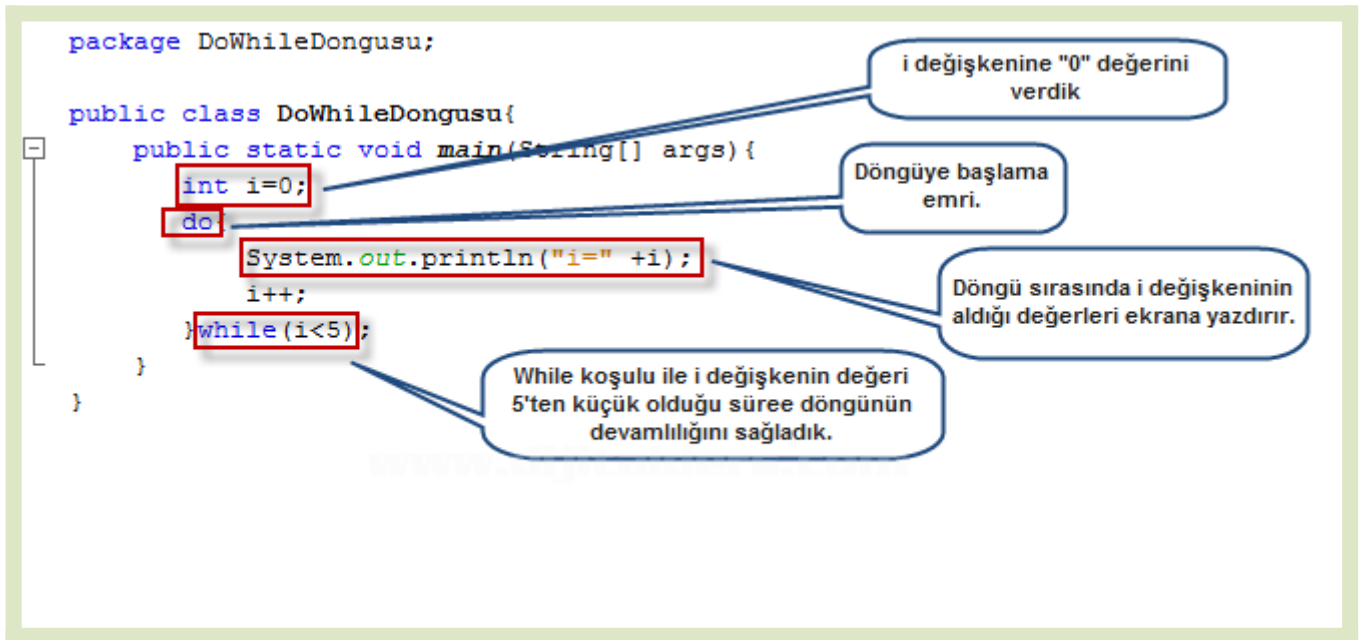
Örnek Kod:

```

public class DoWhileDongusu{
    public static void main(String[] args){
        int i=;
        do{
            System.out.println("i=" +i);
            i++;
        }while(i<);
    }
}

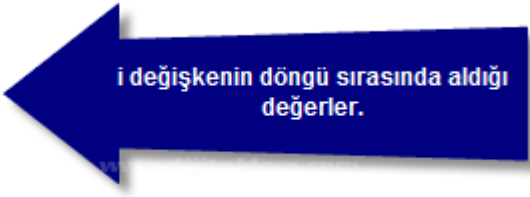
```

Kod incelemesi:



Önizleme:

```
run:
i=0
i=1
i=2
i=3
i=4
BUILD SUCCESSFUL (total time: 0 seconds)
```



For Döngüsü:

For: Döngü şartlarını biz belirliyoruz durumudur.

```
for(başlangıç; koşul; artış durumu){
    tekrarlanacak olan kodlar...
}
```

En temel döngü şeklidir.Tekrarın kaç kez olacağına karar verilebilir.

Örnek Kod:

```
public class ForDongusu{
    public static void main(String[] args){
        for(int i=;i<25;i++){
            System.out.println("i=" + i);
        }
    }
}
```

Kod incelemesi:

```

package ForDongusu;

public class ForDongusu{
    public static void main(String[] args){
        for(int i=0; i<10; i++){
            System.out.println("i=" + i);
        }
    }
}

```

i değişkeninin türünü integer olarak belirledik ve değerini "0" olarak belirledik.

i değişkeninin değerinin 10'dan küçük olduğu sürece döngünün devamlılığını sağladık.

Döngü devam ettiği sürece i değişkeninin değerini 1 artırır.

Döngü süresince i değişkeninin aldığı değerleri ekrana yazdırır.

www.dijitalders.com

Önizleme:

```

run:
i=0
i=1
i=2
i=3
i=4
i=5
i=6
i=7
i=8
i=9
BUILD SUCCESSFUL (total time: 0 seconds)

```

Döngü süresince i değişkeninin aldığı değerler.

www.dijitalders.com

For Döngüsü ile Break Komutu Kullanımı:

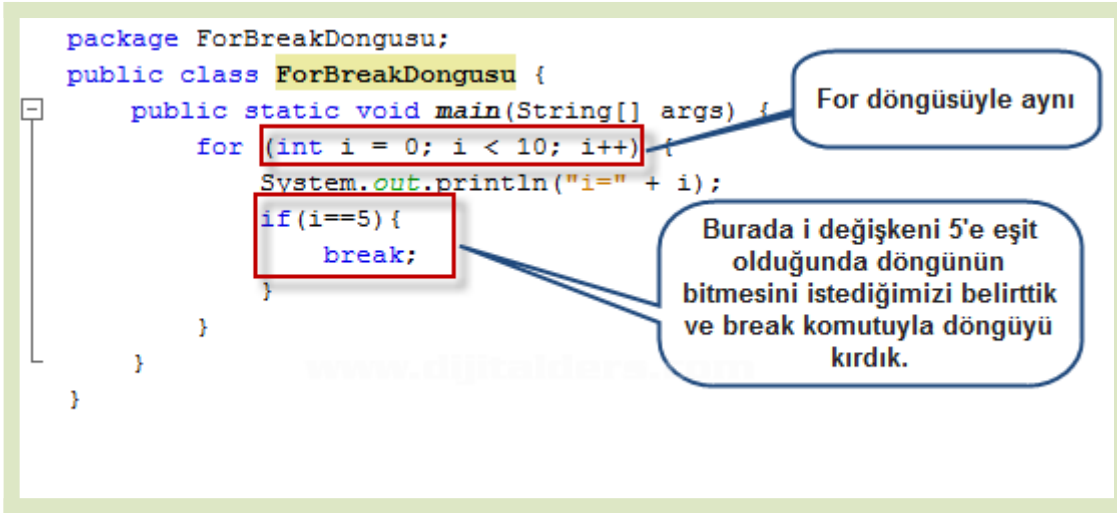
For döngüsü içerisinde koşulun her zaman sonuna kadar tamamlanması gerekli olmayabilir bu durumda **Break** ile döngüyü içindeki başka bir koşulla bitirebiliriz böylece koşul sonuna kadar tekrarlanmamış olur.

```
for(başlangıç; koşul; artış durumu){
    tekrarlanacak olan kodlar...
    if(değişkenin eşit olacağı değer){
        break;
    }
}
```

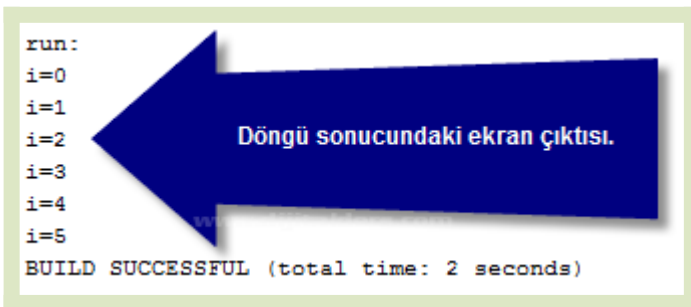
Örnek Kod:

```
package ForBreakDongusu;
public class ForBreakDongusu {
    public static void main(String[] args) {
        for (int i = ; i < 10; i++) {
            System.out.println("i=" + i);
            if(i==5){
                break;
            }
        }
    }
}
```

Kod incelemesi:



Önizleme:



Hazırlayan:Mehmet EMEK