

While Döngüsü

Genel bir while döngüsünün aşağıdaki yapısı vardır:

```
while (koşul) {  
  
    // çalıştırılacak kod  
  
    // koşulu güncelle  
}
```

Koşul, döngünün her bir yinelemesinin başında değerlendirilen bir boolean ifadedir. Eğer koşul doğruysa, döngü içindeki kod çalıştırılır. Kod çalıştırıldıktan sonra, koşul tekrar değerlendirilir. Eğer koşul hala doğruysa, kod tekrar çalıştırılır. Bu süreç, koşul yanlış olana kadar tekrar eder.

Aşağıdaki örnek, 1'den 10'a kadar olan sayıları yazdırır:

```
int i = 1;  
while (i <= 10) {  
    System.out.println(i);  
    i++;  
}
```

Değişken *i* 1'e başlatılır. Koşul *i* <= 10 olarak değerlendirilir. *i* 1 olduğundan koşul doğrudur. Döngü içindeki kod çalıştırılır, *i* değerini yazdırır ve *i* değerini 1 artırır. Koşul tekrar değerlendirilir. Şimdi *i* 2 olduğundan koşul doğrudur. Döngü içindeki kod tekrar çalıştırılır, *i* değerini yazdırır ve *i* değerini 1 artırır. Bu süreç, *i* 11 olduğunda sona erer. *i* 11 olduğunda koşul *i* <= 10 yanlış olduğu için döngü sona er.

Not: Bir while döngüsü herhangi bir başka döngü veya koşul ifadesi içinde kullanılabilir ve bunun tersi de geçerlidir.

1. Aşağıdaki kodun çıktısı nedir?

```
public class punchcard {
    public static void main(String[] args) {
        int x = 5;
        while (x > 0) {
            System.out.print(x + " ");
            x -= 2;
        }
    }
}
```

- A) 5 4 3 2 1
- B) 5 3 1
- C) 5 2
- D) 5 2 0
- E) 5 3 1 -1

cevap: B - 5 3 1

2. Aşağıdaki boşluğu dolduracak seçenekler hangileridir, böylece kod 0'dan 10'a kadar olan çift sayıları yazdırır?

Beklenen çıktı:

```
Counting: 0
Counting: 2
Counting: 4
Counting: 6
Counting: 8
```

```
public class punchcard {
    public static void main(String[] args) {
        int i = 0;
        while (_____) {
            System.out.println("Counting: " + i);
            _____;
        }
    }
}
```

- A) (i < 10) ~~ i += 2
- B) (i <= 10) ~~ i += 2
- C) (i < 10) ~~ i ++
- D) (True) ~~ i += 2
- E) (i <= 10) ~~ i ++

cevap: A - (i < 10) ~~ i += 2

3. Kodun Yanlış Olan Kısmını Tanımlayın. Amaç, 10'dan 5'e kadar olan sayıları azalan şekilde yazdırmaktır.

```
public class punchcard {
    public static void main(String[] args) {
        int num = 10;
        while (num < 5) {
            System.out.println(num);
            num--;
        }
    }
}
```

- A) Döngü sürekli olarak çalışır.

- B) Azaltma işlemi `num -= 2`; olmalıdır.
- C) Döngü hiçbir zaman çalışmaz.
- D) Azaltma işlemi `num++`; olmalıdır.
- E) Döngü sadece bir kez çalışır.

cevap: C

4. Aşağıdaki kodun doğru çıktısı nedir?

```
public class punchcard {  
    public static void main(String[] args) {  
        int y = 3;  
        while (y < 10) {  
            System.out.print(y * 2 + " ");  
            y += 3;  
        }  
    }  
}
```

- a) 6 9
- b) 6 12 15 18
- c) 6 12 18
- d) 6 9 12 15 18
- e) hata

cevap: c) 6 12 18

5. Kodun Yanlış Olan Kısımını Tanımlayın.

```
public class punchcard {  
    public static void main(String[] args) {  
        int z = 0;  
        while (z != 7) {  
            System.out.println(z);  
            z += 2;  
        }  
    }  
}
```

- a) Döngü sürekli olarak çalışır.
- b) Döngü içeriği çalışmaz.
- c) Döngü koşulu yanlıştır.
- d) Döngü içeriği bir kez çalışır.
- e) Kodun herhangi bir sorunu yok.

cevap: a) Döngü sürekli olarak çalışır.

6. Eksik Boşluğu Doldurmak İçin Doğru Cevabı Seçin. Amaç, 2'den 1024'e kadar olan 2'nin kuvvetlerini yazdırmaktır.

```
public class punchcard {  
    public static void main(String[] args) {  
        int e = 2;  
        while (e <= 1024) {  
            System.out.println("Değer: " + e);  
            -----;  
        }  
    }  
}
```

- a) `e = e * e;`

b) `e += 2;`
c) `e = Math.pow(e, 2);`
d)
`e = Math.pow(2, e); e *= 2;`
cevap: e) `e *= 2;`

7. Kodun Yanlış Olan Kısmını Tanımlayın.

```
public class punchcard {  
    public static void main(String[] args) {  
  
        int f = 10;  
        while (f != 0) {  
            System.out.println(f);  
            f -= 0;  
        }  
    }  
}
```

- a) Döngü sürekli olarak çalışır.
- b) Döngü içeriği çalışmaz.
- c) Döngü koşulu yanlıştır.
- d) Döngü içeriği bir kez çalışır.
- e) Kodun herhangi bir sorunu yok.

cevap: a) Döngü sürekli olarak çalışır.

8. Aşağıdaki kodun çıktısı nedir?

```
public class punchcard {  
    public static void main(String[] args) {  
        int g = 1;  
        while (g <= 5) {  
            System.out.print(g * g + " ");  
            g++;  
        }  
    }  
}
```

- a) 1 4 9
- b) 1 4 9 16 25
- c) 1 2 3 4 5
- d) 1 3 5 7 9
- e) 1 3 5 9 16 25

cevap: b) 1 4 9 16 25

9. Aşağıdaki kodun çıktısı nedir?

```
public class punchcard {  
    public static void main(String[] args) {  
        int x = 10;  
        while(x > 4) {  
            System.out.print(x + " ");  
            x--; //burada 1 azalt  
            if (x == 5) {  
                //x 5 olduğunda değeri 5 arttır, böylece tüm işlem tekrarlanır  
                x += 5;  
            }  
        }  
    }  
}
```

```
    }  
    }  
}
```

- a) 10 9 8 7 6 5 4 3 2 1
- b) 10 9 8 7 10 9 8 6 ...
- c) 10 9 8 7 6 5
- d) 10 11 12 13 14 ...
- e) hata

cevap: b) 10 9 8 7 10 9 8 6 ...

10. Hızlı!!! Bombayı etkisiz hale getir!!!

```
import java.util.Scanner;  
  
public class punchcard {  
    public static void main(String[] args){  
        int time;  
        Scanner timeScan=new Scanner(System.in);  
        System.out.print("Süreyi ayarla: ");  
        time=timeScan.nextInt();  
        while(time>=0){  
            System.out.println(time);  
            time--;  
        }  
        if (time==--1)  
            System.out.println("BOOOO MMMM");  
        else  
            System.out.println("Bombayı etkisiz hale getirdin!!!");  
    }  
}
```

- a) 123456436754
- b) 0xCAFE BABE
- C) 0
- d) -12
- e) 1

cevap: a) -12 b) 0xCAFE BABE

12. Boşluğu doldurun. Böylece kod, 3'e bölünebilen veya 10'un modu 3 olan sayıları yazdırır.

```
import java.util.Scanner;  
public class Main {  
    public static void main(String[] args) {  
        Scanner read = new Scanner(System.in);  
        int number = read.nextInt();  
        int n=1;  
        while(n<=number){  
            if(n%3==0 __ n%10==3) { // boşluk bu satırda  
                System.out.println(n);  
            }  
            n++;  
        }  
    }  
}
```

- a) ||
- b) &&
- c) |
- d) or
- e) +

cevap: a) ||

13. Döngü kaç kez çalışır?

```
public class Main {  
    public static void main(String[] args) {  
        int number = 5;  
        int x = number/3;  
        int y = number/x;  
        while (y <= number){  
            if (y % 2 == 0){  
                System.out.println(y);  
            }  
            y = y + 1;  
        }  
    }  
}
```

- a) 0
- b) 1
- c) 2
- d) 11
- e) sonsuz

cevap: b) 1