

JAVA PLATFORM BAĞIMSIZLIĞI

JVM (Java Sanal Makina) .java dosyasını derledikten sonra byte code'a çevirir. Çalışan byte code. class dosyası oluşturur. Bu dosyanın tüm işletim sistemlerinde çalışması ise, Java'nın platform bağımsızlığına örnektir. Farklı işletim sisteminde oluşturulan byte code diğer işletim sisteminde de çalışır.

JAVA'DA HEAP ve STACK

Bellekte 2 alan vardır. stack ve heap diyebiliriz.

Heap'in kelime anlamı öbek, yığın, kalabalık anlamına gelmektedir. Nesneler, sınıflar bu bellekte tutulur. Referans ipler heapte tutulurken değerleri stackte tutulur buna örnek olarak array (list)referans tipi vardır

STRING CLASS'I NASIL İMMUTABLE OLMAYI SAĞLAMAKTADIR

Immutable”ın kelime anlamı sabit, değişmez demektir. String class da immutable bir yapıdır. Oluşturulduktan sonra sabit kalırlar.

- Değişkenler private olarak tanımlanmalı. Private özel anlamına gelir dışardan bu değişkenlere erişilemez.
ÖRN; private String a ;
- Değişkenlerin getter ve setter metodları vardır; getterlar okunma işlemini yapar setterlar ise tanımlanan değişkenin değerinin üzerinde değişiklik yapma işlemini yapar.

```
ÖRN; public String getA() {  
        return a;  
}  
  
public void setA(String a) { // Bu kod bloğunu kullanmayız.  
    this.a = a;  
}
```

JAVA NEDEN ÇOKLU KALITIMI DESTEKLEMEZ

Inheritance yani kalıtım sub classın süper classtan miras alma yöntemi diyebiliriz. Süper classlara anne baba ebevyn class da diyebiliriz bundandır ki her çocuğun bir annesi vardır. Dolayısıyla multiple değil tek bir inheritance vardır.

BUILD TOOL NEDİR? JAVA EKOSİSTEMİNDEKİ BUILD TOOLLAR NELERDİR?

Build tool inşa, build etme derleme diyebiliriz. Java tabanlı bir projemizi derlemek için maven aracını kullanabiliriz.

Bazı derleme araçları;

1. Jenkins
2. Apache Ant
3. Gradle
4. Maven

COLLECTIONS FRAMEWORK

Veri tipi ve eleman sayısı, boyutu belirsiz ise java collection interfacerlerinden yararlanabiliriz.

- **List** : Sıralı liste
 - Vector, ArrayList, LinkedList

```
List<Integer> sayilar = new ArrayList<>(); // boyutu belli değil
```

```
sayilar.add(23);  
sayilar.add(234);  
sayilar.add(44);
```

- **SET** : Kümelerdir. Sıralı olmasına gerek yoktur

HashSet, TreeSet

```
Set<String> benzersizIsimler = new HashSet<>();
```

```
benzersizIsimler.add("Ayşe");  
benzersizIsimler.add("Merve");
```

- **QUEUE**: Listeye eklenme sırası ile listeler

PriorityQueue, LinkedList

- **MAP** : Bir collections değildir. Anahtar ve değerlerden oluşur.

```
Map<Integer, String> anahtarlar = new HashMap<>();
```

```
anahtarlar.put(1, "Ayşe");  
anahtarlar.put(2, "Merve");  
anahtarlar.put(3, "İlker");  
anahtarlar.put(4, "Doğan");
```

