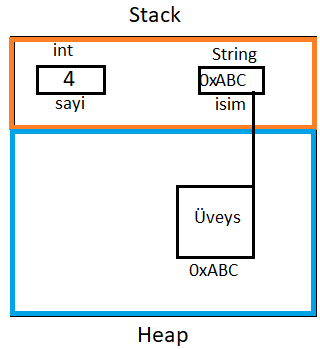
**Java’nın platform bağımsızlığını nasıl sağladığını anlatınız.**

Java kodu not defteri gib text editörlerde yazılabilir. Fakat biz daha kolay yazabilmek için IDE (Eclipse, intelij idea, vb.) indiririz. Burada kodu yazdıktan sonra çalıştırmaya denediğimizde hata verecektir. Bunun nedeni JDK olmamasıdır. JDK yani Java Development Kit bir compiler’dır. JDK içinde JRE bulundurur. JRE ise içinde JVM bulundurur. Ve Java platform bağımsızlına JVM sayesinde sahiptir.

Yazdığımız Java kodları ilk önce javac tarafından byte code’a çevrilir. Bu byte code’lar .class uzantılı dosyalardır. JVM bizim yazdığımız java kodlarını anlamaz o yüzden byte code’lu .class uzantılı dosyayı okur. Ve bizim bilgisayarımızda yazdığımız kod byte code’a çevrildikten sonra bytecode’lar diğer platformlarda kurulu olan JVM sayesinde tüm platformlarda çalışır hale gelir.

**Java’da heap ve stack kavramlarını örneklerle açıklayın.**

Tipler ikiye ayrılır.

1. Referans Tipliler (String, Class, Array)
2. İlkel/Değer tipliler (byte, short, int, long, double, float)

Yandaki şekilde görüldüğü gibi ilkel tipli olanlar direkt olarak **Stack**’te saklanır.

Fakat referans tipli değişkenler için durum böyle değildir. Örneğin yandaki şekilde String bir değer var. String değer referans tipli olduğu için **Stack**’teki alanın içerinsinde **Heap**’teki değerinin adresi bulunur.

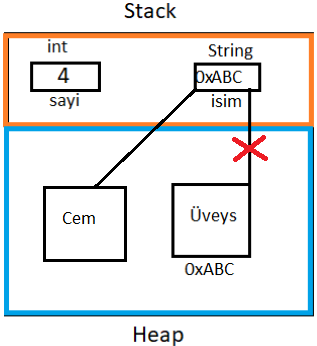
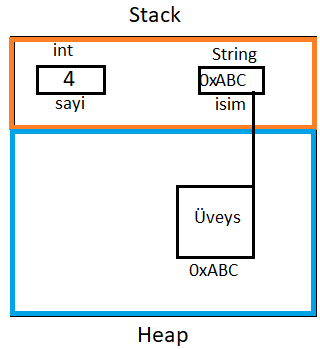
Yani String tipinin değerine ulaşmak için **Stack** içindeki **Heap** adres değerine ulaşılır ve **Heap** adresinden de String içeriğine ulaşılır.

**String class’ı nasıl immutable olmayı sağlamaktadır örnek ve çizimlerle açıklayınız.**

Şekilde görüldüğü gibi String’de aslında bir nesnedir. Ve içinde herhangi set metodu olmadığı için değiştirlemez.

Peki a Stringi’in içeriğini nasıl değiştiriyoruz?

Aşağıda şekilde bir stringin Heapteki yeri gösterilmiştir. a=”Cem” dersek heapteki eski String ile bağlantısı kopacak artık içeriği yeni eşitlediğimiz String olacaktır.

****

→

**Build Tool nedir? Java ekosistemindeki build toolar neler açıklayın?**

Build tool’lar projeyi yönetmek, organize etmek için kullanılan tool’lardır. Özellikle çok sayıda projenin olduğu ortamlarda ve birbirlerine bağlıysalar build tool kullanmak önemlidir.

Java özelinde en çok bilinen Gradle ve Maven’dır. Gradle daha çok kotlin projelerinde kullanılır.

Maven, projeyi parçalara ayırmaya ve daha kolay yönetebilmemize olanak tanır.

pom.xml dosyasına eklenen bağımlılıklar ile kolay bir şekilde paketleri indirmeyi ve proje eklemeyi sağlar.

ÜVEYS SARAÇ