

HW_6_SONUC_RAPORU

MÜH.TĞM. MUHAMMET YURTSEVEN**Anlaşılmayan :** Soru12,Soru15,Soru21,Soru24,Soru30,Soru34,Soru45,Soru50

Soru 1- Cevap C :

Private , Bazı değişken, metot ya da sınıflara başka sınıftaki kodların erişmesini engellemek isteyebiliriz. Bunun için private nitelemesini kullanırız. private erişim belirtkisi, public belirtkesinin karşıtı gibidir. private damgalı öğelere yalnız aynı sınıftaki kodlar erişebilir, başka sınıftaki kodlar erişemez. Kendi alt-sınıfları bile erişemez. Bir alt-sınıf, atasının public ve ön-tanımlı öğelerine erişebilir, ama private öğelerine erişemez.

Protected , Bir sınıf içindeki değişken ve metotlara alt-sınıfların erişebilmesini, ama paket içindeki ya da program içindeki başka kodların erişmesini engellemek isteyebiliriz. Bunun için sözkonusu öğeyi, protected damgası ile nitelemek yetecektir. Bu demektir ki, alt-sınıf, üst-sınıfın protected damgalı öğelerine sanki public öğelermiş gibi erişir.

<http://mail.baskent.edu.tr/~tkaracay/etudio/ders/prg/java/ch11/accessmods.htm>

Soru 2 Cevap B : super() ifadesi, üst sınıftaki bir yapıcıcıyı çağırırken kullanılırken, this() ifadesi aynı sınıftaki bir yapıcıcıyı çağırmak için kullanılır ve Seçenek B'yi doğru ve Seçenek A'yı yanlış yapar. C ve D seçenekleri yanlıştır çünkü bunlar yapıcıcı değildir.

Soru 3- Cevap D

Soru 4 - Cevap D

Soru 5 - Cevap B : Java, primitives ve nesnelerin referanslarını bir yonteme göndermek için pass-by-value kullanır. <https://anilemreozcelik.wordpress.com/2015/07/20/45/>

Soru 6 – Cevap C : Seçenek C doğru cevaptır. Çünkü doğru bir ayarlayıcı bildirimidir, çünkü bir değer alır, geçersiz dönüş türünü kullanır ve doğru adlandırma kuralını kullanır. A seçeneği yanlıştır çünkü alıcı bir değer döndürmelidir. Ayarlayıcı bir değer alması gerektiğinden B seçeneği yanlıştır. Seçenek D yanlıştır, çünkü ayarlayıcı set ile başlamalı ve bir değer döndürmemelidir.

Soru 7 – Cevap B: Seçenek A,C,D doğrudur. B seçeneği yanlıştır; çünkü bir kurucu yalnızca kurucunun ilk satırında this () veya super () olarak çağırabilir, ancak hiçbir zaman aynı kurucuda olamaz.

Soru 8 – Cevap B : Erişimler küçük harfle başlamaz A seçeneği yanlıştır bu yüzden. C ve D seçenekleri ise 10 değeri ile uyumlu olmadığından yanlıştır. Long doğru cevaptır.

Soru 9- Cevap C : Sınıfın tüm örnekleri için her zaman kullanılabilen tek değişkenler statik olarak bildirilen değişkenlerdir; Bu nedenle, Seçenek C doğru cevaptır. Seçenek A doğru olmama sebebi, ancak genel değişkenler yalnızca nesneye yapılan bir başvuru tüm örnekler arasında korunursa kullanılabilir. Java'da local anahtar kelime olmadığı için B seçeneği yanlıştır. Seçenek D de yanlıştır, çünkü instance değişkenine yalnızca onu oluşturan örnek içinde erişilebilir.

Soru 10- Cevap A

Soru 11- Cevap B : A, C ve D seçenekleri doğrudur. Özellikle, Seçenek C, karşılaştırılan iki nesne arasında equals () yöntemlerini yazmamızı sağlar. D seçeneği doğrudur çünkü korumalı Access paket-özel erişim de sağlar. Seçenek B yanlıştır. Paket öznitelikleri, yalnızca iki sınıf arasında genişletilip genişletilmediğine bakılmaksızın aynı pakette görünürse görünür.

Soru 12- Cevap

Soru 13- Cevap C: Doğru Cevap C'dir. Seçenek A yanlıştır, çünkü Java yalnızca sınıfta başka kurucu yoksa yalnızca bağımsız değişken yapıcı ekler. Seçenek B, derleyici tarafından eklenen ve alt sınıfta erişilebilen varsayılan bir bağımsız değişken yapıcıya sahip olabileceğinden, B seçeneği yanlıştır. Son olarak, Seçenek D yanlıştır. İki bağımsız değişken, oluşturucu içermeyen bir sınıf aynı adı taşıyacağından derlenmez. Son olarak, Seçenek C doğrudur. Sınıf ,bağımsız değişken yapıcı içermeyen bir üst sınıfı genişletirse, varsayılan bağımsız değişken

yapıcı derleyici tarafından alt sınıfa otomatik olarak eklenemez. Bunun yerine, geliştirici en az bir kurucuyu açıkça bildirmeli ve açık kurucuya çağrının nasıl yapıldığını açıkça tanımlar.

Soru 14 Cevap A : Bir yöntem en fazla bir varargs parametresi içerebilir ve listede son argüman olarak görünmelidir.

Soru 15- Cevap

Soru 16- Cevap B : Seçenekler A ve D sınıfın derlenmesine izin vermez, çünkü sınıftaki iki yöntem aynı ada ve bağımsız değişkenlere sahip olamaz. C , sınıfın derlenmesine izin verir, ancak farklı bir yöntem adı kullandığı için olmaz. Seçenek B, findAverage () yönteminin geçerli bir aşırı yüklenmiş sürümüdür, çünkü ad aynıdır ancak değişken listesi farklıdır.

Soru 17- Cevap D : A,B,C doğrudur. Encapsulation performans ve eşzamanlılık konusunda hiçbir garanti vermez.

Soru 18- Cevap A : Seçenek B yanlıştır, çünkü String değerler değiştirilemez. C ve D seçenekleri de değişkenler Java'da referansa değil değere iletildiğinden yanlıştır. Seçenek A doğru cevaptır. Nesneye yapılan başvurunun bir kopyası geçirildiği için, bir dizinin içeriği bir yöntemle geçirildiğinde değiştirilebilir.

Soru 19- Cevap B : Static olsaydı C doğru olurdu. Ama soruda bunu istemiyor. Bize en iyi yöntemi sormaktadır. Bu nedenle seçenek Bdir. Çünkü iki sınıfın aynı pakette olduğu göz önüne alındığında, paket adı gerekli olmayacaktır.

Soru 20- Cevap D : A ve B seçenekleri return kullanarak bir değer döndürmesini gerektirir. Void seçeneğinde () bunu gerektirecektir. Bu neden Cevap D.

Soru 21- Cevap

Soru 22- Cevap D: super (), üst sınıftaki bir kurucuyu çağırırken, super ise üst sınıfın bir üyesine başvurmak için kullanılır. this (), mevcut sınıfın bir üyesine başvuruda bulunmak için kullanılır. Bu nedenlerle, Seçenek D doğru cevaptır.

Soru 23- Cevap B: Aynı paketteki sınıflar tarafından erişilebilir, bu da Seçenek B'yi doğru cevap haline getirir.

Soru 24- Cevap

Soru 25- Cevap A: Seçenek adları alt çizgi içerebileceğinden doğru seçenek A'dır.

Soru 26- Cevap D

Soru 27- Cevap B: Seçenek B doğru cevaptır. Bir nesnenin içindeki verilerde yapılan değişiklikler, kopyalanan başvurunun işaret ettiği nesne aynı olduğundan çağırılan yöntem tarafından görülebilir.

Soru 28- Cevap C

Soru 29- Cevap A: JavaBean yöntemleri get,set, ve is boolean değerleri kullanır.

Soru 30- Cevap

Soru 31- Cevap D: Java'da, bir erişim değiştiricisinin olmaması, üyenin package-private olduğunu gösterir, bu nedenle Seçenek D doğrudur.

Soru 32- Cevap B

Soru 33- Cevap A: Bir instance veya constructurun tüm static değişkenlere erişimi vardır. statik değişkenler bildirildiklerinde veya başlatma bloğunda ayarlanmalıdır. Öte yandan, statik yöntemler ve statik başlatıcılar, tüm örneklerde tanımlandıkları için örnek değişkenleri referans edemezler.

Soru 34- Cevap

Soru 35- Cevap C : Overloaded yöntemler aynı ada ancak farklı bir parametre listesine sahip olduğundan birinci ve üçüncü ifadeleri doğru yapar. Overloaded yöntemler aynı veya farklı dönüş türlerine sahip olabileceğinden, ikinci ifade yanlıştır. Bu nedenle, Seçenek C doğru cevaptır.

Soru 36- Cevap C

Soru 37- Cevap D

Soru 38- Cevap A: public, private doğru cevaptır. Public değiştiricisi her şeye erişime izin verir. Private ise yalnızca aynı sınıftaki üyelere erişime izin verir.

Soru 39- Cevap A

Soru 40- Cevap B : A ve D seçenekleri eşdeğerdir ve kodun derlenmesine izin verir. Her ikisi de, bir instance yönteminden statik bir yönteme erişmek için uygun yöntemlerdir. Seçenek B doğru cevaptır. Sınıf derlenmez çünkü this.drink'in derleyiciye bir anlamı yoktur. Seçenek C, kodun derlenmesine izin verir. Statik üyelere statik bir şekilde erişilmesine rağmen gerekli değildir.

Soru 41- Cevap C : 3 tane değer gerekiyor. İlk değer int sonraki değerler string olacak şekilde en uygun C seçeneğidir. call(2, "home", "sweet")

Soru 42- Cevap D : Statik değişkene sınıfın tüm instanceları tarafından erişilebilir.

Soru 43- Cevap A : Bir kurucunun ilk satırı this () veya super () olabilir.

Soru 44- Cevap D

Soru 45- Cevap

Soru 46- Cevap A

Soru 47- Cevap C

Soru 48- Cevap A

Soru 49- Cevap B: The Protected, aynı pakette bulunan herhangi bir alt sınıf veya sınıf tarafından erişime izin verir.

Soru 50- Cevap