

HomeWork_1_Sonuç_Raporu

TURGUT YILDIZ

18.04.2020

4, 10, 11, 12, 18, 21, 22, 28, 32, 33, 37, 43, 44, 45, 46, 48 numaralı 16 sorunun java uygulaması yapılmıştır.

16, 24, 29 numaralı sorular için cevap bulunamamıştır.

CEVAPLAR

1. Java uygulaması public static erişimi olan, void dönüşü ile String[] argümanı olan main() metodundan oluşur.
A: static olmalıdır. (static)
B: String[] argümanı olmalıdır. (String[] args)
C: yanlış erişim ve metod ismi kullanılmıştır. (public, main(String[] mydata))
CEVAP : D
2. **A :** Diyagram javada platform bağımsızlığını göstermez. **X**
B : Diyagram Object-oriented programlamayı göstermektedir. ✓
C : Gold ve Silver classları weight ve color değerlerini Metal classından kalıtır. ✓
D : Gold luster değerini kalıtamaz. ✓
CEVAP : A
3. Javanın derlenmiş bytecode dosyası “.class”tır
CEVAP : C
4. **A :** Satır 1 ve 2 derlenir.
B : Satır 4 derlenmez (Q4 uygulaması) çünkü Date() metodunu hangi paketten alacağı belli değildir.
C : satır 5 derlenir
D : Kod 4. Satırdan dolayı derlenmez.
CEVAP : B
5. Object-oriented programlama dili veriyi classta yoplayarak nesneler üzerinden işlemlerini yapmaktadır.
CEVAP : D

Soruda nesne yönelimli programlama dillerinin bir özelliği olmayan şık aranmıştır. A şıkında ayrı(seperate) olarak nesnelerin tutulmasından bahsedilmiş burası single olarak tek nesnede gruplama eğiliminde olmalıdır.

CEAVAP : A

6. Lokal deęişkenler tanımlandığı scope alanları kadar ömrü vardır. Class, Interface ve Instance deęişkenler daha geniş alanlara sahiptir.

CEVAP : D

7. Temel java sınıflarında java.lang paketini kullanır. Bu pakette deęişken tipleri, erişim denetleyiciler gibi temel fonksiyonlar vardır.

CEVAP: B

8. Javada `/**/` ve `//` ifadeleri yorum satırı olarak tanımlanırken Hashtag(`#`) javada geçerli bir yorum satırı belirtmez.

CEVAP: C

9. .java dosyasında istenildiği kadar class ve interface tanımlanabilir. Sadece 1 tane public class tanımlanabilir.

CEVAP: D

10. main() metodu static olmadığından hem class hemde static instance deęişkenlere erişebilir. 2 tane sınıf, 2 tanede static instance deęişken ve 1 tanede yerel deęişken olduğundan

CEVAP: B (2,2,1)

11. Q11 uygulamasında olduğu gibi kullanılmayan import ifadeleri olması ve birden fazla aynı ifadenin import edilmesi derleme işlemini bozmaz. Fakat import edilen sınıfın deęişkeni veya metodu kullanılıyorsa bu importun kaldırılması derlemeyi bozar.

CEVAP: B

12. Q12 uygulamasında gösterildiği gibi bird deęişkeni ya static yapılarak yada Q12 sınıfından bir nesne oluşturularak kullanılabilir. Bu hali ile derlenmez.

CEVAP :A

13. I – Java yalnızca derlenmiş .class dosyalarını çalıştırabilir. **X**

II – Java Object-Oriented yapıdadır.

X

III – Java sanal makine(JVM)'de çalışan bir kod olduğundan makine kodu olmayan bayt kodda derlenir. **X**

CEVAP: D

14. .java dosyasının başlangıcına import ile sınıf tanımlaması, package ile paket ekleme işlemi ve yorum satırı yapılabilir. Fakat bir scope'a ait olmayan deęişken tanımlanamamaktadır.

CEVAP: D

15. A: Sınıflar bir package içinde olmadanda oluşturulabilirler. **X**
B: package oluşturmak için package.init diye bir dosya eklemeye gerek yoktur. **X**
C: package oluşturduğumuzda başka package veya classlardan o paket altındaki değişken veya metodlara erişim sağlanması kısıtlanabilir. **✓**
D: Bir package içinde nesne ve metodlara erişim kısıtlanabilir. **X**
CEVAP : C

16. ---

javac bir java kodunu derleyebilmesi için path'inin tam verilmesi ve .java dosya uzantısı ile dosya adını alarak dosya ile uyumlu çalıştırılabilir kodu üretir.

javac Manager.java

java Manager

CEVAP : B

17. **Platform Independence** : JVM yani Java Virtual Machine, Java dilinde yazılan programların her platformda (farklı işlemci ya da işletim sistemlerinde) çalışabilmesi için geliştirilmiş olan, fiziksel bir bilgisayar gibi program çalıştırabilen bir sanal makinedir.

Object Orientation : Mantıksal işlemlerden ziyade, **nesnelere (object)** ve **nesneler** üzerinde işlemlere odaklanan programlama dili modelidir.

Inheritance : Bir sınıfın başka miras aldığı başka bir sınıfın özelliklerini de kullanmasına denir.

Encapsulation : Class dışından ya da yanlış kullanım sonucunda kodu erişimi koruyan bir mekanizmadır.

CEVAP : D

18. Height değişkeni if bloğu içinde oluşturulduğundan bu alan dışından bu değişkene ulaşılamaz. Bu şekilde derlenemez. Derlenebilecek çözüm için Q18 uygulamasına bakılmalıdır.

CEVAP : D

19. Java bytecode JVM yüklü her makinede çalışabilmesi için tasarlanmıştır. Buda javayı platform bağımsız duruma getiri.

Java kodunun bir makinede çalışması için .java kaynak koduna ihtiyaç yoktur .class dosyaları ile bu sanal makina çalışmaktadır. Bu .class dosyaları metin düzenleyici ile açıldığında makina koduna erişilir ve buda kolayca değiştirilmesine imkan sağlamaz.

CEVAP : A

20. Javada bir ifadeyi sonlandırmak için (;) ifadesi kullanılır.

CEVAP : D

21. Static olmayan değişkenler o sınıfa ait bir nesne oluşturularak ilgili metodlarda kullanılabilir. Static değişkenler hem nesne oluşturup hem de doğrudan kullanılabilir.
Yerel değişkenler sadece tanımlandığı scope alanlarda kullanılır. Q21 uygulamasına bakıldığında sonuç = 31 olur.

CEVAP : C

22. 1. satırda **class** tanımlaması eksiktir. **X**
2. satırda 2 tane değişken tipi tanımlaması yapılmıştır. **X**
3. satırda doğru bir metod tanımlaması yapılmıştır. **✓**
4. satırda void değişkenlerde private erişim engelleyicisi kullanılmamalıdır. Bu satırda limit değişken yapılırsa void ve private ifadesi silinmelidir. Metod yapılacaksa private ifadesi silinip “**static void** limit() {}” şeklinde yazılabilir. **X**

CEVAP : C

23. Java'nın en önemli özelliğinden biride platform bağımsız olmasıdır. JVM sayesinde çeşitli aygıt ve bilgisayarlarda çalışabilir.

CEVAP : D

24. JVM: Platform bağımsızlığını destekler. Java kodlarını makine kodlarına çevirir. Uygulama için makine belleğini yönetebilir. Decode/Encode işlemlerini sadece JVM'de gerçekleştirir.

CEVAP : ---

Java bytecode makine dilinde olduğundan kolayca decoded/decompiled edilememektedir.

CEVAP : A

25. Paket değişkeni diye bir şey yoktur. . **X**
Class değişkenleri programın herhangi bir noktasından erişim sağlanabileceği için program dahilinde geçerlidir. **✓**
Instance değişken sınıfın belli bir örneğinde geçerlidir. . **X**
Yerel değişkenler tanımlandığı alanda geçerlidir. . **X**

CEVAP : B

26. import television.actor.*;
import movie.director.*;
Yukarıdaki ifadede (*) belirteci o paketin altındaki tüm alt paketleri eklemeye yarar.
television.actor.recurring.Marie **X** // tek bir paket eklendiğini belirtmiş
movie.directors.John **X** // Hem tek paket eklenmiş hem de director olması gerekir
television.actor.Package **✓** // Tüm alt paketleri eklemiş DOĞRU
movie.NewRelease **X** // Başka bir alt paket eklemesi yapılmıştır.

CEVAP : C

27. Java'da önce package, import daha sonrada class bildirimi yapılır.

CEVAP : D

28. Standart her java uygulamasında arka planda import edilen lang paketlerini kaldırabiliriz. Stars.* ve stars.Blackhole paketlerinden de 1 tanesini kullanarak kodu derleyebiliriz.

stars.* tüm alt paketleri koda eklerken, stars.Blackhole bir tane alt paket koda eklemektedir.

CEVAP : D

29. ---

theInput[2] dizinin 3. Elemanını ekrana yazdırır.

java forest.Deer çalıştırıldığında. Şıkılardaki 3. elemanlar aşağıdadır.

A: "White-tailed deer"

B: 3

C: White-tailed

D: NULL

CEVAP : C

30. **javac** : JDK içinde bulunan Java ile yazdığımız kodları derleyen Java derleyicisidir. Java dil kurallarına uygun şekilde yazılmış olan kodları(.java) derleyerek bytecode(.class) halinde çevirir. Böylelikle kodlarımız JVM'de çalışır hale gelir.

CEVAP : B

31. A. Java nesne yönelimli programlama dilidir.

B. Metodların overloading olmasına olanak sağlar.

C. Operatörlerin aşırı yüklenmesine olanak sağlamaz.

D. JVM bellekte değişken yerlerde çalıştığından bellekteki nesnelere doğrudan erişim sağlayamaz.

CEVAP : B

32. Q32 uygulamasında gerçeklediğim gibi 1.)package 2.)class 3.)int(return ifadesi olduğundan dönüş değeride integer olduğundan metodun dönüş değeri int veya long olacaktır.)

CEVAP : D

33. Q33 uygulamasında olduğu gibi; main() metodunda Q33 sınıfı oluşturulurken kurucu metoduna 10 ve fly() metoduna 5 değeri gönderilmektedir. Kurucu metodun sonunda end değeri 4 oluyor ve fly metodu end,start ve main() metodundan gelen 5 değeri ile ekrana { 2 5 } ifadesini çıkarıyor.

CEVAP : A

34. Yazılımcılar inheritance(kalıtım) kullanımını oluşturulacak yeni classların daha az kod ile daha hızlı ve karmaşık olmadan geliştirilmesi için kullanılmaktadır.

CEVAP : D

35. Javada yorum satırı (// ve /**/) ifadeleri ile yapılır. Bu ifadelerden sonra istenilen şekil ve yorum istendiği kadar kendini tekrar edebilir. (///////////////yorum_satırı //) gibi

CEVAP : A

36. Java main() metodlar **public static void** main(String[] args) şeklinde genel olarak tanımlanır ve **final** olarakta ifade edilebilir. main() metodunda String değişkenini ifade etmek için String... ve String[] şeklinde tanımlama yapmalıyız.

CEVAP : B

37. **public String color;** değişkeni class dışında tanımlanamaz ve dönüş tipi olmayan (void) metodlarda kullanılması hata vereceğinden a2 ve a4 e eklenebilir. (Q37 uygulamasında belirtilmiştir.)

CEVAP : B

38. Java sınıf dosyası tanımlamak için .java dosyasında class yapısı bulunmalıdır.

CEVAP : A

39. Java kaynak kodları .java dosyasında bulunur. .class dosyasında makina kodları bulunmaktadır.

CEVAP : D

40. java.lang ve pocket.complex içinde Math kütüphanesi olması programa bu paketlerin import edilmesini engellemez. Fakat Math() metodu kullanılacaksa hangi paketten çekileceği belirtilmelidir. Örneğin (complex.Math) gibi. Derleme işlemi Math metodunun kullanılacağı satırda hata verecektir.

CEVAP : C

41. import dog.puppy* paketi ile bir alt paketlere erişebiliriz. dog.puppy.female.KC ifadesi 2 alt paket olduğu için erişemeyiz. Ve java.lang.* ifadesi tüm java uygulamalarına import edildiği için java.lang.Object ifadesine ulaşılabilir.

CEVAP : A

42. Verileri nesneler halinde yöneten ve yapılandıran yönteme Object-Oriented programlama denmektedir.

CEVAP : B

43. Q43 uygulamasına bakıldığında tüm import ifadeleri kullanılmaktadır.

CEVAP : A

44. Q44 uygulamasındaki gibi numLock ifadesine static yapılmadan veya Q44 sınıfının bir nesnesi oluşturulmadan erişilemez.

CEVAP : C

45. Q45 uygulamasındaki gibi 2 tane yerel değişken ve 1 tanede Q45 sınıfının değişkeni toplanarak cevap 20 bulunur.

CEVAP : D

46. Q46 uygulamasındaki gibi `printColor("blue")` ifadesi ile metod blue string ifadesi ile çağırılır. `color = "purple";` ile blue değeri purple olur ve ekrana çıkarılır.

CEVAP : B

47. Java paketleri ayırmak için (.) kullanır. (java.lang)
javac .java source kodu .class makine koduna dönüştürür.

CEVAP : C

48. Q48 uygulamasındaki gibi program derlenir. Dile ait bir sorun yoktur fakat main() metodu giriş argümanı almadığından çalıştırıldığında runtime hatası verir.

CEVAP : D

49. Tabloda **Book** sınıfının **numberOfPages** değişkeni ve **getRating()** metodu olduğu görülmektedir.

CEVAP : C

50. **A** : JVM garbage collector çalışması için belli bir zamanlama yoktur.

B : Sonsuz döngüye giren bir uygulamayı JVM yerine işletim sistemi de sonlandırabilir.

C : JVM uygulamayı başlatabilmesi için main() metodunun olduğu yere erişmesi gerekmektedir.

D : JVM olmadan herhangi bir makinada java kodu çalışmaz.

CEVAP : C