

1. Cevap: D →Doğru

Açıklama: Java uygulamasın bir giriş noktası, tek bir String [] argümanına sahip bir main () methodundan, void dönüş türünden ve public ve staticten oluşur.

2. Cevap: D →Yanlış(A)

Açıklama: Platform bağımsızlığının diyagramla ilgisi yoktur.

3. Cevap: C →Doğru

Açıklama: Java tarafından derlenmiş bir bayt kodu dosyası için uygun uzantı .class

4. Cevap: B →Doğru

Açıklama: Satır 4 te kullanılan Date sınıfı belirsiz.

5. Cevap: A →Doğru

Açıklama: Nesne yönelimli bir proje verileri ve bu verilerle ilgili eylemleri tek bir nesnede grupta eğiliminde olduğundan A Seçeneği yanlıştır.

6. Cevap: D →Doğru

Açıklama: Yalnızca local değişkenlerin böyle bir kapsamı vardır

7. Cevap: B →Doğru

Açıklama: Java.lang paketi her Java sınıfına alınır. C ve D seçenekleri standart Java çalışma zamanında mevcut değildir.

8. Cevap: C →Doğru

Açıklama: Java, seçenek A, B ve D'yi geçerli yorumlar olarak kabul eder.

9. Cevap: D →Doğru

Açıklama: Geçerli bir .java dosyası, herhangi bir sayıda sınıf veya interfaces tanımlayabilir, ancak en fazla bir ortak sınıfa sahip olabilir. Ayrıca herhangi bir ortak sınıf tanımlayamaz.

10. Cevap: B →Doğru

Açıklama: İki sınıf değişkeni ve ikide instance tanımlandığından, Seçenek B doğru yanıtıdır

11.Cevap: B →Doğru

12.Cevap: A →Doğru

Açıklama: Kod, 5. satır nedeniyle derlenmez

13.Cevap: D →Doğru

Açıklama: Java komutu yalnızca derlenmiş .class dosyalarını çalıştırabilir.

Javac komutu, Java sanal makinesinde (JVM) çalıştırılması gereken ve yerel makine kodu olmayan bayt koduna derlenir, bu nedenle III de yanlıştır.

14.Cevap: D →Doğru

Açıklama: Sınıf Değişken tanımla başlayamaz

15.Cevap: C →Doğru

Açıklama: C seçeneği, uygulamamızı paketler halinde düzenlemenin başlıca nedenlerinden biri olduğu için doğrudur.

16.Cevap: B →Doğru

Açıklama: B seçeneği derleme ve çalıştırma komutları kümesidir.

17.Cevap: D →Doğru

Açıklama: Bu özellik, dilin nesne kullanıcısından gereksiz uygulama ayrıntılarını saklar. Oluşturulan bir class içerisinde kullanıcının işlemlerini daha kolay gerçekleştirebilmesi için bazı işlemler birleştirilerek tek bir işlem gibi gösterilir. Bu birleştirme işlemine kapsülleme denir.

18.Cevap: D →Doğru

Açıklama: Height değişken sadece if bloğunun içerisinde tanımlanmış.

19.Cevap: A →Doğru

Açıklama: Java bayt kodu dosyası, uyumlu bir JVM'ye sahip herhangi bir bilgisayarda çalışmak üzere tasarlanmıştır.

20.Cevap: D →Doğru

21.Cevap: C →Doğru

22.Cevap: C →Doğru

23.Cevap: D →Doğru

24.Cevap: A →Doğru

25.Cevap: B →Doğru

26.Cevap: C →Doğru

Açıklama: Soruda verilen import seçeneğiyle eklenebilecek sınıf sadece C seçeneğinde mevcuttur.

27.Cevap: D →Doğru

Açıklama: Java sınıfları şu sırayla tanımlanır: paket ifadesi, import ifadeleri, sınıf bildirimi. Ayrıca bu ifadelerin hepsinin gerekli değildir.

28.Cevap: C →Yanlış(D)

Açıklama: Java.lang her Java sınıfında otomatik olarak import edildiğinden, java.lang.* ve java.lang.Object import ifadelerinin her ikisi de gerekli değildir. Bu yüzden 2 tane import ifadesi silinebilir.

29.Cevap: B →Yanlış(C)

30.Cevap: B →Doğru

Açıklama: Javac komutu, bir .java dosyasını bir .class bayt kodu dosyasına derler.

31.Cevap: B →Doğru

Açıklama: Java alt sınıflarda overloading yapmamıza olanak sağlar.

32.Cevap: D →Doğru

33.Cevap: A →Doğru

34.Cevap: D →Doğru

Açıklama: Java'nın mirası desteklemesinin en önemli nedenlerinden biride sınıflar arasında kodların yeniden kullanılmasına izin vermek.

35.Cevap: A →Doğru

36.Cevap: B →Doğru

Açıklama: B seçeneği, bir diziyi bağımsız değişken olarak almadığı için tek geçersiz bildirimdir.

37. Cevap: →(B)

38.Cevap: A →Doğru

39.Cevap: D →Doğru

40.Cevap: C →Doğru

41.Cevap: A →Doğru

42.Cevap: B →Doğru

43.Cevap: A →Doğru

Açıklama: Bu sınıftaki tüm import ifadeleri gereklidir.

44.Cevap: C →Doğru

Açıklama: Numlock değişkenine erişilemediği için kod derlenmez

45.Cevap: D →Doğru

46.Cevap: B →Doğru

47.Cevap: C →Doğru

Açıklama: Paketleri ayırmak için nokta (.) Kullanır, eğik çizgi (/) değil

48.Cevap: D →Doğru

49.Cevap: C →Doğru

50. →(C)