

HomeWork_2_Sonuç_Raporu

TURGUT YILDIZ

24.04.2020

CEVAPLAR

1. Aynı satırda farklı değişken bildirimleri yapılamaz.

CEVAP : A

2. Q2 uygulamasında olduğu gibi değişken oluşturulduktan sonra eğer bir değer atanmazsa o değişken sadece değer ataması yapılması için çağırılır başka bir işlem için çağırılamaz.

CEVAP : D

3. Q2 uygulamasında String değer ataması yapılmazsa “null” olacağı belirtilmiştir.

CEVAP : B

4. Değişken isimleri sayı ile başlayamaz. Sayı ilk harften sonra gelebilir. “_” ve “\$” işaretleri değişkenin başına gelebilir.

CEVAP : B

5. Javada standart sınıf isimlendirmesi = “FooBar”
Javada standart metod ve değişken isimlendirmesi = “fooBar”
Javada standart sabit değer isimlendirmesi = “FOO_BAR”

CEVAP : B

6. Primitiv değişken isimlendirmesi ile instance metodlar çağırılamaz. Bunun için Wrapper (Kapsayıcı) sınıflar kullanılır.

Link : <https://yazdoldur.com/programlama/java/degisken-nedir-primitive-wrapper-class-kavramlari/>

CEVAP : C

7. 2 sayı arasında “_” kullanımı yapılabilir. Fakat değerın başında veya sonunda kullanılamaz.

Örneğin = `int num = 9_9_9;` ekran çıktısı = 999 olacaktır.

CEVAP : C

8. Javadaki wrapper classlar aşağıdaki gibidir.

Primitive Data Type	Wrapper Class
char	Character
byte	Byte
short	Short
long	Integer
float	Float
double	Double
boolean	Boolean

CEVAP : C

9. Q9 uygulamasındaki gibi “integer” isminde bir Wrapper class yoktur. Bu sınıf ismi “Integer” olarak değiştirilirse kod derlenir.

CEVAP : C

10. **new** anahtarı ile o classtan yeni bir object oluşturulur.

CEVAP : C

11. Kayan noktalı gösterim için “double” ve “float” değişkenleri kullanılır. Double değişkenlerde virgüllü sayıyı belirtmek için “f” kullanılmasına gerek yokken float değişkenlerde kesinlikle “f” gösterimi olmalıdır.

CEVAP : D

12. byte = 1 byte
char = 2 byte
float = 4 byte
double = 8 byte
bigint = 8 byte = fakat primitiv değişken değildir.

byte – char – float - double

CEVAP : A

13. –

CEVAP : D

14. 2. satırda 2 farklı değişken tanımlaması yapılmıştır. Bundan dolayı bir hata gelecektir. Ayrıca 3. satırda değer atanmamış “null” değişken çağırılamaz. 3. Satırda hata verecektir.

CEVAP : B

15. Q15 uygulamasıda yeni bir nesne oluşturulduğunda önce static olan satır çağırılır. Daha sonra instance olarak gelen 2. ve 7. satır derlenir, en son public erişim denetleyicisi olan constructor metod derlenir.

CEVAP : C

16. “defaultValue” değişkeni main() metodu içinde yerel değişken olduğundan kendiliğinden initialized edilmemektedir.

Fakat Q15 class’ında `static double defaultValue , static int defaultValue veya static short defaultValue` şeklinde bir tanımlama yapılsaydı “int ve short” için cevap “0” `double` için “0.0” olurdu.

CEVAP : A

17. finalize() metodu çöp toplama mekanizmasını devreye sokup ilgili nesneleri silerek bellekte yer açmaya çalışır. Bu metodla stack alanında bir değişkene bağlanmayan heap’teki nesneler “garbage collector” tarafından finalize() metodu ile silinirken stack alanında bir değişkene refere edilen heap alanındaki nesneler kullanılıyor olarak görülmekte ve silinmemektedir. Bir nesne için finalize() metodu ya 1 kere çağrılır ve nesneyi yok eder, yada hiç çağrılmaz.

CEVAP : A

18. String wrapper class değildir. Wrapper classlar cevap 8’deki gibidir.

CEVAP : D

19. `Q19 q19_1 = new Q19("x", null);`
`Q19 q19_2 = new Q19("y", q19_1);`
`Q19 q19_3 = new Q19("z", q19_2);`
Yukarıda 3 tane nesne oluşturuluyor.

`q19_2.setNext(q19_3);`

`q19_3.setNext(q19_2);`

q19_2 nesnesi q19_3 nesnesini göstermektedir.

`q19_1 = null`

`q19_3 = null;`

2 nesneye ulaşamamaktadır.

Son durumda 1 ve 3 e ulaşamaz “null” ve “q19_2” “q19_3’ü” göstermektedir.

CEVAP : C

20. Byte ve short’ta kayan noktalı gösterim bulunmamaktadır. Double ve Floatta kayan noktalı gösterim vardır. Fakat floatta pi sayısını gösterebilmek için;

`float pi = 3.14f`

`double pi = 3.14`

CEVAP : C

21. Aşağıdaki kelimeler “reserved words” tür. Bu isimler değişken ismi olamaz

abstract	do	if	private
assert	double	implements	protected
boolean	else	import	public
break	enum	instanceof	return
byte	extends	int	short
case	false	interface	static
catch	final	long	strictfp
char	finally	native	super
class	float	new	switch
const	for	null	synchronized
default	goto	package	

CEVAP : B

22. Foo.bar ile instance bir değere erişilebilir, bar bir fonksiyonsa write ve read işlemleri yapılabilir. Fakat yerel değişkene erişilemez.

CEVAP : B

23. Class isimleride değişken isimleri gibi ele alınmalıdır. Cevap 4 ile aynıdır.

CEVAP : C

24. `double d = new Double (0.0);` şeklinde bir tanımlama yapılabilirdi.

CEVAP : D

25. Yerel değişkenler default olarak bir değer atanmaz muhakkak kullanılacaksa bir değer atanmalıdır.

CEVAP : C

26. `static double defaultValue; // 0.0 degerini verir.`
`static int defaultValue; // 0 degerini verir.`
`static long defaultValue; // 0 degerini verir.`
`static short defaultValue; // 0.0 degerini verir.`
Double hariç diğer seçeneklerde “0” değeri gözlemlenir.

CEVAP : C

27. primitivler int,double,short gibi deęişken tipleridir.

- Bu tipler arraylist'e Wrapper classları olmadan direk atanamazlar.
- valueOf() metodu ile String'te bulunan deęerin Wrapper metoda ait deęerine erişilebilir.
- Primitiv tiplerden Wrapper classlara kolayca dönüşüm sağlanabilir.

CEVAP : B

28. Q28 uygulamasındaki gibi primitiv deęişkenlerden metod() çağırılmaz.Bundan dolayı son satır derlenmez.

CEVAP : C

29. Q29 uygulamasındaki gibi kurucu metod yeni bir nesne oluşturulduğunda ilk çalışan kod parçacıdır.

new Q29();

CEVAP : D

30. String cat = "animal", dog = "animal"; doğru gösterimdir.

String cat = "animal"; dog = "animal"; “;” ile ayrıldığından 2. String belirtilmelidir.

String cat, dog = "animal"; doğru bir gösterim değildir.

String cat, String dog = "animal"; cat deęişkenine deęer atanmalıdır.

CEVAP : A

31. 8. Soru incelemesindeki tabloya bakıldığında “char” ve “int” deęişken tiplerinin Wrapper classları Character ve Integer dięer deęişken tipleri için baş harf büyük yazılarak Wrapper class oluşturulmuştur.

CEVAP : C

32. String deęişkene “**null**” atanabilir. Farklı deęişkenler atanabilir. Bu atamalar sınıfın dıęından ve constructor olmadan yapılabilir.

CEVAP : A

33. Bütün primitive tipler küçük harf ile başlamaktadır. String primitive type değildir.

CEVAP : A

34. Çöp toplama mekanizması system.gc(); ifadesi ile çalıştırılabilir.

CEVAP : B – CEVAP : D

35. Q35 uygulamasında belirtildięi gibi bütün “fruit” nesneleri “apple” String ifadesini barındırmaktadır. Bundan dolayı fruit2 ve fruit3 silinebilir.

CEVAP : C

36. new işleminden sonra Wrapper class ismi verilmelidir. Sadece “B” seçeneğinde bu kurala uyulmuştur. Hem primitiveden hemde Wrapper classtan nesne oluşturulabilir.

CEVAP : B

37. Q37 uygulamasındaki gibi bir classtan yeni bir nesne oluşturulduğunda ilk çalışan metod constructor'dır. Bu uygulamada print() metodu çalıştırılması için o class önce constructor() metodu çalıştıracaktır.

CEVAP : B

38. Primitive tiplere "**null**" atanamaz. Wrapper classtan oluşturulmuş nesnelere "**null**" değer atanabilir.

CEVAP : C

39. Instance metoda Instance ifadeler dışında erişilemezken static metodlara hem static hem instance olarak erişilebilir.

CEVAP : C

40. "_" underscore ifadesi 2 rakam arasında olmadığında hata verecektir.

CEVAP : B

41. byte "1 byte"
short "2 byte"
int "4 byte"
long "8 byte"

CEVAP : A

42. Bir classtaki değişene ulaşabilmek için "." kullanılır. Q42 uygulamasındaki gibi String ifadeye değer atamak için `q42.name = "Sadie"`; kullanılmıştır.

CEVAP : A

43. Finalize() metodu ile finalizer() metodu aynı değildir. finalizer() kullanıcı tarafından girilen bir metoddur. Q43 uygulamasına bakılmalıdır.

CEVAP : B

44. "[]" dizi tanımlamasında kullanılan ayraçtır. Q44 uygulamasına bakılmalıdır.

CEVAP : A

45. `valueOf()`; metodu Wrapper classlar ile birlikte kullanılır. `parseInt()`; metodu ise Wrapper classtan primitive değişkenlere değer ataması gerçekleştirir.

CEVAP : B

46. Q46 uygulamasına bakıldığında elena nesnesi diana nesnesine refere etmektedir. Daha sonra diana nesnesi "null" edilmiştir. sonra elana nesnesine zoe nesnesi refere ettirildiğinden diana nesnesi çöp konumuna geçmiştir.

CEVAP : B

47. Constructor() metodlar class ismi ile aynı isimde olup dönüş değerine sahip olmazlar. Bir nesne oluşturulduğunda ilk olarak bu metod çağırılır.

CEVAP : C

48. Garbage collector finalize() metodunu kullanır ancak bu kesin değildir. Hafızanın doluluk oranına göre istediği zaman çalışabilmektedir. Q48 uygulamamda ben garbage collector ile finalize() metodunun içine giriş yapamadım. Bellek kullanımım çok düşük olduğundan bu metoda belkide JVM kapanma durumunda girecektir. Fakat bu 2 nesnede play metodunu kesinlikle çağıracaktır.

CEVAP : A

49. Wrapper classlarda diğer classlar gibi önce constructor metodunu çalıştıracaktır.

CEVAP : B

50. Q50 uygulamasındaki gibi constructor önceliklendirmesine göre çıktı sağlanır.

CEVAP : C