S1: Java platformu neden bağımsızdır?

C1: Herhangi bir sistemde çalışabilen bayt kodlarından dolayı.

​

S2: Java neden %100 nesne yönelimli değil?

C2: Çünkü nesne olmayan 8 primitive data türünü kullanır.

//int, byte, short, long, float, double, boolean, and char.

​

S3: public static void main (String [] args) Java'da yer alan bu yapıyı açıklayınız.

C3: Bu, herhangi bir Java programı için giriş noktasıdır

public: bu yönteme kimin erişebileceğini belirtmek için kullanılan bir erişim değiştiricidir ve bu yöntemin herhangi bir sınıf tarafından erişilebilir olacağı anlamına gelir.

static: Java'da sınıf tabanlı olduğunu tanımlayan bir anahtar kelimedir. main (), bir sınıfın oluşumunu oluşturmadan erişilebilmesi için Java'da static yapılır.

void: methodun return type dır.

main (): JVM tarafından aranan methodun adıdır. JVM: Java sanal makinesi

String args ise main methoda iletilen parametredir.

'S4: Wrapper Class'lar nelerdir?'C4: Wrapper Class'lar Java primitive datalarını o sınıfın başvuru türl'rine veya o

Class'ın nesnelerine dönüştürürler.

''S5: Java da Constructor nedir?

C5: Bir nesneyi başlatmak için kullanılan bir kod bloğunu ifade eder. Sınıfın adıyla

aynı ada sahip olmalıdır.

'S6: Kaç çeşit Constructor vardır?

C6:'2 çeşit Constructor vardır.

1) Default Constructor: herhangi bir girdi ya da değer almaz.

2) Parmeterized Con'tructor: durum değişkenlerini sağlanan değerlerle başlatabilir.

''S7: Bir Class i nasıl singleton yapabiliriz?

C7: O Class'ta ki Constructor i priva'e yaparak.

'//Singleton sınıfının birincil amacı, nesne oluşturma sayısının sınırını yalnızca bir ile sınırlamaktır. Bu genellikle soket veya veritabanı bağlantısı gibi kaynaklara erişim kontrolünün olmasını sağlar.

'//Örneğin, bizde bir lisans var ve sadece bir veritabanı bağlantımız var veya varsayalım ki JDBC sürücümüz multithreading yapmamıza izin vermiyorsa, o zaman Singleton sınıfı devreye giriyor ve bir seferde yalnızca tek bir bağlantı olmasını sağlıyor. veya tek bir iş parçacığı bağlantıya erişebilir.

''''S8: Java da ArrayList ve Vector arasınd'ki fark nedir?

C8: ArrayList hızlı olması için senkronize edilmemiştir ancak Vector, iş parçacığı açısından güvenli olduğundan yavaş olması için senkronize edilmiştir.

'//A Synchronized class is a thread-safe class. Non-Synchronized means that two or more threads can access the methods of that particular class at any given time. StringBuilder is an example of a non-synchronized class. Generally, a non-synchronized class is not thread'safe.

'//Bir Synchronized s'nif, iş parçacığı açısından güvenli bir sınıftır. Senk'onize Olmayan, iki veya daha fazla iş parçacığının herhangi bir zamanda söz konu'u sınıfın yöntemlerine e'işebileceği anlamına gelir. StringBuilder, senkronize olmayan bir'sınıf örneğidir. Genel olarak, senkronize olmayan bir sınıf iş parçacığı açısından güvenli değildi'.

'//Vector 's synchronized, which means only one thread at a time can access the code, while ArrayList is not synchronized, which means multiple threads can work on ArrayList at the same time. For example, if one thread is performing an add operation, then there can be another thread performing a remove operation in a multithreading environment. If multiple threads access ArrayLi't concurrently, then we must sync'ronize the block of the code which modifies the list structur'lly or allow simple element modifications. Structural modification m'ans the addition or deletion of element(s) from the list. Setting the value of an existing element is not a structural modification.

'//Vektör senkronize edilir, yani aynı anda yalnızca bir iş parçacığı koda erişebilirken, ArrayList senkronize edilmez, bu da birden çok i' parçacığının ArrayList üzerinde aynı anda çalışabileceği anlamına gelir. Örneğin, bir iş parçacığı ekleme işlemi gerçekleştiriyorsa, çok iş parçacıklı bir ortamda kaldırma işlemi gerçekleştiren başka bir iş parçacığı olabilir. Birden fazla iş parçacığı ArrayList'e eşzamanlı olarak erişiyorsa, listeyi yapısal olarak değiştiren veya basit öğe değişi'liklerine izin veren kod bloğunu senkronize etmeliyiz. Yapısal değişiklik, listeden eleman(lar)ın eklenmesi veya silinmesi anlamına gelir. Mevcut bir öğenin değerini ayarlamak, yapısal bir değişiklik değildir.

​

​

S9: Java da equals () ve = = işareti arasındaki fark nedir?

C9: equals () operatörü primitive dataları ve Objectleri karşılaştırmak için kullanılır. == ise iki tane Object i karşılaştırmak için kullanılır.

​

​

s10: Java'da Heap ve Stack Memory arasındaki farklar nelerdir?

C10: Heap hafızası bir Application in tüm parçaları tarafından kullanılır. Objectler Heap içinde depolanır ve herkesçe genel kullanıma müsaittir. Stack hafıza ise sadece bir yürütme dizisi tarafından kullanılır ve diğerleri tarafından asla kullanılamaz.

​

​

S11: Java da package nedir?

C11: İlgili Class'ların toplandığı yerdir.

​

S12: Package'ların avantajları nelerdir?

C12:

​

1. İsim çatışmasını önler.

2. Kodlara kolay ulaşım sağlar.

3. Gizli Class'ları barındırır ki Class dışından görülemesin.

​

S13: Neden pointer lar Java da kullanılmazlar?

C13: Çünkü güvenli değillerdir ve program içindeki karmaşıklığı artırırlar.

//Pointer bir değişkenin adresini taşıyan başka bir değişkendir. Bu adres o değişkenin bellekteki (memory) yerinin adresidir. Bu adres kullanılarak o değişkenin değeri değiştirilebilir.

​

S14: Java da JIT derleyici (compiler) nedir?

C14: JIT derleyici yani tam zamanlı derleyici, Java bayt kodlarını talimatlara

dönüştürmeye yardımcı olan bir programdır.

​

S15: Java da access modifier lar nelerdir?

C15: Default, Private, Public ve Protected.

​

S16: Java'da ki access modifier lari açıklayınız?

C16: Default: ayni package ve class tan ulaşılabilir.

Public: tüm package lar ve class tan ulaşılabilir.

Private: sadece ayni class tan ulaşılabilir.

Protected: sadece farklı pakette ama alt olmayan class tan ulaşılamaz. Diğer şekillerde ulaşılabilir.

​

(Uyarı: Protected a dikkat edin lütfen yani ayni package ya da farklı package olması önemli değil sadece farklı package in alt class in da olmayan dosyadan erişim sağlanamıyor.)

\*Tablo burada daha net anlaşılabilir.

same => ayni;

different => farklı;

subclass - alt class => alt kumesi gibi; non-subclass => alt class i olmayan demek.

​

S17: Java'da class i tarif ediniz?

C17: Class, datalarımızı barındıran içinde field variable'ları ve method'ların olduğu bizim çalışma alanımızdır.

​

S18: Java da obje (Object) nedir ve nasıl oluşturulur?

C18: Obje, örnek değişkenlere sahip sınıfın bir örneğidir. Java'da new anahtar sözcüğü kullanılarak bir Obje oluşturulur.

​

​

​

S19: OOP nesne yönelimli programlama nedir?

C19: Bu bir programlama modeli veya yaklaşımı, büyük ve karmaşık kodlu programlar için idealdir ve aktif olarak güncellenmesi ve sürdürülmesi gerekir. Programlar, mantık ve işlevler yerine objeler etrafında düzenlenmiştir.

​

​

S20: Java da OOPs nin ana konsepti nelerdir?

C20: İnheritance, Encapsulation, Abstraction, Polymorphism.

​

​

S21: Bunları (OOPs nin ana konseptleri) açıklayınız?

C21:

Inheritance kalıtım-miras: Bir sınıfın başka bir sınıfın özelliklerini elde ettiği miras aldığı bir süreçtir.

Encapsulation-kapsülleme: Verileri sarmalama mekanizmasıdır.

Abstraction-soyutlama: Kullanıcıdan var olan detayları gizleme metodolojisidir.

Polymorphism: Bir değişkenin birden çok form alma yeteneğidir.

​

S22: Local variables ile instance variables arasındaki fark nedir?/team24

C22: Local variables: tipik olarak bir method, constructor veya block içinde kullanılır ve yalnızca yerel kapsamı vardır.

Instance variables: objesinin kendisine bağlı bir değişkendir. Bir class içinde ancak method dışında bildirilirler.

​

S23: Java da Constructors ve Methods arasındaki farklar nelerdir?

C23:

1. Constructors: Bir objenin durumunu başlatmak için kullanılır.

Methods: Bir objenin davranışını temsil etmek için kullanılır.

2. Constructors: Herhangi bir dönüş türü (return type) yoktur.

Methods: Bir (return type) dönüş türüne sahip olmalıdır.

3. Constructors: (Örtülü) İmplicitly olarak çağrılır

Methods: (Açıkça) Explicitly çağrılması gerekiyor

4. Constructors: Class adıyla aynı isimde olmak zorundadır.

Methods: Olabilir veya olmayabilir de.

​

​

S24: Java'da final keyword unu açıklayınız?

C24: Final Java özel bir keyword dur ve non-access modifier (erişim olmayan

değiştirici) olarak kullanılır.

​

S25: Final keyword un kullanımını açıklayınız?

C25: Final keywordunu bir variable ile kullandığımızda onun value yani değeri bir kez atandığı için değiştirilemez.

Final Methodlar override edilemez. Final Class'lar extends edilemez.

​

​

S26: Break ve Continue yapıları arasındaki farklar nelerdir? C26:

1. Break Switch ve Loop'larda kullanılır.

Continue sadece Loop'larda kullanılır.

2. Break sistem çalışırken bulunduğu işlemi sonlandırır.

Continue işlemi sonlandırmaz sadece sıradaki diğer seçeneğe atlar.

​

​

​

S27: Java'da sonsuz döngü (Loop) nedir?

C27: Java'da bir işlevsellik karşılanmadığında, bir talimat dizisi tarafından sonsuz döngüye girilir.

​

​

S28: Java'da this() ve super() anahtar kelimeler arasındaki fark nedir?

C28: This() aynı class in varsayılan Constructor'larını çağırmak için kullanılır, ancak super() onları ebeveyn / temel class tan çağırmak için kullanılır.

​

Not: this () ve super () (anahtar sözcükleri) keywords bir bloğun ilk satırında yer almalıdırlar.

​

S29: Java da String Pool nedir?

C29: Heap hafıza içinde String'lerin toplandığı havuz yani yerdir. Bir Object oluşturulduğu zaman onun bu havuzda önceden olup olmadığını kontrol eder.

​

S30: Static method ve non-static methodlar arasındaki farklar nelerdir?

C30:

1. Static method: Static anahtar sözcüğü method adından önce kullanılmalıdır. Ancak non-static method anahtar kelimesini statik kullanmaya gerek yoktur.

2. Static method: Class kullanılarak çağrılır

Ancak non-static methodlar genel methodlar gibi çağrılabilir.

3. Static method: Statik olmayan değişkenlere (variables) erişilemez Ancak non-static method bunlara erişebilir.

​

S31. String'ler ve StringBuilder'lar arasındaki farklar?

Q31: 1. saklama alanı; String'ler StringPool'da

StringBuilder Heap hafızada

2. İş parçacığı güvenliği; String'ler evet

3. Form;

String'ler Değişmez

StringBuilder değiştirilebilir

4. Performans;

String'ler Hızlı

StringBuilder daha verimli, özellikle setter ve getter methodlarinda.

​

​

S32: Constructor'lar inherited edilebilir mi?

C32: Hayır edilemez.

​

S33: ClassLoader nedir?

C33: JVM in altkümesi ve class dosyalarının yüklenmesinden sorumlu yapı.

​

S34: Kaç tane ClassLoader vardır ve nelerdir?

C34: Java 3 tane ClassLoader destekler

1. Bootstrap

2. Extension

3. System / Application

:pray::pray::skin-tone-2:

6

​

​

​

Seren TechPro Mentor

1:45 PM

S35: Java'da String'ler neden doğası gereği değişmezdir?

C35: Basitçe, String objesi oluşturulduktan sonra durumunun değiştirilemeyeceği anlamına gelir. Uygulamanın güvenlik önbelleğini, senkronizasyonunu ve performansını artırır.

​

​

S36: Array ve ArrayList arasındaki farklar nelerdir?

C36:

1. Array, farklı veri türlerinin değerlerini içeremez.

ArrayList bunları içerebilir.

2. Array size'ı tanımlanmalıdır.

ArrayList size'ı dinamik olarak değiştirilebilir.

3. Veri eklemek için Array'in indeksi belirtmesi gerekir.

ArrayList buna ihtiyacı yok.

4. Array'ler primitive data ve nesneler (Objects) içerebilir.

ArrayList yalnızca objeleri içerebilir.

​

S37: Java da Map nedir?

C37: Map, benzersiz anahtarları (keys) değerlerle (values) eşleştiren bir Util Paketi ara yüzüdür.

Yinelenen anahtarlar (key) içermez

Her anahtar (key) en fazla bir değerle(value) eşlenebilir

​

​

S38: Java da Collection Class nedir?

C38: Collection bir mimar gibi bir grup nesne objeyi depolamak ve işlemek için hareket eden bir çerçeveden framework tan oluşur.

​

S39: Collection Class'lar neler içerirler açıklayınız?

C39: Interface, class ve methodlar içerirler. List, queue ve set en önemli parçalarıdır Collection Class'ların.

​

S40: Polymorphism nedir?

C40: Bu soruya en iyi örnek; kamera, hesap makinesi, mp3 çalar yada uzaktan kumanda gibi kullanılan bir cep telefonudur.

​

​

Q41: Kaç çeşit Polymorphism vardır?

C41: 2 çeşittir;

1- Compile time Polymorphism method overloading dir.

2- Run Time Polymorphism dir inheritance and interface tarafından yapılır.

​

​

​

S42: Java'da Abstraction (soyutlama) nedir?

C42: Temelde ayrıntıları gizlemek ve önemli olanları kullanıcıya göstermekle

ilgilenir. İki yol olabil...